WBM-5000 DLL interface function description

1、DLL Document Specifications

1.1、DLL description

Document name：WBM_5000.dll

Version：V1.0

Function description： For WBM-5000 development。

Card Standard：

ISO-7811/12 magnetic card

ISO-7816 T=0 CPU card

ATMEL、SIEMENS series card

PHILIPSM1 S50/S70 RF card

1.2、 Get the version information

int APIENTRY GetSysVerion(HANDLE ComHandle， char *strVerion);

Parameter:

ComHandle：Comhandle of the opened serial interface。

strVerion: Store the version information string

Return value: Right=0, Wrong=not 0

2、Card machine status code and function error return code

2.1、function error return code description

Return code	Description
-101	Comhandle error
-1	Communication Error
0x4E	Communication is ok，but Execute command error（The error code in the card operation function）

ERR_CODE list description:

Code	Explanation
0X30	Can't find the RF Card
0X31	Operation sectors wrong（not the sector after the password be confirmed）
0X32	Operation card serial number wrong
0X33	Password confirmation wrong
0X4E	Operation failed
0X45	No card in machine
0X46	Card scraped
0X57	The location of the card is not allowed to operation

Note：0 stands not return the error code。

3、Serial port operation basic function

3.1、Open the serial interface（the default Baude rate is 9600bps）

HANDLE APIENTRY CommOpen(char *Port);

Parameter:

Port:the open serial interfaces, for example, open com1, then *Port will store “com1”

Note:1.Must call this function，To obtain a serial port’s serial port file comhandle，Then you can call other functions。

2.You can open multiple serial ports at the same time for multiple file handles serial port，But can't open the same serial many times。

3.End the using，You must call CommClose() to close the serial port。

3.2、Open the serial interface according to the corresponding baud rate

HANDLE APIENTRY CommOpenWithBaut(char *Port, unsigned int _data);

Parameter:

Port:the open serial interfaces, for example, open com1, then *Port will store “com1”

_data =

9600=> valid value9600

19200=> valid value19200

38400=> valid value38400

Return value: return the comhandle of the serial interface correctly, error=0:

Note:1.Must call this function，To obtain a serial port’s serial port file comhandle，Then you can call other functions。

2.You can open multiple serial ports at the same time for multiple file handles serial port，But can't open the same serial many times。

3.End of the using，You must call CommClose() to close the serial port。

3.3、Close the corresponding serial port

int APIENTRY CommClose(HANDLE ComHandle);

Parameter:

ComHandle：the comhandle will be closed。

Return value:Right=0, Wrong=not 0

Note:Use with CommOpen() function，and end of the using，you must call it to close the serial port。

4、Card machine operation function

4.1、Read card status information from the card machine

int APIENTRY WBM5000_CardState(HANDLE ComHandle, BYTE *State1, BYTE *State2, BYTE *State3);

Parameter:

ComHandle：

ComHandle file，refer to CommOpen（）function。

output parameter_***State1：**

Code	Explanation
0X46	Long card in the card machine(the card length is longer than standard)
0X47	Short card in the card machine(the card length is shorter than standard)
0X48	card in the front-end no card location（means capture the card in the front-end）
0X49	Card in the front-end card position
0X4A	Card in the stop card position
0X4B	Card in the IC card position，and IC card down after getting the electronic shock
0X4C	Card in the back-end card position
0X4D	Card in the back-end no card position（means capture the card in the back-end,recover the card）
0X4E	No card in the card machine

output parameter_***State2：**

Code	Explanation
0X49	Allow the card into the machine in the magnetic card way, only magnetic card be allowed to open the gate into .
0X4A	Allow the card into the machine in the on-off mode，magnetic card,IC card,M1 RF card and Combi-card are allowed into.
0X4B	Allow the card into the machine in the magnetic signal way,paper magnetic card,thin card are allowed into.
0X4E	Card is forbidden into

output parameter ***State3：**

Code	Explanation
0X4A	Allow the card in the back-end into card，magnetic card,IC card,M1 RF card and Combi-card are allowed into.
0X4E	Card is forbidden into the back-end

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

4.2、Read sensor status from the card machine

int APIENTRY WBM5000_SensorState(HANDLE ComHandle, BYTE SenState[], BYTE *ERR_Code);

Parameter:

ComHandle：

ComHandle file，refer to CommOpen（）function。

output parameter **SenState[]:**

The sensor state data.

