API Programmer Guide

1. imprint(Revision V1.00).

2. explain of backtrack value.

2.1 the function return values

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:
0x02	checksum error.
0x03	Not selected COM port
0x04	time out reply
0x05	check sequence error
0x07	Receive error
0x0A	the parameter value out of range

2.2 System Error/Status Codes (0x80-0xFF)

0x80	SET OK. (success)
0x81	SET FAILURE
0x82	Reader reply time out error
0x83	the card do not exist
0x84	the data is error
0x85	the authentication failure
0x86	Unknown Internal Error
0x89	operation error
0x8f	Reader received unknown command
0x90	show the card could not support this command
0x91	show the command format have a mistake
0x92	show the command could not support OPTION form
0x93	show the inputed block is inexistence.
0x94	show the inputed block had been locked
0x95	show Locked the block is not successful

0x96	show the write card operation is not successful.

3、System Commands

```
3.01    HANDLE API_OpenComm(
                                   char      *com,
                                   int        Baudrate);
```

Description

Open the comm port and set the baud rate for further communication with the reader.

Example

Select COM1 and set the baud rate to 115200bps.

```
    int Baudrate=115200;
    handle comhandle;       { the ‘comhandle’ is the serial port handle.}
     comhandle=API_OpenComm("COM1",Baudrate);
      if (comhandle<>0)
       {          //successful      }
      else
       { //Not successful }
```

Input Parameter Description

Com Character pointer to C string of the name of the serial port where the reader is connected. (e.g. COM1, COM2, COM3, COM4)

Baudrate The communication baud rate of serial port

(Possible values : 9600, 19200, 38400, 57600, 115200).

Output Parameter

none

Return value:

HANDLE , succeed to open the serial port handle

if you open it succeed, the return value is the serial port handle.

if you open it unsuccessfull, the return value is 0.

3.02 int API_CloseComm(const HANDLE commHandle);

Description

Close the communication port. The API_CloseComm ( ) should be called to release the serial port before closing the application program.

Input Parameter Description

CommHandle you need to close the serial port handle

Output Parameter

None

Return value:

0 ： closed the serial port in the handle

-1 ： inpurted the handle value is 0, it couldn't close.

```
3.03 int API_SetDeviceAddress(
                         HANDLE          commHandle,
                         int             DeviceAddress,
                         unsigned char   newAddr,
                         unsigned char   *buffer);
```

Description: set the new address for reader, the reader back to the setting address

Input Parameter Description

commHandle,	the serial port handle
DeviceAddress,	formerly system address
newAddr,	new system address
*buffer	buffer send a pointer, it is used to return the received value.

Output Paramete

If Command success

*buffer the read date(It means you have set the new address in this function)

If Command Failure

*buffer System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

```
3.04  int API_SetBandrate(
                       HANDLE          commHandle,
                       int             DeviceAddress,
                       unsigned char   newBaud,
                       unsigned char   *buffer);
```

Description:

set up the baud rate between the reader and PC communication. the baud rate will be saved in EEPROM and it will be a new default baud rate

Input Parameter Description

commHandle the serial port handle

DeviceAddress equipment address

newBaud, require set new baud rate.

baud rate code:

0x00 – 9600 bps

0x01 – 19200 bps

0x02 – 38400 bps

0x03 – 57600 bps

0x04 – 115200 bps

*buffer introduction a finger,back to the received date.

Output Paramete

If Command success

*buffer， new baud rate(It means set new baud rate code in this function)

（0x00 – 9600 bps，0x01 – 19200 bps，0x02 – 38400 bps，0x03 – 57600 bps，0x04 – 115200 bps）

If Command Failure

*buffer System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

```
3.05   int   API_SetSerNum( HANDLE        commHandle,
                       int            DeviceAddress,
                       unsigned char  *newValue,
                       unsigned char  *buffer);
```

Description:

set 8 byte serial number which be supplied by manufactory

Input Parameter Description:

commHandle the serial port handle

DeviceAddress	equipment address
*newValue	8 byte serial number
*buffer	buffer send a pointer, it is used to return the received value.

