

Print Electronics Embeded Device Programmer's Guide

BioLife Embeded Device Programmer's Guide.....	1
1.Introduction	3
2. Method.....	4
2.1 GetEnrollData	4
2.2 SetEnrollData.....	5
2.3 GetUserTmp	5
2.4 SetUserTmp.....	6
2.5 DeleteEnrollData	6
2.6 DelUserTmp	6
2.7 ReadGeneralLogData	6
2.8 GetGeneralLogData.....	7
2.9 ReadAllGLogData.....	8
2.10 GetAllGLogData.....	8
2.11 GetDeviceStatus.....	8
2.12 GetDeviceInfo	9
2.13 SetDeviceInfo.....	10
2.14 EnableDevice	12
2.15 EnableUser	12
2.16 SetDeviceTime	13
2.17 GetDeviceTime.....	13
2.19 PowerOffDevice	14
2.20 ModifyPrivilege	14
2.21 ReadAllUserID.....	15
2.22 GetAllUserID	15
2.23 GetSerialNumber.....	16
2.24 ClearKeeperData	16
2.25 ClearAdministrators.....	16
2.26 GetFirmwareVersion.....	17
2.27 GetSDKVersion.....	17
2.28 ClearGLog.....	17
2.29 GetFPTempLength.....	17
2.30 Connect_Com.....	17
2.31 Connect_Net	17
2.32 Disconnect	18
2.33 SetUserInfo.....	18
2.34 GetUserInfo	18
2.35 SetDeviceIP	19
2.36 GetDeviceIP.....	19
2.37 FPTempConvert	19
2.38 GetSerialNumber	19
2.39 SetCommPassword	19
2.40 GetSMS.....	19
2.41 SetSMS	20
2.42 CaptureImage.....	20
2.43 UpdateFirmware.....	20

2.44 BackupData.....	21
2.45 RestoreData.....	21
2.46 WriteLCD.....	21
2.47 ClearLCD.....	22
2.48 Beep	22
2.49 PlayVoiceByIndex.....	22
2.50 ACUnlock	23
2.51 GetUserGroup.....	23
2.52 SetUserGroup.....	24
2.53 GetTZInfo	24
2.54 SetTZInfo	24
2.55 GetUnlockGroups.....	25
2.56 SetUnlockGroups.....	25
2.57 GetGroupTZStr.....	26
2.58 SetGroupTZStr.....	26
2.59 GetUserTZStr	26
2.60 SetUserTZStr.....	27
2.61 GetLastError.....	27

1. Introduction

Printsdk is the software development kit for Print embedded fingerprint machine. It can develop communication programs with embedded machines. For example, you can get enrolled user information from a device, or set user information to a device. The information may contain user name, user password, user privilege, or user fingerprint template, and you can manage devices on a computer. You can also get In/Out logs from a device.

Installation:

Copy all DLL files to the system folder or the current folder where the program runs. Then run regsvr32.exe to register the controls: Regsvr32 zkenkeeper.dll.

Note:

Functions of zkenkeeper that are not in this document are not supported.

Parameter *dwEMachineNumber* equals *dwMachineNumber*, *dwEMachineNumber* only for compatibility with old versions.

2. Method

2.1 GetEnrollData

[Function]

Download the enrolled user's Fingerprint template from device

[Protocol]

```
    BOOL    GetEnrollData(  
                long        dwMachineNumber ,  
                long        dwEnrollNumber ,  
                long        dwEMachineNumber ,  
                long        dwBackupNumber ,  
                long FAR*    dwMachinePrivilege ,  
                long FAR*    dwEnrollData ,  
                long FAR*    dwPassWord);
```

[Parameters]

dwMachineNumber

The Machine Number of operating device

dwEMachineNumber

equal to *dwMachineNumber*

dwEnrollNumber

The User ID

dwBackupNumber

The Backup Number of the User, Backup Number means

Value	Description
0	Fingerprint Template 0
1	Fingerprint Template 1
2	Fingerprint Template 2
3	Fingerprint Template 3
4	Fingerprint Template 4
5	Fingerprint Template 5
6	Fingerprint Template 6
7	Fingerprint Template 7
8	Fingerprint Template 8
9	Fingerprint Template 9
10	Password

dwMachinePrivilege

The user's privilege at the device, as following:

Value	Description	
0	General user	
1	Administrator (user enrollment, set options of the device)	[Level 1]
2	Administrator (user enrollment,)	[Level 2]
3	Administrator (set options of the device)	[Level 3]

dwEnrollData

The pointer of buffer for save fingerprint template, buffer size must great or equal than 1836 BYTES.

dwPassWord

The password.