SenState[]The data format is as follows：

PSS1	PSS2	PSS3	PSS4	PSS5	CTSW	KSW

PSS1—PSS5： Infrared sensor state：

PSS1（2…5）=0X30 means the sensor location did't detect the card；

PSS1（2…5）=0X31 means the card been detected。

CTSW:Gate status information

CTSW=0X30 means gate has been closed；

CTSW=0X31 means gate has been opened。

KSW:sensor state when via on-off mode into the card KSW=0X30 means on-off didn't detect the signal of card inserted into the gate；

KSW=0X31 means on-off detected the card inserted to the gate。

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

4.3、Reset and upload card machine version number

int APIENTRY WBM5000_ResetVerion(HANDLE ComHandle, BYTE _Verion[]);

Parameter:

ComHandle：

ComHandle file，refer to CommOpen（）function.

output parameter **_Verion[]:**

card machine version number.

Return value: Right=0, wrong=not 0

4.4、Reset and remove card

int APIENTRY WBM5000_ResetMove(HANDLE ComHandle, BYTE Reset_Type);

Parameter:

ComHandle：

ComHandle file,refer to CommOpen（）function.

input parameter **Reset_Type:**

Reset options。
0X30:Initialization card reader, eject card from front(Don't upload version information)；
0X31: Initialization card reader, capture the card to error card bin(Don't upload version information)；
0X32:Initialization card reader, move the card to mag-card position(Don't upload version information)
0X33:Initialization card reader, move the card to IC card position(Don't upload version information)。

output parameter ***ERR_Code**：
error code。

Return value: Right=0, wrong=not 0

4.5、Card stop location setting

int APIENTRY WBM5000_CardStop(HANDLE ComHandle, BYTE Stop_Type, BYTE *ERR_Code);

Parameter:

ComHandle：
ComHandle file,refer toCommOpen（）function。

input parameter **Stop_Type:**
Pm=0x30 Card stop in the front-end,no card。
Pm=0x31 Card stop in the front-end,has card。
Pm=0x32 card stop inside,but IC card contactor not contact with the card， M1 RF card can read and write
Pm=0x33 card stop inside,and IC card contactor contact with the card,then IC card and M1 rf card can operating directly.
Pm=0x34 card stop in the back-end,has card.
Pm=0x35 capture the card in the back-end,no card.

output parameter ***ERR_Code**：
error code。

Return value: Right=0, wrong=not 0

4.6、move card to the specified location

int APIENTRY WBM5000_CardMove(HANDLE ComHandle, BYTE Move_Type, BYTE *ERR_Code);

Parameter:

ComHandle：
ComHandle file,refer toCommOpen（）function。

input parameter **Move_Type:**
Move the card to the specified location。
Pm=0x2E Move the card to the inside again ,after successful operation,then processed the M1 RF card operation.
Pm=0x2F Move the card to the inside again ,and contact with the IC contactor,after successful operation,then processed the IC card operation.
Pm=0x30 Move the card to the front again,no card。
Pm=0x31 Move the card to the front again,has card。
Pm=0x32 Move the card to the back again,has card。
Pm=0x33 Move the card to the back again,no card。

Pm=0x34 Clear out the abnormal length card (long card,short card) from inside,capture the card in the back,short card need manual auxiliary operation with inserting the card in the mount.(this command can be used to clear the card machine inside)

output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

4.7、Card entering setting

int APIENTRY WBM5000_CardIN(HANDLE ComHandle, BYTE CardIn_Type1, BYTE CardIn_Type2, BYTE *ERR_Code);

Parameter:

ComHandle：

ComHandle file,refer toCommOpen（）function。

input parameter **CardIn_Type1:**

Pm1=0x31 Prohibit card front entering

Pm1=0x32 magnetic card way entering（Magnetic signals + on-off effectively at the same time）, only allow magnetic card entering from front gate.