Output Paramete:

If Command Failure

*buffer	System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

```
3.06   int    API_GetSerNum(
                              HANDLE          commHandle,
                              int             DeviceAddress,
                              unsigned char   *buffer);
```

Description:

read one byte reader address and 8 byte serial number which be supplied by manufactory

Input Parameter Description:

commHandle	the serial port handle
DeviceAddress	equipment address
*buffer	buffer send a pointer, it is used to return the received value.

Output Paramete:

*buffer	buffer[0]	reader address
	buffer[1...8]	8 byte reader serial number

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

```
3.07   int   WriteUserInfo(HANDLE     commHandle,
                           int        DeviceAddress,
                           int        num_blk,
                           int        num_length,
                           char       *user_info);
```

Description:

the reader consist of 4 blocks(each block less than 120 byte),the user data space in all 480 byte. the user could base the requirement to deposited the relevant userinfo into the reader.

Input Parameter Description:

commHandle	the serial port handle
DeviceAddress	equipment address
num_blk	the block number
num_length	data length
*user_info	user information

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

```
3.08   int    ReadUserInfo(HANDLE      commHandle,
                           int          DeviceAddress,
                           int           num_blk,
                           int           num_length,
                           char            *user_info);
```

Description:

Read the date from the reader, the reader consist of 4 blocks

(each block less than 120 byte)

Input Parameter Description:

commHandle	the serial port handle
DeviceAddress	equipment address
int num_blk	the block number
int num_length	data length
*user_info	wait for reading user date

Output Paramete:

*user_info	If Command Failure,then user_info[0] is error code.
	If Command success,then user_info[0..N] is read user information

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

```
3.09 int   GetVersionNum(HANDLE      commHandle,
                         int         DeviceAddress,
                         char        *VersionNum);
```

Description:
  read reader version number

Input Parameter Description:

commHandle	the serial port handle
DeviceAddress	equipment address
*VersionNum	wait for reading version number

Output Paramete:

*VersionNum   If Command Failure,then VersionNum [0] is error code.
if Command success,then VersionNum [0..N] is reading version number

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

```
3.10 int   API_ControlLED(
                 HANDLE          commHandle,
                 int             DeviceAddress,
                 unsigned char   freq,
                 unsigned char   duration,
                 unsigned char   *buffer);
```

Description: the work state of set light,include,light Cyc and repeating times

Input Parameter Description:

commHandle	the serial port handle
DeviceAddress	equipment address
freq	periodicity
duration	times
*buffer	wait for return value parameter

Output Paramete:

*buffer	If Command Failure,then buffer [0] 为 is error code. If Command success ,then buffer [0] is 0x80

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

3.11 int API_ControlBuzzer(

```
                HANDLE          commHandle,
                int             DeviceAddress,
                unsigned char   freq,
                unsigned char   duration,
                unsigned char   *buffer);
```

Description: the work state of setting buzzer,include buzzer work Cyc and repeating times

Input Parameter Description:

commHandle	the serial port handle
DeviceAddress	equipment address
freq	periodicity
duration	times
*buffer	wait for return value parameter

Output Paramete:

*buffer	If Command Failure,,then buffer [0] 为 is error code. If Command success,then buffer [0] is 0x80

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

4 ISO14443 Type-A Commands

4.1 Type-A Commands

4.1.1 int MF_Request(

```
                HANDLE            commHandle,
                 Int              DeviceAddress,
                 unsigned char    inf_mode,
                 unsigned char    *buffer);
```

Description:： send ISO14443 A seeking card instruction

Input Parameter Description：

commHandle	the serial port handle
DeviceAddress	equipment address
mode	seeking card mode

0x01 –Idle mode（operate one card once a time）
0x00 –All mode（operate a lot of card once a time）

*buffer	wait for return value parameter

Output Paramete:

*buffer	If Command FAILURE, then buffer [0] is error code. if Command OK,then buffer [0..1]，return data bunch within 2 byte

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

4.1.2 int MF_Anticoll(

```
              HANDLE              commHandle,
               int                DeviceAddress,
               unsigned   char    *snr,
               unsigned char      &status);
```

Description:：test card quantity, single or much more, and then return the card number within 4 byte(if there are many cards, the only return one of card number

Input Parameter Description：

commHandle the serial port handle
DeviceAddress equipment address
Status send a finger，return to the number of the card
*snr send a finger，return to 4 byte card number

