User Manual of DLL Function

Index

[A. DLL instruction 1](#_Toc4280)

[B. “FLock.dll” Description 1](#_Toc25968)

[1、 Write Guest card function 1](#_Toc11309)

[2、 Read Guest Card function 3](#_Toc25209)

[3、 Disable guest card function 4](#_Toc21629)

[4、 Write energy saver information function 5](#_Toc25191)

[5、 Read energy saver information function 6](#_Toc25785)

[6、 Write data to other sector(0-2 block) function 7](#_Toc4548)

[7、 Read data to other sector(0-2 block) function 7](#_Toc1557)

[8、 Read card serial number(UID) 8](#_Toc12408)

[9、 Encoder buzzer function 9](#_Toc28215)

[10、 Get guest card number function 9](#_Toc17779)

[11、 Conversion of a Binary String into a Hexadecimal String 10](#_Toc30919)

[12、 Conversion of a Hexadecimal String into a Binary String 10](#_Toc13408)

[13、 Send mobile key function 10](#_Toc16066)

[14、 Delete mobile key function 11](#_Toc28235)

[C. Code of Function Error 11](#_Toc8473)

# DLL instruction

* 1. “FLock.dll” is the main DLL of this interface；
  2. “umf.dll” is used for call Encoder’s DLL.
  3. “libeay32.dll”、“ssleay32.dll” are used for calling mobile key’s DLL.
  4. The suitable card of this DLL interface is MF1 S50.

# “FLock.dll” Description

## Write Guest card function

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Check-in new guest card | | |
| Function Explain | Delphi 7.0 | function Write\_Card(Port, ReaderType, SectorNo: Byte; HotelPwd: PChar; CardNo, GuestIdx, IsMoreLock, CanOpenDoubleLock: Integer; Dt, DoorID, DoorID1, DoorID2, SuitDoor, BeginTime, EndTime, LiftData: PChar): Integer; stdcall; far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Write\_Card(int Port, int ReaderType, int SectorNo, [MarshalAs(UnmanagedType.AnsiBStr)]string HotelPwd,  int CardNo, int GuestIdx, int IsMoreLock, int CanOpenDoubleLock, [MarshalAs(UnmanagedType.AnsiBStr)]string Dt, [MarshalAs(UnmanagedType.AnsiBStr)]string DoorID, [MarshalAs(UnmanagedType.AnsiBStr)]string DoorID1, [MarshalAs(UnmanagedType.AnsiBStr)]string DoorID2, [MarshalAs(UnmanagedType.AnsiBStr)]string SuitDoor, [MarshalAs(UnmanagedType.AnsiBStr)]string BeginTime, [MarshalAs(UnmanagedType.AnsiBStr)]string EndTime,  [MarshalAs(UnmanagedType.AnsiBStr)]string LiftData); | |
| Visual Basic  6.0 | Declare Function Write\_Card Lib "FLock.dll" (ByVal Port As Byte, ByVal ReaderType As Byte, ByVal SectorNo As Byte,ByVal HotelPwd As String,ByVal CardNo As Long,ByVal GuestIdx As Long,ByVal IsMoreLock As Byte,ByVal CanOpenDoubleLock As Byte,ByVal Dt As String,ByVal DoorID As String, ByVal DoorID1 As String,ByVal DoorID2 As String,ByVal SuitDoor As String, ByVal BeginTime As String, ByVal EndTime As String, ByVal LiftData As String) As Integer | |