[Return]

TRUE if success, FALSE else.

[Note]

If dwBackupNumber is between 0 and 9, the specified fingerprint template saved to a buffer, which dwEnrollData parameter pointed to. And parameter dwPassWord is ignored. On the other hand, if dwBackupNumberValue is 10, the password be saved to dwPassWord, and dwEnrollData parameter is ignored.

2.2 SetEnrollData

[Function]

Upload a specified user Fingerprint template to a device, Overwrite the exists user's data.

[Protocol]

```

    BOOL    SetEnrollData(
                long        dwMachineNumber ,
                long        dwEnrollNumber ,
                long        dwEMachineNumber ,
                long        dwBackupNumber ,
                long        dwMachinePrivilege ,
                long FAR*   dwEnrollData ,
                long        dwPassWord);

```

[Parameters]**dwMachineNumber**

The Machine Number of operating device

dwEnrollNumber

The User ID

dwEMachineNumber

equal to *dwMachineNumber*

dwBackupNumber

see Method [GetEnrollData](#)

dwMachinePrivilege

The user's privilege for this device

dwEnrollData

The fingerprint template. If dwBackupNumber is 10 it be ignored

dwPassWord

The user's password. If dwBackupNumber less than 10 it be ignored.

[Return]

TRUE if success, FALSE else.

2.3 GetUserTmp

Download the enrolled user's Fingerprint template from device.It's speed faster than GetEnrollData function

2.4 SetUserTmp

Upload a specified user Fingerprint template to a device, Overwrite the exists user's data.it upload the fingerprint to the device directly,so the device may have the same fingerprints,it is different from SetEnrollData.the speed faster than SetEnrollData

2.5 DeleteEnrollData

[Function]

Delete Fingerprint template of enrolled user.

[Protocol]

```
BOOL DeleteEnrollData(  
    long dwMachineNumber ,  
    long dwEnrollNumber ,  
    long dwEMachineNumber ,  
    long dwBackupNumber );
```

[Paramters]

See [GetEnrollData](#)

[Return]

TRUE if success, FALSE else.

[Notes]

If there is no specified user in the device, return TRUE.

If dwBackupNumberValue is 11, all of fingerprint template of the user, who specified by dwEnrollNumber, will be deleted.

If dwBackupNumberValue is 12, all of fingerprint template and password of the user, who specified by dwEnrollNumber, will be deleted.

2.6 DelUserTmp

Only Delete a Fingerprint template of enrolled user.

2.7 ReadGeneralLogData

[Function]

Read general transication LOGs(attendance data) into internal buffer from device.

[Protocol]

```
BOOL ReadGeneralLogData(  
    long dwMachineNumber );
```

[Paramters]

dwMachineNumber

The Machine Number of operating device

[Return]

TRUE if success, FALSE else.

[Notes]

After call ReadGeneralLogData to read data to buffer, you can call GetGeneralLogData to get every record of log.

2.8 GetGeneralLogData

[Function]

Get a record of log from internal buffer, which filled by ReadGeneralLogData.

[Protocol]

```
BOOL GetGeneralLogData(  
    long dwMachineNumber ,  
    long FAR* dwTMachineNumber ,  
    long FAR* dwEnrollNumber ,  
    long FAR* dwEMachineNumber ,  
    long FAR* dwVerifyMode ,  
    long FAR* dwInOutMode ,  
    long FAR* dwYear ,  
    long FAR* dwMonth ,  
    long FAR* dwDay ,  
    long FAR* dwHour ,  
    long FAR* dwMinute );
```

[Parameters]

dwMachineNumber

The Machine Number of operating device

dwTMachineNumber

The Machine Number of device, where verified the user.

dwEnrollNumber

The User ID.

dwEMachineNumber

equal to *dwMachineNumber*

dwVerifyMode

The method of a successful verification. It's as following:

Value	Description
1	Fingerprint
2	Password

dwInOutMode

The attendance status. It's as following:

Value	Description
0	Duty On
1	Duty Off
2	Overtime Begin
3	Overtime End
4	Lock Out
5	Lock In

dwYear,
dwMonth,
dwDay,

dwHour,
dwMinute

The different parts of the date and time.