Output Paramete:
If Command success
Status the checked card number(0x00 means have checked a single card,0x01 means have checked more cards
*snr 4 byte card number（snr[0..3]）
If Command Failure
*snr System Error/Status Codes(You can consult the 2.2)

Return value:
0x00 Command OK. (success)
0x01 Command FAILURE

4.1.3 int MF_Select(
HANDLE commHandle,
int DeviceAddress,
unsigned char *snr);

Description:
Select card, make the card as been select stat…

Input Parameter Description：
commHandle the serial port handle
DeviceAddress equipment address
*snr send a finger,input 4 byte card number,and return to 4 byte card numbe

Output Paramete:
Status checked card numbe
*snr 4 byte card number4（snr[0..3]）

Return value:
0x00 Command OK. (success)
0x01 Command FAILURE

4 .1.4 int MF_Halt(
HANDLE commHandle,

```
                    int                    DeviceAddress );
```

Description:

select card,make the card as been break off

Input Parameter Description:

commHandle the serial port handle

DeviceAddress equipment address

Output Paramete:

None.

Return value:

0x00 Command OK. (success)

0x01 Command FAILURE

4.2 Mifare Appilication Commands

```
4.2.1 int    API_PCDRead(HANDLE              commHandle,
                           int               DeviceAddress,
                           unsigned char     mode,
                           unsigned char     blk_add,
                            unsigned char    num_blk,
                            unsigned char      *snr,
                            unsigned char    *buffer);
```

Description:

read the appointed length date at the appointed station

Input Parameter Description:

commHandle the serial port handle

DeviceAddress equipment address

mode， read mode

(Request Idle + Key A mode=00 , Request Idle + Key B mode= 02,

Request All + Key A mode=01 , Request All + Key B mode=03）

（the up number is hex）

blk_add， read block address

num_blk， read block amount

*snr， a finger, transfer eight byte secret key

*buffer wait receive the variable of output finger

Output Paramete:

If Command success

*snr， 4 byte card number

*buffer， the read date (the fact number is： num_blk*16)

If Command Failure

buffer[0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00 Command OK. (success)

0x01 Command FAILURE

```
4.2.2 int   API_PCDWrite(HANDLE          commHandle,
                         int             DeviceAddress,
                     unsigned char       mode,
                     unsigned char       blk_add,
                     unsigned char       num_blk,
                     unsigned char       *snr,
                     unsigned char       *buffer);
```

Description:

Read-in date At appoint station

Input Parameter Description：

commHandle the serial port handle

DeviceAddress equipment address

mode， needs write mode

(Request Idle + Key A mode=00 , Request Idle + Key B mode= 02,
Request All + Key A mode=01 , Request All + Key B mode=03）

blk_add， needs fill in block address

num_blk， needs fill in block number

*snr， wait fill in date

*buffer， afferent pointer sign to output the date

Output Paramete:

If Command success

snr[0..3]， four byte card number

If Command Failure

buffer[0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00 Command OK. (success)

0x01 Command FAILURE

```
4.2.3 int     API_PCDInitVal(
                                    HANDLE          commHandle,
                                    int             DeviceAddress,
                                    unsigned char   mode,
                                    unsigned char   SectNum,
                                    unsigned char   *snr,
                                    int               value);
```

Description:

Initialize card

Input Parameter Description：

commHandle the serial port handle

DeviceAddress equipment address

mode， initialize mode

(Request Idle + Key A mode=00 , Request Idle + Key B mode= 02,
Request All + Key A mode=01 , Request All + Key B mode=03）

SectNum, need initialize fan number

*snr, six byte secret key (introduction as pointer)

value 4 byte initialize the date

Output Paramete:

If Command success

snr[0..3] 4 byte card number

If Command Failure

snr[0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

```
4.2.4 int    API_PCDDec(
                              HANDLE              commHandle,
                              Int                 DeviceAddress,
                              unsigned char       mode,
                              unsigned char       SectNum,
                              unsigned char       *snr,
                              int                 *value);
```

Description:

Devalue work to the fan of the card

Input Parameter Description:

commHandle   the serial port handle

DeviceAddress   equipment address

mode，   work mode

(Request Idle + Key A   mode=00 ,   Request Idle + Key B   mode= 02,
  Request All   + Key A   mode=01 ,   Request All   + Key B   mode=03）