| PowerBuilder  8.0 | function int Write\_Card(int Port, int ReaderType, int SectorNo, char HotelPwd[8], long CardNo, int GuestIdx, int IsMoreLock, int CanOpenDoubleLock, char Dt[4], char DoorID[8], char DoorID1[8], char DoorID2[8], char SuitDoor[6], char BeginTime[10], char EndTime[10], char EndTime[32]) library "FLock.dll" | |
| Function Explain | Name | String Name | Explain |
| Port | Port number | If the encoder is serial port, fill in value. 1~4 represent COM1, COM2, COM3 and COM4 respectively. If the encoder is USB port, the port value is not effective. |
| ReaderType | Encoder type | 1：S9 |
| SectorNo | Sector number | Default ：1（Range from 0 to 13） |
| HotelPwd | Hotel SID | The value is 6-digit character string which with the content of ACSⅡ |
| CardNo | Guest card No. | Guest serial number is generated in according with computer time. Convert the time of computer to 4 bits integer serial number via“SerialNo\_FromNow”function. |
| GuestIdx | Guest index | Guest Index（Range from 1 to 65535） |
| IsMoreLock | More room | 0：No；  1：Yes； |
| CanOpenDoubleLock | Double lock | 0：No；  1：Yes； |
| Dt | Duplicate and new card | 00：Duplicate card；  AA：New card；  Remark：CardNo must be over or equal to the current guest card number |
| DoorID | Door ID | Door ID is 6 digits, please check the software door ID |
| DoorID1 | Door ID1 | Door ID is 6 digits, please check the software door ID, it will be activated when the value of CanOpenDoubleLock is 1 。 |
| DoorID2 | Door ID2 | Door ID is 6 digits, please check the software door ID, it will be activated when the value of CanOpenDoubleLock is 1 。 |
| SuitDoor | SuitDoor | Inner door No. of a suite room is the code string of inner doors that can be opened by a guest card in a suite room. Its length is 4 and the default value is “0000”. A guest card can open up to 16 inner doors and 1 outer door of a suite room. If the first six digits of the lock codes of two or more rooms in a door lock system are identical, it indicates that suite rooms are set in the door lock system. If no suite room is set in the lock system, or it is not necessary to open inner doors (all guest cards can open external doors of suite rooms), it can be filled inner default value “0000”. If a guest card can open No.01, No.02, No. 07 and No.08 inner doors of a suite room, the binary string will be “0000 0000 1100 0011”, which can be converted into a hexadecimal-string: “00C3” |
| BeginTime | Check in time | Effective begin time, the length is 10, the time format is “\_ \_ (year) \_ \_ (month) \_ \_ (date) \_ \_(hour)\_ \_ (minute)” |
| EndTime | Check out time | Effective begin time, the length is 10, the time format is “\_ \_ (year) \_ \_ (month) \_ \_ (date) \_ \_(hour)\_ \_ (minute)” |
| LiftData | Elevator control |  |