[Return]

TRUE if success, FALSE else.

[Notes]

Call ReadGeneralLogData to read data into internal buffer before this method. You can call it repeatedly until it return FALSE for getting all of records.

See also Method [ReadGeneralLogData](#)

2.9 ReadAllGLogData

Same ReadGeneralLogData

2.10 GetAllGLogData

Same GetGeneralLogData

2.11 GetDeviceStatus

[Function]

Get device status.

[Protocol]

```
BOOLGetDeviceStatus(  
    long          dwMachineNumber ,  
    long          dwStatus ,  
    long FAR*    dwValue );
```

[Paramters]

dwMachineNumber

The Machine Number of operating device

dwStatus

Indicate which status. It's as following:

Value	Description
1	Count of administrators
2	Count od users
3	Count of fingerprint template
4	Count of password.
5	Count of new administration transication LOGs.
6	Count of new general LOGs

dwValue

The returned value.

[Return]

TRUE if success, FALSE else.

2.12 GetDeviceInfo

[Function]

Get the options of a device.

[Protocol]

```
BOOL GetDeviceInfo(  
                long          dwMachineNumber ,  
                long          dwInfo ,  
                long FAR*     dwValue );
```

[Parameters]

dwMachineNumber

The Machine Number of operating device

dwInfo

Indicate which option. It's as following:

1. Maximum number of administrator. The value is from 0-10.
2. Machine Number of the device. The value is from 1 to 255.
3. Language

Value	Description
0	English
1	Chinese
2	Korean
4. Auto shutdown device time(minutes). This value is from 0 to 255.
5. Lock Open output control

Value	Description
0	Output Open Lock signal
1	Not output Open Lock signal
6. Alarm number of general log. It's from 0 to 1500.
7. Alarm number of administration log. It's from 0 to 255.
8. Minimum interval(seconds) of two general log.
9. Baud rate of communication

Value	Description
0	1200 bps
1	2400 bps
2	4800 bps
3	9600 bps
4	19200 bps
5	38400 bps
10. Odd/Even Check

Value	Description
0	None
1	Even
2	Odd
11. Stop Bits

Value	Description
0	One Bit
1	Two Bits
12. Date Delimiter
Not support now, but you can implement it use device info item 34 named Date format
13. Network
14. RS232
15. RS485
16. Voice
17. Identification speed

- 18.idle
- 19.Shutdown time
- 20.PowerOn time
- 21.Sleep time
- 22.Auto Bell
- 23.Match threshold
- 24.Register threshold
- 25.1:1 threshold
- 26.Show score
- 27.Unlock person count
- 28.Only verify number card
- 29.Net Speed
- 30.Must registe card
- 31.Time out of temp state keep
- 32.Time out of input number
- 33.Time out of menu keep
- 34.Date formate
- 35.Only 1:1

dwValue

The returned value.

[Return]

TRUE if success, FALSE else.

2.13 SetDeviceInfo

[Function]

Set the options of a device.

[Protocol]

```

BOOLSetDeviceInfo(
                long           dwMachineNumber ,
                long           dwInfo ,
                long           dwValue );

```

[Paramters]

dwMachineNumber

Device ID

dwInfo

Device info index

- 1.Registrable administrators number
- 2.Device ID
- 3.Languages
- 4.Auto power off time
- 5.Lock control delay(20ms)
- 6.In and out record warning
- 7.manage record warning
- 8.Confirm interval time
- 9.Baud Rate
- 10.Even and Odd
- 11.Stop bit
- 12.Date list separator

Not support now,but you can implement it use device info item 34 named Date formate,pls refer to dwValue 34.