SectNum,   the fan number which need to write value 00-0F

*snr,   6 byte   secret key（introduction as pointer）

value   need decrease value,4 byte length

Output Paramete:

If Command success

snr[0..3]，   4 byte   card number

value[0..3]   date bunch after 4 byte work

If Command Failure

snr[0]   System Error/Status Codes(You can consult the 2.2)

Return value:

0x00   Command OK. (success)

0x01   Command FAILURE

4.2.5 int API_PCDInc (

```
                    HANDLE          commHandle,
                    Int             DeviceAddress,
                    unsigned char   mode,
                    unsigned char   SectNum,
                    unsigned char   *snr,
                    int             *value);
```

Description:

Add value work to appointed fan of the card.

Input Parameter Description：
commHandle the serial port handle
DeviceAddress equipment address
mode， work mode
(Request Idle + Key A mode=00 , Request Idle + Key B mode= 02,
Request All + Key A mode=01 , Request All + Key B mode=03）
SectNum, need add value fan number 00-0F
*snr, 6 byte secret key(introduction as pointer)
value need add value, 4 byte length

Output Paramete:
If Command success
snr[0..3]， 4 byte card number
value[0..3] the date bunch after 4 byte work
If Command Failure
snr[0] System Error/Status Codes(You can consult the 2.2)

Return value:
0x00 Command OK. (success)
0x01 Command FAILURE

```
4.2.6 int   GET_SNR (
                  HANDLE            commHandle,
                  int               DeviceAddress,
                  unsigned char     mode,
                  unsigned char     halt,
                  unsigned char     *snr
                  unsigned char     *value);
```

Description:

Return 1byte single card or numerous card sign，4 byte card number.

Input Parameter Description：
commHandle the serial port handle
DeviceAddress equipment address
mode， mode command （mode command 26 or 52 ）
0x26 –Idle mode（one time only work to one card）
0x52 –All mode（one time can work to many card）

halt, whether need halt card （halt select 00 or 01 ）
00 don’t need perform halt order
01 reader perform halt order

*snr, returned 1byte single card or numerous card sign (if read card fail,return error code)
*value return 4byte card number

Output Paramete:

If Command success

snr[0]， 1 byte single card or numerous
value[0..3] return 4 byte card number

If Command Failure

snr [0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

4.2.7 int MF_Restore(HANDLE commHandle,
int DeviceAddress,
unsigned char mode,
int cardlength,
unsigned char *carddata);

Description:

According the selected mode to send the date

Input Parameter Description：

commHandle the serial port handle
DeviceAddress equipment address
mode， mode command 0x00 —don’t need check CRC
0x01 —need check CRC
cardlength, card date length
*carddata, send time（card date）
incept time（return date）

Output Paramete:

If Command success

carddata[0..N],　　incept the return date

If Command Failure

carddata[0]　　System Error/Status Codes(You can consult the 2.2)

Return value:

0x00　　Command OK. (success)

0x01　　Command FAILURE

5. ISO14443 Type-B Commands

5.1 int　RequestType_B(

HANDLE　　commHandle,

int　　DeviceAddress,

unsigned char　　*buffer)

Description: this order perform REQB order of ISO14443B, get PUPI code of the card

Input Parameter Description：

commHandle　　the serial port handle

DeviceAddress　　equipment address

*buffer,　　the date bunch after card reposition　　（ATQB）

Output Paramete:

If Command success

*buffer,　　the date bunch after card reposition　　（ATQB）

buffer[0]　　card reposition date length

buffer[0..N]　　the date after work（ATQB）

If Command Failure

buffer[0]　　System Error/Status Codes(You can consult the 2.2)

Return value:

0x00　　Command OK. (success)

0x01　　Command FAILURE

5.2　int　AntiType_B(

HANDLE　　commHandle,

```
                    int                 DeviceAddress,
                    unsigned char       *buffer);
```

Description: this order perform Anticol1B of ISO14443B

Input Parameter Description：
commHandle   the serial port handle
DeviceAddress   equipment address
*buffer,   The date after return card（ATQB）

Output Paramete:
  If Command success
    Buffer[0..N],   the date of card return（ATQB）
  If Command Failure
    buffer[0]   System Error/Status Codes(You can consult the 2.2)