## Read Guest Card function

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Read guest card information | | |
| Function Explain | Delphi 7.0 | function Read\_Card(Port, ReaderType, SectorNo: Byte; HotelPwd: PChar; var CardNo, GuestIdx, IsMoreLock, CanOpenDoubleLock: Integer;Dt, DoorID, DoorID1, DoorID2, SuitDoor, BeginTime, EndTime, LiftData: PChar): Integer; stdcall; far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Read\_Card(int Port, int ReaderType, int SectorNo, [MarshalAs(UnmanagedType.AnsiBStr)]string HotelPwd, ref int CardNo, ref int GuestIdx, ref int IsMoreLock, ref int CanOpenDoubleLock, StringBuilder Dt, StringBuilder DoorID, StringBuilder DoorID1, StringBuilder DoorID2, StringBuilder SuitDoor, StringBuilder BeginTime, StringBuilder EndTime, StringBuilder LiftData); | |
| Visual Basic  6.0 | Declare Function Read\_Card Lib "FLock.dll" (ByVal Port As Byte, ByVal ReaderType As Byte, ByVal SectorNo As Byte, ByVal HotelPwd As String, CardNo As Long, GuestIdx As Long, IsMoreLock As Byte, CanOpenDoubleLock As Byte, ByVal Dt As String, ByVal DoorID As String, ByVal DoorID1 As String, ByVal DoorID2 As String, ByVal SuitDoor As String, ByVal BeginTime As String, ByVal EndTime As String, ByVal LiftData As String) As Integer | |
| PowerBuilder  8.0 | function int Read\_Card(int Port, int ReaderType, int SectorNo, char HotelPwd[8], ref long CardNo, ref int GuestIdx, ref int IsMoreLock, ref int CanOpenDoubleLock, ref char Dt[4], ref char DoorID[8], ref char DoorID1[8], ref char DoorID2[8], ref char SuitDoor[6], ref char BeginTime[10], ref char EndTime[10], ref char EndTime[32]) library "FLock.dll" | |
| Function Explain | Name | String Name | Explain |
| Port | Port number | If the encoder is serial port, fill in value. 1~4 represent COM1, COM2, COM3 and COM4 respectively. If the encoder is USB port, the port value is not effective. |
| ReaderType | Encoder type | 1：S9 |
| SectorNo | Sector number | Default ：1（Range from 0 to 13） |
| HotelPwd | Hotel SID | The value is 6-digit character string which with the content of ACSⅡ |
| CardNo | Guest card No. | Guest serial number is generated in according with computer time. Convert the time of computer to 4 bits integer serial number via“SerialNo\_FromNow”function. |
| GuestIdx | Guest index | Guest Index（Range from 1 to 65535） |
| IsMoreLock | More room | 0：No；  1：Yes； |
| CanOpenDoubleLock | Double lock | 0：No；  1：Yes； |
| Dt | Duplicate and new card | 00：Duplicate card；  AA：New card；  Remark：CardNo must be over or equal to the current guest card number |
| DoorID | Door ID | Door ID is 6 digits, please check the software door ID |
| DoorID1 | Door ID1 | Door ID is 6 digits, please check the software door ID, it will be activated when the value of CanOpenDoubleLock is 1 。 |
| DoorID2 | Door ID2 | Door ID is 6 digits, please check the software door ID, it will be activated when the value of CanOpenDoubleLock is 1 。 |
| SuitDoor | SuitDoor | Inner door No. of a suite room is the code string of inner doors that can be opened by a guest card in a suite room. Its length is 4 and the default value is “0000”. A guest card can open up to 16 inner doors and 1 outer door of a suite room. If the first six digits of the lock codes of two or more rooms in a door lock system are identical, it indicates that suite rooms are set in the door lock system. If no suite room is set in the lock system, or it is not necessary to open inner doors (all guest cards can open external doors of suite rooms), it can be filled inner default value “0000”. If a guest card can open No.01, No.02, No. 07 and No.08 inner doors of a suite room, the binary string will be “0000 0000 1100 0011”, which can be converted into a hexadecimal-string: “00C3” |
| BeginTime | Check in time | Effective begin time, the length is 10, the time format is “\_ \_ (year) \_ \_ (month) \_ \_ (date) \_ \_(hour)\_ \_ (minute)” |
| EndTime | Check out time | Effective begin time, the length is 10, the time format is “\_ \_ (year) \_ \_ (month) \_ \_ (date) \_ \_(hour)\_ \_ (minute)” |
| LiftData | Elevator control |  |

## Disable guest card function

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Disable guest card information | | |
| Function Explain | Delphi 7.0 | function Disabled\_Card(Port, ReaderType, SectorNo: Byte; HotelPwd: PChar): Integer; stdcall; far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Disabled\_Card(int Port, int ReaderType, int SectorNo, [MarshalAs(UnmanagedType.AnsiBStr)]string HotelPwd); | |
| Visual Basic  6.0 | Declare Function Disabled\_Card Lib"FLock.dll" (ByVal Port As Byte, ByVal ReaderType As Byte, ByVal SectorNo As Byte, ByVal HotelPwd As String) As Integer | |
| PowerBuilder  8.0 | function int Disabled\_Card(int Port, int ReaderType, int SectorNo, char HotelPwd[8]) library "FLock.dll" | |
| Function Explain | Name | String Name | Explain |
| Port | Port number | If the encoder is serial port, fill in value. 1~4 represent COM1, COM2, COM3 and COM4 respectively. If the encoder is USB port, the port value is not effective. |
| ReaderType | Encoder type | 1：S9 |
| SectorNo | Sector number | Default ：1（Range from 0 to 13） |
| HotelPwd | Hotel SID | The value is 6-digit character string which with the content of ACSⅡ |