Pm1=0x33 on-off mode entering，allow magnetic card ,IC card,M1 RF card and Combi-card entering from front gate

Pm1=0x34 magnetic signal entering,For thin magnetic card like paper card entering.

CardIn_Type2:

Pm2=0x30 Allow card back entering .

Pm2=0x31 Prohibit card back entering.

output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

5、magnetic card operation function

5.1、Read magnetic card decoding data

int APIENTRY WBM5000_MagCardReadData(HANDLE ComHandle, BYTE Mode, BYTE Track_Type, BYTE *RLEN, BYTE TrackData[]);

Parameter:

ComHandle：Comhandle of the opened serial interface。

input parameter **Mode:**

0x30 read data based on ASCII

0x31 read data based on binary code

Track_Type:

0x30 Read no data

0x31 Read ISO track 1 data

0x32 Read ISO track 2 data

0x33 Read ISO track 3 data

0x34 Read ISO track 1&2 tracks data

0x35 Read ISO track 2&3 tracks data

0x36 Read ISO track 1&3 tracks data

0x37 Read ISO track 1&2&3 tracks data

output parameter ***RLEN:**

store the length of returned data packet。

output parameter **TrackData[]:**

Magnetic card data packet：

Track1 data packet n byte +Track2 data packet n byte +Track3 data packet n byte

Description：

Each track packet format is as follows：

Track data start word+read card status word+card track data

Track data start word： 0x1F

read card status word： 0x59 track data is correct，the card track data is the track information data.

0x4E card read is wrong，the card track data is wrong data.

0x4F the track don't need read，then the card track data is 0xE0；

Error information： 0xE1 the track data read error，not have the start bit STX

0xE2 the track data read error，not have the end bit ETX

0xE3 the track data read error，bit check error VRC

0xE4 the track data read error，byte check bit error LRC

0xE5 the track data read error，the track data is blank .

When setting read card based on ASCII,convert every track information to a ASCII byte,then upload the card information

for example：The first byte of track data： 0x03 (HEX)

The track data packet when upload data： 0x33 (ASCII)

When setting read card based on binary system,convert every byte data of every track information to a ASCII byte with four per to upload the information.

for example：The first byte of track data： 0x03 (HEX)

The track data packet when upload data： 0x30 0x33

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

6、RF card operation function

6.1、Rf card getting serial number

int APIENTRY WBM5000_RFGetSN(HANDLE ComHandle, BYTE SNData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

output parameter **SNData[]:**

Card serial number， four bytes。

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

6.2、RF card Scanning

int APIENTRY WBM5000_RFScanCard(HANDLE ComHandle, BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

output parameter **ReadData[]:**

Card type, two bytes。

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

6.3、RF card password checking

int APIENTRY WBM5000_RFCheckPW(HANDLE ComHandle, BYTE Mode, BYTE Sector, BYTE PassWord[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

input parameter **Mode:**

Checking mode： = 0x32 KeyA

= 0x39 KeyB。

input parameter **Sector:**

Sector number。Sector number= 0x00 0x1 0x02 ……0x0F

input parameter **PassWord[]:**

password ： 6 bytes。

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

6.4、RF card change the password

int APIENTRY WBM5000_RFChangePW(HANDLE ComHandle, BYTE Sector, BYTE PassWord[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

input parameter **Sector:**
Sector number。Sector number= 0x00 0x1 0x02 ……0x0F

input parameter **PassWord[]:**
password ： 6 bytes。

output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

Note：Execute this command only can change the password of KEYA,and change the password to:“0xFF, 0xFF, 0xFF,0xFF,0xFF,0xFF”meanwhile control word is：“0xFF, 0x07, 0x80, 0x69” (default values of the card factory).。