- 13.Network

- 14.RS232
- 15.RS485
- 16.Voice
- 17.Identification speed
- 18.idle
- 19.Shutdown time
- 20.PowerOn time
- 21.Sleep time
- 22.Auto Bell
- 23.Match threshold
- 24.Register threshold
- 25.1:1 threshold
- 26.Show score
- 27.Unlock person count
- 28.Only verify number card
- 29.Net Speed
- 30.Must registe card
- 31.Time out of temp state keep
- 32.Time out of input number
- 33.Time out of menu keep
- 34.Date formate
- 35.Only 1:1

dwValue

- 1.Registrable administrators number
- 2.Device ID
- 3.Languages

Value	Description
0	English,
1	Simplified Chinese
2	Traditional Chinese

- 4.Auto power off time
- 5.Lock control delay(20ms)
- 6.In and out record warning
- 7.manage record warning
- 8.Confirm interval time
- 9.Baud Rate
- 10.Even and Odd
- 11.Stop bit
- 12.Date list separator
- 13.Network
- 14.RS232
- 15.RS485
- 16.Voice
- 17.Identification speed
- 18.idle
- 19.Shutdown time
- 20.PowerOn time
- 21.Sleep time
- 22.Auto Bell
- 23.Match threshold
- 24.Register threshold
- 25.1:1 threshold
- 26.Show score

Value	Description
0	Whether show verify score, False
1	True.

- 27.Unlock person count
- 28.Only verify number card
- 29.Net Speed

30.Must registe card

Verify card that card must registred.

31.Time out of temp state keep

Keep time for temp state,for example OT-IN,OT-OUT....But CheckIn/CheckOut is normal state.

32.Time out of input number

Keep time for wait input number

33.Time out of menu keep

Menu keep time when no option on it.

34.Date formate

Value	Description
0	YY-MM-DD
1	YY/MM/DD
3	YY.MM.DD
4	MM-DD-YY
5	MM/DD/YY
6	MM.DD.YY
7	DD-MM-YY
8	DD/MM/YY
9	DD.MM.YY
10	YYYYMMDD

35.Only 1:1

Only 1:1 verify mode.0:False,1:true.

[Return]

TRUE if success, FALSE else.

2.14 EnableDevice

[Function]

Allow/not allow the front user operating the device.

[Protocol]

```
BOOL EnableDevice(  
                long          dwMachineNumber ,  
                BOOL         bFlag );
```

[Paramters]

dwMachineNumber

The Machine Number of operating device

bFlag

TRUE means allow user input keys and fingerprint, FALSE means not allow

[Return]

TRUE if success, FALSE else.

[Notes]

Please call EnableDevice with bFlag=FALSE to disable front user's operation before calling other functions to exchange data between host and device, such as ReadGenealLogData, GetEnrollData etc. And call EnableDevice with bFlag=TRUE after your transactions finish.

2.15 EnableUser

[Function]

Enable or disable a user.

[Protocol]

```
BOOL EnableUser(  
    long          dwMachineNumber ,  
    long          dwEnrollNumber ,  
    long          dwEMachineNumber ,  
    long          dwBackupNumber ,  
    BOOL         bFlag );
```

[Parameters]

bFlag

TRUE means enable the user, FALSE means disable him/her.

Other parameters see Method [GetEnrollData](#).

[Return]

TRUE if success, FALSE else.

2.16 SetDeviceTime

[Function]

Set the device date and time.

[Protocol]

```
BOOLSetDeviceTime(  
    long          dwMachineNumber );
```

[Parameters]

dwMachineNumber

The Machine Number of operating device

[Return]

TRUE if success, FALSE else.

[Notes]

This method set the device date and time same as host.

2.17 GetDeviceTime

[Function]

Get the device date and time.

[Protocol]

```
BOOLGetDeviceTime(  
    long          dwMachineNumber ,  
    long FAR*    dwYear ,  
    long FAR*    dwMonth ,  
    long FAR*    dwDay ,  
    long FAR*    dwHour ,
```

```
long FAR* dwMinute,  
long FAR* dwDayOfWeek);
```

[Parameters]

dwMachineNumber

The Machine Number of operating device

dwYear,

dwMonth,

dwDay,

dwHour,

dwMinute

The different parts of the device date.

dwDayOfWeek

The Day of Week of the device date

[Return]

TRUE if success, FALSE else.

2.19 PowerOffDevice

[Function]

Shut down the device.

[Protocol]

```
BOOL PowerOffDevice(  
    long dwMachineNumber );
```

[Parameters]

dwMachineNumber

The Machine Number of operating device

[Return]

TRUE if success, FALSE else.

2.20 ModifyPrivilege

[Function]

Modify a user privilege in a device.