## Write energy saver information function

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Write guest card information | | |
| Function Explain | Delphi 7.0 | function Write\_PowerSwitch(Port, ReaderType, SectorNo: Byte; PowerSwitchPwd, DoorID, EndTime: PChar): Integer; stdcall; far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Write\_PowerSwitch(int Port, int ReaderType, int SectorNo, [MarshalAs(UnmanagedType.AnsiBStr)]string PowerSwitchPwd, [MarshalAs(UnmanagedType.AnsiBStr)]string DoorID, [MarshalAs(UnmanagedType.AnsiBStr)]string EndTime); | |
| Visual Basic  6.0 | Declare Function Write\_PowerSwitch Lib"FLock.dll" (ByVal Port As Byte, ByVal ReaderType As Byte, ByVal SectorNo As Byte, ByVal PowerSwitchPwd As String, ByVal DoorID As String, ByVal EndTime As String) As Integer | |
| PowerBuilder  8.0 | function int Write\_PowerSwitch(int Port, int ReaderType, int SectorNo, char PowerSwitchPwd[14], char DoorID[8], char EndTime[12]) library "FLock.dll" | |
| Function Explain | Name | String Name | Explain |
| Port | Port number | If the encoder is serial port, fill in value. 1~4 represent COM1, COM2, COM3 and COM4 respectively. If the encoder is USB port, the port value is not effective. |
| ReaderType | Encoder type | 1：S9 |
| SectorNo | Sector number | 15 |
| PowerSwitchPwd | PowerSwitchPwd | The value is setting in according with power switch settings. (Default value is FF FF FF FF FF FF ),The value is 12-digit character string which with the content of hexadecimal |
| DoorID | Door ID | Door ID is 6 digits, please check the software door ID |
| EndTime | Check out time | Effective begin time, the length is 10, the time format is “\_ \_ (year) \_ \_ (month) \_ \_ (date) \_ \_(hour)\_ \_ (minute)” |

## Read energy saver information function

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Read guest card information | | |
| Function Explain | Delphi 7.0 | function Read\_PowerSwitch(Port, ReaderType, SectorNo: Byte; PowerSwitchPwd, DoorID, EndTime: PChar): Integer; stdcall;far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Read\_PowerSwitch(int Port, int ReaderType, int SectorNo, [MarshalAs(UnmanagedType.AnsiBStr)]string PowerSwitchPwd, StringBuilder DoorID, StringBuilder EndTime); | |
| Visual Basic  6.0 | Declare Function Read\_PowerSwitch Lib"FLock.dll" (ByVal Port As Byte, ByVal ReaderType As Byte, ByVal SectorNo As Byte, ByVal PowerSwitchPwd As String,ByVal DoorID As String, ByVal EndTime As String) As Integer | |
| PowerBuilder  8.0 | function int Write\_PowerSwitch(int Port, int ReaderType, int SectorNo, char PowerSwitchPwd[14], char DoorID[8], char EndTime[12]) library "FLock.dll" | |
| Function Explain | Name | String Name | Explain |
| Port | Port number | If the encoder is serial port, fill in value. 1~4 represent COM1, COM2, COM3 and COM4 respectively. If the encoder is USB port, the port value is not effective. |
| ReaderType | Encoder type | 1：S9 |
| SectorNo | Sector number | 15 |
| PowerSwitchPwd | PowerSwitchPwd | The value is setting in according with power switch settings. (Default value is FF FF FF FF FF FF ),The value is 12-digit character string which with the content of hexadecimal |
| DoorID | Door ID | Door ID is 6 digits, please check the software door ID |
| EndTime | Check out time | Effective begin time, the length is 10, the time format is “\_ \_ (year) \_ \_ (month) \_ \_ (date) \_ \_(hour)\_ \_ (minute)” |

## Write data to other sector(0-2 block) function

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Write data to other sector(0-2 block) | | |
| Function Explain | Delphi 7.0 | function Write\_Block(Port, ReaderType, SectorNo, BlockNo: Byte; SectorPwd: PChar; Data: PChar): Integer; stdcall; far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Read\_Block(int Port, int ReaderType, int SectorNo, int BlockNo, [MarshalAs(UnmanagedType.AnsiBStr)]string PowerSwitchPwd, StringBuilder Data); | |
| Visual Basic  6.0 | Declare Function Write\_Block Lib"FLock.dll" (ByVal Port As Byte, ByVal ReaderType As Byte, ByVal SectorNo As Byte, ByVal BlockNo As Byte, ByVal SectorPwd As String, ByVal Data As String) As Integer | |
| PowerBuilder  8.0 | function int Write\_Block(int Port, int ReaderType, int SectorNo, int BlockNo, char SectorPwd[14], char Data[98]) library "FLock.dll" | |
| Function Explain | Name | String Name | Explain |
| Port | Port number | If the encoder is serial port, fill in value. 1~4 represent COM1, COM2, COM3 and COM4 respectively. If the encoder is USB port, the port value is not effective. |
| ReaderType | Encoder type | 1：S9 |
| SectorNo | Sector number | Default ：1（Range from 0 to 13） |
| BlockNo | Block Number | Range 0--2 |
| SectorPwd | Sector password | 12 bit string with hexadecimal code characters. |
| Data | Write data | 12 bit string with hexadecimal code characters. |