6.5、RF card reading data

int APIENTRY WBM5000_RFReadData(HANDLE ComHandle, BYTE Sector, BYTE Block, BYTE ReadData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

input parameter **Sector:**
Sector number。Sector number= 0x00 0x1 0x02 ……0x0F

input parameter **Block:**
Block number。Block number＝ 0x00 0x01 0x02 0x03

output parameter **ReadData[]:**
Return the data reading packet,16bytes。

output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

6.6、RF card writing data

int APIENTRY WBM5000_RFWriteData(HANDLE ComHandle, BYTE Sector, BYTE Block, BYTE Block_Data[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

input parameter **Sector:**
Sector number。Sector number= 0x00 0x1 0x02 ……0x0F
input parameter **Block:**
Block number。Block number＝ 0x00 0x01 0x02 0x03
input parameter **Block_Data[]:**
Return the data writing packet,16bytes。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

6.7、Rf card adding value operation

int APIENTRY WBM5000_RFAddVal(HANDLE ComHandle, BYTE Sector, BYTE Block, BYTE Val_Data[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

input parameter **Sector:**
Sector number。Sector number= 0x00 0x1 0x02 ……0x0F
input parameter **Block:**
Block number＝0x00 0x01 0x02 0x03
input parameter **Val_Data[]:**
Value，four bytes。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

6.8、RF card decrease value operation

int APIENTRY WBM5000_RFDecVal(HANDLE ComHandle, BYTE Sector, BYTE Block, BYTE Val_Data[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

input parameter **Sector:**
Sector number。Sector number= 0x00 0x1 0x02 ……0x0F
input parameter **Block:**

Block number＝0x00 0x01 0x02 0x03

input parameter **Val_Data[]:**

value，four bytes。

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

6.9、RF card initialization value operation

int APIENTRY WBM5000_RFInitVal(HANDLE ComHandle, BYTE Sector, BYTE Block, BYTE Block_Data[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

input parameter **Sector:**

Sector number。Sector number= 0x00 0x1 0x02 ……0x0F

input parameter **Block:**

Block number＝0x00 0x01 0x02 0x03

input parameter **Block_Data[]:**

To be written data packet，16 bytes。

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

Note：

initialization value：According to MIFARE value data format enter write 16 bytes data,the format is as follows:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Value				/Value				Value				Adr	/Adr	Adr	/Adr

Va lue ： initialize 4 byte value, kindly to note the low byte is ahead and the high byte in back

/Value: initialize 4 byte value in opposite

Adr: the block address need to initialization value ： Adr= sector number X 4 + block number

/Adr: the block address need to initialization value is opposite

6.10、RF card reading operation

int APIENTRY WBM5000_RFReadVal(HANDLE ComHandle, BYTE Sector, BYTE Block, BYTE ReadData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

input parameter **Sector:**

Sector number。Sector number= 0x00 0x1 0x02 ……0x0F

input parameter **Block:**

Block number＝0x00 0x01 0x02 0x03

output parameter **ReadData[]:**
Return block data packet,16bytes。

output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

Note：

Data format is as follows：

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Value				/Value				Value				Adr	/Adr	Adr	/Adr

Va lue : 4 byte value, kindly to note the low byte is ahead and the high byte in back

/Value: 4 byte value in opposite

Adr: the block address of the value： Adr=sector number X 4 + block number

/Adr: opposite value of the block address

7、IC card operation function

7.1、IC card power on

int APIENTRY WBM5000_ICPowerON(HANDLE ComHandle, BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

。

7.2、IC card power off

int APIENTRY WBM5000_ICPowerOFF(HANDLE ComHandle, BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.3、CPU card resetting

int APIENTRY WBM5000_CPUCardReset(HANDLE ComHandle, BYTE *RLEN, BYTE ResetData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

output parameter ***RLEN:**
1 byte length，The length of resetting information；
output parameter **ResetData[]:**
The return data packet of resetting。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.4、CPU card T=0 APDU command

int APIENTRY WBM5000_CPUCardAPDU(HANDLE ComHandle, BYTE APDUSendData[], BYTE *RLEN, BYTE APDURecData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface。

input parameter **APDUSendData[]:**
APDU command。
output parameter ***RLEN:**
The length of return data packet，1 byte；
output parameter **APDURecData[]:**
The return data packet。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.5、SIM card resetting