[Protocol]

```
BOOL ModifyPrivilege(  
    long dwMachineNumber ,  
    long dwEnrollNumber ,  
    long dwEMachineNumber ,  
    long dwBackupNumber ,  
    long dwMachinePrivilege);
```

[Parameters]

See Method [GetEnrollData](#).

[Return]

TRUE if success, FALSE else.

2.21 ReadAllUserID

[Function]

Read all User IDs from device into internal buffer.

[Protocol]

```
BOOL ReadAllUserID (  
    long          dwMachineNumber );
```

[Parameters]**dwMachineNumber**

The Machine Number of operating device

[Return]

TRUE if success, FALSE else.

[Notes]

Call Method GetAllUserID to get every record of User ID.

2.22 GetAllUserID

[Function]

Read a record of user ID from buffer that ReadAllUserID filled.

[Protocol]

```
BOOL GetAllUserID (  
    long dwMachineNumber,  
    long FAR* dwEnrollNumber,  
    long FAR* dwEMachineNumber,  
    long FAR* dwBackupNumber,  
    long FAR* dwMachinePrivilege,  
    long FAR* dwEnable);
```

[Parameters]**dwEnable**

Indicate if the user is enabled. Value 1 means a normal(enabled) user, and value 2 means disabled user.

[Return]

TRUE if success, FALSE else.

[Notes]

Call GetAllUserID repeatedly until it return FALSE for get all of them.

2.23 GetSerialNumber

[Function]

Get the product information or serial number

[Protocol]

```
BOOLGetSerialNumber (
    long dwMachineNumber,
    BSTR FAR* lpszSerialNumber);
```

[Parameters]

dwMachineNumber

The Machine Number of operating device

lpszSerialNumber

The returned string

[Return]

TRUE if success, FALSE else.

2.24 ClearKeeperData

[Function]

Delete all enrolled user/administrator fingerprint templates name and In/out logs.

[Protocol]

```
BOOL ClearKeeperData(
    long dwMachineNumber);
```

[Parameters]

dwMachineNumber

The Machine Number of operating device

[Return]

TRUE if success, FALSE else.

2.25 ClearAdministrators

[Function]

Clear all administrator's privilege in the device.

[Protocol]

ClearAdministrators(dwMachineNumber As Long) As Boolean

[Parameters]

dwMachineNumber

The Machine Number of operating device

[Return]

TRUE if success, FALSE else.

2.26 GetFirmwareVersion

[Function]

Get firmware version of device.

[Protocol]

GetFirmwareVersion(dwMachineNumber As Long, strVersion As String) As Boolean

[Parameters]

dwMachineNumber

The Machine Number of operating device

strVersion

Firmware version.

[Return]

TRUE if success, FALSE else.

2.27 GetSDKVersion

Get zkemkeeper version.

2.28 ClearGLog

Clear all logs from device.

2.29 GetFPTempLength

Get fingerprint template length.

2.30 Connect_Com

[Function]

Connect to the device by com port.

[Protocol]

Connect_Com(ComPort As Long, MachineNumber As Long, BaudRate As Long) As Boolean

[Parameters]

ComPort

Com port number

MachineNumber

The Machine Number of operating device

BaudRate

Communication BaudRate

[Return]

TRUE if success, FALSE else.

2.31 Connect_Net

[Function]

Connect to the device by Ethernet

[Protocol]

Connect_Net(IPAdd As String, Port As Long) As Boolean

[Parameters]

IPAdd

IP address

Port

Port number, Default 4370

[Return]

TRUE if success, FALSE else.

2.32 Disconnect

Disconnect with device.

2.33 SetUserInfo

[Function]

Set user's information to device. user's information contain user's name and password.

[Protocol]

SetUserInfo(dwMachineNumber As Long, dwEnrollNumber As Long, Name As String, Password As String, Privilege As Long, Enabled As Boolean) As Boolean

[Parameters]

dwMachineNumber

The Machine Number of operating device

dwEnrollNumber

Enroll number you want to set

Name

User name

Password

User Password

Privilege

User privilege

Enabled

whether Enabled user

[Return]

TRUE if success, FALSE else.

2.34 GetUserInfo

[Function]

Get user's information from device. user's information contain user's name and password.