## Read data to other sector(0-2 block) function

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Read data to other sector(0-2 block) | | |
| Function Explain | Delphi 7.0 | function Read\_Block(Port, ReaderType, SectorNo, BlockNo: Byte; SectorPwd: PChar; Data: PChar): Integer; stdcall; far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Write\_Block(int Port, int ReaderType, int SectorNo, int BlockNo, [MarshalAs(UnmanagedType.AnsiBStr)]string PowerSwitchPwd, [MarshalAs(UnmanagedType.AnsiBStr)]string Data); | |
| Visual Basic  6.0 | Declare Function Read\_Block Lib"FLock.dll" (ByVal Port As Byte, ByVal ReaderType As Byte, ByVal SectorNo As Byte, ByVal BlockNo As Byte, ByVal SectorPwd As String, ByVal Data As String) As Integer | |
| PowerBuilder  8.0 | function int Read\_Block(int Port, int ReaderType, int SectorNo, int BlockNo, char SectorPwd[14], ref char Data[98]) library "FLock.dll" | |
| Function Explain | Name | String Name | Explain |
| Port | Port number | If the encoder is serial port, fill in value. 1~4 represent COM1, COM2, COM3 and COM4 respectively. If the encoder is USB port, the port value is not effective. |
| ReaderType | Encoder type | 1：S9 |
| SectorNo | Sector number | Default ：1（Range from 0 to 13） |
| BlockNo | Block Number | Range 0--2 |
| SectorPwd | Sector password | 12 bit string with hexadecimal code characters. |
| Data | Write data | 12 bit string with hexadecimal code characters. |

## Read card serial number(UID)

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Read card serial number(UID) | | |
| Function Explain | Delphi 7.0 | function Read\_Snr(Port, ReaderType: Byte; Snr: PChar): Integer; stdcall; far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Read\_Snr(int Port, int ReaderType, StringBuilder Snr); | |
| Visual Basic  6.0 | Declare Function Read\_Snr Lib"FLock.dll" (ByVal Port As Byte, ByVal ReaderType As Byte, ByVal Snr As String) As Integer | |
| PowerBuilder  8.0 | function int Read\_Snr(int Port, int ReaderType, ref char Snr[10]) library "'FLock.dll" | |
| Function Explain | Name | String Name | Explain |
| Port | Port number | If the encoder is serial port, fill in value. 1~4 represent COM1, COM2, COM3 and COM4 respectively. If the encoder is USB port, the port value is not effective. |
| ReaderType | Encoder type | 1：S9 |
| Snr | Card serial number | Card’s UID |

## Encoder buzzer function

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Encoder buzzer function | | |
| Function Explain | Delphi 7.0 | function Reader\_Alarm(Port, ReaderType, AlarmCount: Byte): Integer; stdcall; far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Reader\_Alarm(int Port, int ReaderType, int AlarmCount); | |
| Visual Basic  6.0 | Declare Function Reader\_Alarm Lib"FLock.dll" (ByVal Port As Byte, ByVal ReaderType As Byte, ByVal AlarmCount As Byte) As Integer | |
| PowerBuilder  8.0 | function int Reader\_Alarm(int Port, int ReaderType, int AlarmCount) library"FLock.dll" | |
| Function Explain | Name | String Name | Explain |
| Port | Port number | If the encoder is serial port, fill in value. 1~4 represent COM1, COM2, COM3 and COM4 respectively. If the encoder is USB port, the port value is not effective. |
| ReaderType | Encoder type | 1：S9 |
| AlarmCount | Buzzer times |  |

## Get guest card number function

|  |  |  |
| --- | --- | --- |
| Function Explain | Convert the time of computer to 4 bits integer serial number via“SerialNo\_FromNow”function****. The value must be positive number.**** | |
| Function Explain | Delphi 7.0 | function SerialNo\_FromNow():Integer; stdcall; far; external 'FLock.dll'; |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int SerialNo\_FromNow(); |
| Visual Basic  6.0 | Declare Function SerialNo\_FromNow Lib"FLock.dll" ( ) As Long |
| Function Explain | Parameter return value: > 0 | |