int APIENTRY WBM5000_SIMCardReset(HANDLE ComHandle, BYTE *RLEN, BYTE CardNumber, BYTE ResetData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **CardNumber:**
SIM cassette number。
output parameter ***RLEN:**
1 byte length，The length of resetting information；
output parameter **ResetData[]:**
The return data packet of resetting。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.6、SIM card T=0/T=1 APDU command

int APIENTRY WBM5000_SIMCardAPDU(HANDLE ComHandle, BYTE CardNumber, BYTE APDUSendData[], BYTE *RLEN, BYTE APDURecData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **CardNumber:**
SIM cassette number。
input parameter **APDUSendData[]:**
APDU command。
output parameter ***RLEN:**
The length of return data packet，1 byte；
output parameter **APDURecData[]:**
The return data packet。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.7、AT24xx read data

int APIENTRY WBM5000_AT24XXRead(HANDLE ComHandle, BYTE CardType, BYTE RLEN, BYTE ADDRH, BYTE ADDRL, BYTE ReadData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **CardType:**
Card type
0x30: AT24C01
0x31: AT24C02
0x32: AT24C04
0x33: AT24C08
0x34: AT24C016
0x35: AT24C032
0x36: AT24C064
input parameter **RLEN:**
1 byte length，the length of reading data；
input parameter **ADDRH:**
Initial address high byte。
input parameter **ADDRL:**
Initial address low byte。
output parameter **ReadData[]：**
The return data packet。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.8、AT24xx write data

int APIENTRY WBM5000_AT24XXWrite(HANDLE ComHandle, BYTE CardType, BYTE WLEN, BYTE ADDRH, BYTE ADDRL, BYTE WriteData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **CardType:**
Card type
0x30: AT24C01
0x31: AT24C02
0x32: AT24C04
0x33: AT24C08
0x34: AT24C016
0x35: AT24C032
0x36: AT24C064

input parameter **WLEN:**
1 byte length，the length of writing data。

input parameter **ADDRH:**
Initial address high byte。

input parameter **ADDRL:**
Initial address low byte。

input parameter **WriteData[]：**
Writing data packet。

output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.9、AT24xx write data with checking

int APIENTRY WBM5000_AT24XXCheckWrite(HANDLE ComHandle, BYTE CardType, BYTE WLEN, BYTE ADDRH, BYTE ADDRL, BYTE WriteData[], BYTE ReadData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **CardType:**
Card type
0x30: AT24C01
0x31: AT24C02
0x32: AT24C04
0x33: AT24C08
0x34: AT24C016
0x35: AT24C032
0x36: AT24C064

input parameter **WLEN:**
1 byte length，the length of writing data。

input parameter **ADDRH:**
Initial address high byte。

input parameter **ADDRL:**

Initial address low byte。

input parameter **WriteData[]：**

Writing data packet。

output parameter **ReadData[]：**

The return data packet。

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

7.10、SLE4442 resetting

int APIENTRY WBM5000_SLE4442Reset(HANDLE ComHandle, BYTE RecData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

output parameter **RecData[]：**

Resetting data packet 4 byte。

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

7.11、SLE4442 read data

int APIENTRY WBM5000_SLE4442Read(HANDLE ComHandle, BYTE RLEN, BYTE ADDR, BYTE ReadData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **RLEN:**

The length of reading data,1 byte length；

input parameter **ADDR:**

Initial address of the reading data，1 byte length；

output parameter **ReadData[]:**

Return data packet。

output parameter ***ERR_Code：**

error code。

Return value: Right=0, wrong=not 0

Note：

ADR=00-FF L =0x01—0x80

The operation length L=0X01~0X80 ， The minimum length is 1 BYTE，The maximum length is 128 BYTE .

4442 main storage area only 256 byte Please note the operation address and length should be in the allowed range.