Return value:
0x00   Command OK. (success)
0x01   Command FAILURE

```
5.3 int   SelectType_B (
                    HANDLE              commHandle,
                    int                 DeviceAddress,
                    unsigned char       *SerialNum);
```

Description:
    this order perform ATTRIB of ISO14443B, distribute a sign   for CID to the know card

Input Parameter Description：
commHandle   the serial port handle
DeviceAddress   equipment address
*SerialNum,   the serial number of the card

Return value:
0x00   Command OK. (success)
0x01   Command FAILURE

```
5.4 int    Request_AB(
                    HANDLE              commHandle,
                    int                 DeviceAddress,
                    unsigned char       * buffer);
```

Description:

This order performation integration REQUEST AND ATTRIB order of ISO14443B,use one order to make card reposition.

Input Parameter Description：

commHandle	the serial port handle
DeviceAddress	equipment address
* buffer,	return to 4 byte serial number of the worked card

Output Paramete:

If Command success

buffer[0..3], return to 4 byte serial number of the worked card

If Command Failure

buffer[0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

```
5.5 int   API_ISO14443TypeBTransCOSCmd(
                                        HANDLE          commHandle,
                                        int             DeviceAddress,
                                        unsigned char   *cmd,
                                        int             cmdSize,
                                        unsigned char   *buffer);
```

Description:

ISO14443 remit order, any effect order and date can be transfer by this order

Input Parameter Description：

commHandle	the serial port handle
DeviceAddress	equipment address
*cmd,	date which wait to be send
cmdSize,	date length
* buffer,	return date

Output Paramete:

If Command success

buffer[0..N] the date which returned from the card

If Command Failure

buffer[0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE

6 ISO15693 COMMANDS

6.1 int ISO15693_Inventory(

```
                    HANDLE              commHandle,
                    int                 deviceAddress,
                    unsigned char       flag,
                    unsigned char       afi,
                    const unsigned char *pData,
                    unsigned char       *nrOfCard,
                    unsigned char       *pBuffer);
```

Description:

This order is to get all the card's serial number which in the reading card district（the getable card number is relate to the output rate of the module antenna, commonly can read 2~6 card within anticollision）

Input Parameter Description：

commHandle	the serial port handle
DeviceAddress	equipment address
flag,	sign byte(length is 1 byte)
afi,	AFI is date length
*pData,	the sent date(the fact is a array, transfer by a pointer)
*nrOfCard	the return card number(length is 1 byte)
*pBuffer	the return date（include LAG, DSFID and 8*n byte card number）

Output Paramete:

IF：Command success (return 0x00)

*nrOfCard return card number (one byte)

*pBuffer return date（include FLAG, DSFID and 8*n byte card number）

IF：Command Failure (return 0x01)

*nrOfCard	System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is 2.2

6.2 int API_ISO15693Read (

```
HANDLE          commHandle,
int             DeviceAddress,
unsigned char   flags,
unsigned char   blk_add,
unsigned char   num_blk,
unsigned char   *uid,
unsigned char   *buffer);
```

Description:

It's for reading 1 and numerous fan value. If need to read the safe digit of every block, make Option flag of FLAGS as 1 ,means FLAG=0x42,every fan will return 5 byte, include one byte show safe state and 4 byte block content, here the most can read 12 block, IF FLAG=02,will only return 4 byte block content, here the most can read 63 block.

Input Parameter Description：

commHandle	the serial port handle
DeviceAddress	equipment address
flags	0x02 without uid
	0x22 with uid
	0x42 without uid but need to read the safe digit
blk_add,	needed reading origin block number
num_blk,	block quantity
*uid	UID message
*buffer	return value

Output Paramete:

If : Command success (return 0x00)

*buffer	return date
buffer[0]	return flag
buffer[1..N]	Data

If : Command Failure (return 0x01)

buffer[0]	System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is 2.2

6.3 int API_ISO15693Write(

```
                    HANDLE          commHandle,
                    int             DeviceAddress,
                    unsigned char   flags,
                    unsigned char   blk_add,
                    unsigned char   num_blk,
                    unsigned char   *uid,
                    unsigned char   *data);
```

Description: make writting work to every block(every time only can write one block)

Input Parameter Description：

commHandle	the serial port handle	
DeviceAddress	equipment address	
flags	0x02	without uid
	0x22	with uid
	0x42	without uid but need to read the safe digit
blk_add,	needed writting origin block number	
num_blk,	writing block quantity	
*uid	UID message	
*buffer	return value	