## Conversion of a Binary String into a Hexadecimal String

|  |  |  |
| --- | --- | --- |
| Function Explain | Conversion of a Binary String into a Hexadecimal String | |
| Function Explain | Delphi 7.0 | function Bin\_Hex(Dest: PChar; Source: PChar; Len: Integer): Integer; stdcall; far; external 'FLock.dll'; |
| Function Explain | Name | Explain |
| Dest | Destination hexadecimal character string. |
| Source | Source binary character string. |
| Len | Length of the source character string. |
| Parameter return value | 0 indicates successful conversion, -1 indicates conversion failure. |

## Conversion of a Hexadecimal String into a Binary String

|  |  |  |
| --- | --- | --- |
| Function Explain | Conversion of a Hexadecimal String into a Binary String | |
| Function Explain | Delphi 7.0 | function Hex\_Bin(Dest: PChar; Source: PChar; Len: Integer): Integer; stdcall; far; external 'FLock.dll'; |
| Function Explain | Name | Explain |
| Dest | Destination binary character string. |
| Source | Source hexadecimal character string. |
| Len | Length of the source hexadecimal character string. |
| Parameter return value | 0 indicates successful conversion, -1 indicates conversion failure. |

## Send mobile key function

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Send mobile key | | |
| Function Explain | Delphi 7.0 | function Send\_Key(AUserName, APassword, ABleId, ARecUserName, AKeyName, AStartDate, AEndDate: PChar; var AKeyId: Integer): Integer; stdcall; far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Send\_Key([MarshalAs(UnmanagedType.AnsiBStr)]string AUserName, [MarshalAs(UnmanagedType.AnsiBStr)]string APassword, [MarshalAs(UnmanagedType.AnsiBStr)]string ABleId, [MarshalAs(UnmanagedType.AnsiBStr)]string ARecUserName, [MarshalAs(UnmanagedType.AnsiBStr)]string AKeyName, [MarshalAs(UnmanagedType.AnsiBStr)]string AStartDate, [MarshalAs(UnmanagedType.AnsiBStr)]string AEndDate, out int AKeyId); | |
| Function Explain | Name | String Name | Explain |
| AUserName | Login Account | Create account |
| APassword | Login password | Create login password |
| ABleId | BLE ID | BLE ID |
| ARecUserName | Account to receive mobile key | Account to receive mobile key |
| AKeyName | Mobile key name |  |
| AStartDate | Check in time | Format：YYYY-MM-DD HH:MM:SS |
| AEndDate | Check out time | Format：YYYY-MM-DD HH:MM:SS |
| AKeyId | mobile key ID |  |

## Delete mobile key function

|  |  |  |  |
| --- | --- | --- | --- |
| Function Explain | Delete mobile key | | |
| Function Explain | Delphi 7.0 | function Delete\_Key(AUserName, APassword: PChar; AKeyId: Integer): Integer; stdcall; far; external 'FLock.dll'; | |
| C# | [DllImport("FLock.dll", CharSet = CharSet.Ansi, CallingConvention = CallingConvention.StdCall)]  public static extern int Delete\_Key([MarshalAs(UnmanagedType.AnsiBStr)]string AUserName, [MarshalAs(UnmanagedType.AnsiBStr)]string APassword, int AKeyId); | |
| Function Explain | Name | String Name | Explain |
| AUserName | Login Account | Create account |
| APassword | Login password | Create login password |
| AKeyId | mobile key ID |  |

# Code of Function Error

|  |  |
| --- | --- |
| Return value | Explain |
| 0 | Success |
| 255 | Failure |
| 1 | Open serial com port error |
| 2 | No card error |
| 3 | Card type error |
| 4 | Read card error |
| 5 | Hotel password (system ID) error |
| 6 | Write card error |
| 7 | It is not a guest card |
| 8 | Parameter error |
| 10 | The card is incorrect system ID. |
| 10003 | Token not existed |
| 10004 | Token no authorization ，token invalid or cancel authorization |
| 10007 | Username or password error |
| 20001 | The user does not belong to this mobile key |
| 20002 | Not the user of the lock |
| 20003 | The correspondence between key ID, lock ID and user is wrong |
| 20004 | Mobile key is not existed |
| 20006 | Receiver is not existed. |
|  |  |
|  |  |