7.12、SLE4442 write data

int APIENTRY WBM5000_SLE4442Write(HANDLE ComHandle, BYTE WLEN, BYTE ADDR, BYTE WriteData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **WLEN:**
The length of writing data,1 byte length；
input parameter **ADDR:**
Initial address of the writing data， 1 byte length；
input parameter **WriteData[]:**
The data packet to be written。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.13、SLE4442 checking password

int APIENTRY WBM5000_SLE4442CheckPW(HANDLE ComHandle, BYTE PassWord[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **PassWord[]:**
Password： 3 byte。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.14、SLE4442 change the password

int APIENTRY WBM5000_SLE4442ChangePW(HANDLE ComHandle, BYTE PassWord[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **PassWord[]:**
Password： 3 byte。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.15、SLE4442 write the protection bit

int APIENTRY WBM5000_SLE4442WriteProtect(HANDLE ComHandle, BYTE WLEN, BYTE ADDR, BYTE WriteData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **WLEN:**
The length of writing data,1 byte length；
input parameter **ADDR:**
Initial address of the writing data， 1 byte length；
input parameter **WriteData[]:**
The data packet to be written。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.16、SLE4442 read the protection bit

int APIENTRY WBM5000_SLE4442ReadProtect(HANDLE ComHandle, BYTE ReadData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

output parameter **ReadData[]:**
Return data packet。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.17、SLE4442 read PSC

int APIENTRY WBM5000_SLE4442ReadPSC(HANDLE ComHandle, BYTE ReadData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

output parameter **ReadData[]:**
Return SPC data packet，3 byte。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.18、SLE4428 resetting

int APIENTRY WBM5000_SLE4428Reset(HANDLE ComHandle, BYTE RecData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

output parameter **RecData[]：**
Resetting data packet 4 byte。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.19、SLE4428 reading data

int APIENTRY WBM5000_SLE4428Read(HANDLE ComHandle, BYTE RLEN, BYTE ADDRH, BYTE ADDRL, BYTE ReadData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **RLEN:**
1 byte length，the length of reading data；
input parameter **ADDRH:**
Initial address high byte。
input parameter **ADDRL:**
Initial address low byte。
output parameter **ReadData[]：**
Return data packet。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

Note：
ADR=0000-03FF L =0x01—0x80
The operation length L=0X01~0X80 ， The minimum length is 1 BYTE，The maximum length is 128 BYTE
4428main storage area only 1K byte . Please note the operation address and length should be in the allowed range.。

7.20、SLE4428 writing data

int APIENTRY WBM5000_SLE4428Write(HANDLE ComHandle, BYTE WLEN, BYTE ADDRH, BYTE ADDRL, BYTE WriteData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **WLEN:**
1 byte length，the length of reading data；
input parameter **ADDRH:**
Initial address high byte。
input parameter **ADDRL:**
Initial address low byte。
input parameter **WriteData[]：**
Data packet。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.21、SLE4428 checking password

int APIENTRY WBM5000_SLE4428CheckPW(HANDLE ComHandle, BYTE PassWord[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **PassWord[]：**
2 byte password。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.22、SLE4428 change password

int APIENTRY WBM5000_SLE4428ChangePW(HANDLE ComHandle, BYTE OldPassWord[], BYTE NewPassWord[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **OldPassWord[]：**
2 byte old password。
input parameter **NewPassWord[]：**
2 byte new password。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.23、SLE4428 card writing protection bit

int APIENTRY WBM5000_SLE4428WriteProtect(HANDLE ComHandle, BYTE WLEN, BYTE ADDRH, BYTE ADDRL, BYTE WriteData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **WLEN:**
1 byte length，the length of reading data；
input parameter **ADDRH:**
Initial address high byte。
input parameter **ADDRL:**
Initial address low byte。
input parameter **WriteData[]：**
Data packet。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0

7.24、SLE4428 card reading protection bit

int APIENTRY WBM5000_SLE4428ReadProtect(HANDLE ComHandle, BYTE RLEN, BYTE ADDRH, BYTE ADDRL, BYTE ReadData[], BYTE *ERR_Code);

Parameter:

ComHandle：Comhandle of the opened serial interface.

input parameter **RLEN:**
1 byte length，the length of reading data；
input parameter **ADDRH:**
Initial address high byte。
input parameter **ADDRL:**
Initial address low byte。
output parameter **ReadData[]：**
Return data packet。
output parameter ***ERR_Code：**
error code。

Return value: Right=0, wrong=not 0