Output Paramete:

If : Command success (return 0x00)

*buffer	return date
buffer[0]	return flag
buffer[1..N]	Data

If : Command Failure (return 0x01)

buffer[0]	System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:2.2

6.4 int API_ISO15693Lock(

```
                    HANDLE          commHandle,
                    int             DeviceAddress,
                    unsigned char   flags,
                    unsigned char   num_blk,
                    unsigned char   *uid
                    unsigned char   *buffer);
```

Description: for lock block content, caution : this process can not in reverse(can not unlock) the content can not revise when the block be locked.

Input Parameter Description：

commHandle	the serial port handle	
DeviceAddress	equipment address	
flags	0x02	without uid
	0x42	without uid but need to read the safe digit
	0x22	with uid
num_blk,	locked block number	
*uid	UID message	
*buffer	return value	

Output Paramete:

If : Command success (return 0x00)，

buffer[0]　　return 0x80 ， means work ok.,

If : Command Failure (return 0x01)

buffer[0]　　System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:2.2

6.5 int　ISO15693StayQuiet(

```
                    HANDLE          commHandle,
                    int             DeviceAddress,
                    unsigned char   flags,
                    unsigned char   *uid,
                    unsigned char   *buffer );
```

Description:

This order is to make the card to sleep sate, must use the address mode if the sent date is same to the card serial number, after this work finish, the card will be sleep, otherwise the state will not change.

Input Parameter Description:

commHandle	the serial port handle
DeviceAddress	equipment address
flags	sign byte 1byte
*uid	UID message
*buffer	return value

Output Paramete:

If : Command success (return 0x00)

buffer[0] return 0x80，means work ok.

If : Command Failure (return 0x01)

buffer[0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:2.2

6.6 int ISO15693Select(

```
HANDLE          commHandle,
int             DeviceAddress,
unsigned char   flags,
unsigned char   *uid,
unsigned char   *buffer ) ;
```

Description:

This order must use address mode, if the sent date is same to the card serial number, after the work ok, the card will be selected, otherwise the state will not change

Input Parameter Description:

commHandle	the serial port handle
DeviceAddress	equipment address
flags	sign byte 1 byte
*uid	UID message
*buffer	return value

Output Paramete:

If : Command success (return 0x00)

buffer[0]	return to 0x80，means work ok

If : Command Failure (return 0x01)

buffer[0]	System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is 2.2

```
6.7 int   ResetToReady(
                              HANDLE           commHandle,
                              int              DeviceAddress,
                              unsigned char    flags,
                              unsigned char    *uid,
                              unsigned char    *buffer );
```

Description:

After the work ok, the card will return to Ready state .

Input Parameter Description:

commHandle	the serial port handle	
DeviceAddress	equipment address	
flags	sign byte 1 byte	
	0x02	without uid
	0x42	without uid but need to read safe digit
	0x22	with uid
*uid	UID message	
*buffer	return value	

Output Paramete:

If : Command success (return 0x00)

buffer[0]	return 0x80，means work ok

If : Command Failure (return 0x01)

buffer[0]	System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)

0x01	Command FAILURE, the error code is:

6.8 int WriteAFI(

```
                    HANDLE          commHandle,
                    int             DeviceAddress,
                    unsigned char   flags,
                    unsigned char   afi,
                    unsigned char   *uid,
                    unsigned char   *buffer );
```

Description:

Write AFI to the card .

Input Parameter Description：

commHandle the serial port handle

DeviceAddress equipment address

flags sign byt 1 byte

0x02 without uid

0x42 without uid but need to read the safe digit

0x22 with uid

afi wait write AFI

*uid UID message

*buffer return value

Output Paramete:

If : Command success (return 0x00)

buffer[0] return 0x80，means work ok

If : Command Failure (return 0x01)

buffer[0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:2.2

6.9 int LockAFI(

```
                    HANDLE          commHandle,
                        int                 DeviceAddress,
                        unsigned char       flags,
                        unsigned char       *uid,
                        unsigned char       *buffer );
```

Description:

For lock AFI of the card, after lock AFI can not change

Input Parameter Description：

commHandle	the serial port handle
DeviceAddress	equipment address
flags	sign byte 1 byte
	0x02 without uid
	0x42 without uid but need to read the safe digit
	0x22 with uid
*uid	UID message
*buffer	return value

Output Paramete:

If : Command success (return 0x00)，

buffer[0]	return 0x80，means work ok

If : Command Failure (return 0x01)

buffer[0]	System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:2.2

6.10 int WriteDSFID(

```
                        HANDLE          commHandle,
                        int             DeviceAddress,
                        unsigned char   flags,
                        unsigned char   DSFID,
                        unsigned char   *uid,
                        unsigned char   *buffer );
```

Description:

Write DSFID to the card

Input Parameter Description：

commHandle	the serial port handle

DeviceAddress   equipment address

flags   sign byte 1 byte
0x02   without   uid
0x42   without   uid but need to read the safe digit
0x22   with   uid

DSFID   the writted DSFID byte, the length is 1 byte
*uid   UID message
*buffer   return value

Output Paramete:
If : Command success (return 0x00)，
buffer[0]   return 0x80，means work ok
If : Command Failure (return 0x01)
buffer[0]   System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:

6.11

```
int    LockDSFID(
                    HANDLE           commHandle,
                    int              DeviceAddress,
                    unsigned char    flags,
                    unsigned char    *uid,
                    unsigned char    *buffer );
```

Description:

For lock DSFID of the card, after lock, DSFID can not change

Input Parameter Description：
commHandle   the serial port handle
DeviceAddress   equipment address

flags   sign   byte (length is 1 byte)
0x02   without   uid
0x42   without uid
0x22   with uid

*uid	UID message
*buffer	return value

Output Paramete:

If : Command success (return 0x00)，

buffer[0]	return 0x80，means work ok

If : Command Failure (return 0x01)

buffer[0]	System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:2.2

6.12 int ISO15693_GetSysInfo(

```
                    HANDLE          commHandle,
                    int             deviceAddress,
                    unsigned char   flag,
                    unsigned char   *uid,
                    unsigned char   *Buffer);
```

Description:

For get the particular message of the card, the fact content please refer to ISO15693 agreement date.

Input Parameter Description：

commHandle	the serial port handle
DeviceAddress	equipment address

flags	sign byte 1 byte	
	0x02	without uid
	0x42	withou uid but need to read the safe digit
	0x22	with uid

*uid	UID message
*buffer	return value

Output Paramete:

If : Command success (return 0x00)，

Then Buffer [0]: Flags
Buffer[1]: INFO Flags
Buffer[2..9]: UID
Buffer[10]: DSFID
Buffer[11]: AFI
Buffer[12..N]: Other fields

If : Command Failure (return 0x01)
Buffer[0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:2.2

6.13 int ISO15693_GetMulSecurity(

```
HANDLE              commHandle,
int                 deviceAddress,
unsigned char       flag,
unsigned char       blkAddr,
unsigned char       blkNum,
const unsigned char *uid,
unsigned char       *pBuffer);
```

Description: for get the date of every safe state block of the card

Input Parameter Description：

commHandle	the serial port handle
DeviceAddress	equipment address
flag	0x02 without uid
	0x22 with uid
	0x42 without uid but need to read the safe digit
blkAddr,	the read origin block number
blkNum,	the read block quantity
*uid	UID message
*pBuffer	return value

Output Paramete:
If : Command success (return 0x00)，

*pBuffer return value

pBuffer [0] return flags pBuffer [1..N] Block security status (the safe state of block)

If : Command Failure (return 0x01)

pBuffer[0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:2.2

6.14 int API_ISO15693TransCOSCmd(

```
HANDLE          commHandle,
int             DeviceAddress,
unsigned char   *cmd,
int             cmdSize,
unsigned char   *buffer);
```

Description: a currency order, the user can use this order to make kinds of handle to the card

Input Parameter Description:

commHandle	the serial port handle
DeviceAddress	equipment address
*cmd,	the date need to be send.
cmdSize,	the date length
*buffer	return value

Output Paramete:

If : Command success (return 0x00)，

*buffer return data

buffer [0..N] the date return from the card

If : Command Failure (return 0x01)

buffer[0] System Error/Status Codes(You can consult the 2.2)

Return value:

0x00	Command OK. (success)
0x01	Command FAILURE, the error code is:2.2