[Protocol]

SetUserInfo(dwMachineNumber As Long, dwEnrollNumber As Long, Name As String, Password As String, Privilege As Long, Enabled As Boolean) As Boolean

[Parameters]

dwMachineNumber

The Machine Number of operating device

dwEnrollNumber

Enroll number t

Name

User name

Password

User Password

Privilege

User privilege

Enabled

whether Enabled user

[Return]

TRUE if success, FALSE else.

2.35 SetDeviceIP

Set device IP address.

2.36 GetDeviceIP

Get device IP address.

2.37 FPTempConvert

Convert the fingerprint template of device to Biokey fingerprint template, then you can verify the fingerprint on computer by Biokey. Biokey SDK is BioLife pc online fingerprint development kit.

2.38 GetSerialNumber

Get serial number of device.

2.39 SetCommPassword

[Function]

Set communication Password to connection. the password can reject Unauthorized connection.

[Protocol]

BOOL SetCommPassword (long Commkey)

[Parameters]

Commkey

The communication password. for example: if the password is "123456" then CommKey=123456

[Return]

TRUE if success, FALSE else.

[Notes]

2.40 GetSMS

[Function]

Get SMS

[Protocol]

GetSMS(dwIndex As Long, dwEnrollNumber As Long, SMS As String) As Boolean

[Parameters]

dwIndex

Index of SMS

dwEnrollNumber

the User ID of the SMS

SMS

Value of the SMS

[Return]

TRUE if success, FALSE else.

[Notes]

2.41 SetSMS

[Function]

Set SMS

[Protocol]

SetSMS(dwIndex As Long, dwEnrollNumber As Long, SMS As String) As Boolean

[Parameters]

dwIndex

Index of SMS

dwEnrollNumber

specify the User ID of the SMS

SMS

Value of the SMS

[Return]

TRUE if success, FALSE else.

[Notes]

2.42 CaptureImage

[Function]

Capture current image of fingerprint scanner

[Protocol]

CaptureImage(FullImage As Boolean, Width As Long, Height As Long, Image As Byte, ImageFile As String) As Boolean

[Parameters]

FullImage

If true, the function will return Full image, else, it will return valid image, the valid image is the image for verify.

Width

specify width of the image

Height

specify the height of the image

Image

Byte array of image

ImageFile

Image File

[Return]

TRUE if success, FALSE else.

[Notes]

2.43 UpdateFirmware

[Function]

Update firmware

[Protocol]

UpdateFirmware(FirmwareFile As String) As Boolean

[Parameters]

Firmware file
firmware File

[Return]

TRUE if success, FALSE else.

[Notes]**2.44 BackupData****[Function]**

Backup user data of device, contain user ID, password, fingerprint, log, etc.

[Protocol]

BackupData(DataFile As String) As Boolean

[Parameters]

DataFile
Backup data File

[Return]

TRUE if success, FALSE else.

[Notes]**2.45 RestoreData****[Function]**

Restore data that backup before.

[Protocol]

RestoreData(DataFile As String) As Boolean

[Parameters]

Data file
Backup data file

[Return]

TRUE if success, FALSE else.

[Notes]**2.46 WriteLCD**

[Function]

Write LCD of Devie

[Protocol]

WriteLCD(Row As Long, Col As Long, Text As String) As Boolean

[Paramters]

Row

Col

Text

[Return]

TRUE if success, FALSE else.

[Notes]**2.47 ClearLCD****[Function]**

Clear LCD of Devie

[Protocol]

ClearLCD() As Boolean

[Paramters]**[Return]**

TRUE if success, FALSE else.

[Notes]**2.48 Beep****[Function]**

Beep of buzzer

[Protocol]

Beep(DelayMS As Long) As Boolean

[Paramters]

DelayMS

Delay millisecond

[Return]

TRUE if success, FALSE else.

[Notes]**2.49 PlayVoiceByIndex****[Function]**

Play voice

[Protocol]

PlayVoiceByIndex(Index As Long) As Boolean

[Parameters]

Index

Index of voice.

[Return]

TRUE if success, FALSE else.

[Notes]

2.50 ACUnlock

[Function]

send a signal for unlock.

[Protocol]

ACUnlock(dwMachineNumber As Long, Delay As Long) As Boolean

[Parameters]

dwMachineNumber

Device ID.

[Return]

TRUE if success, FALSE else.

[Notes]

2.51 GetUserGroup

[Function]

GetUserGroup

[Protocol]

GetUserGroup(dwMachineNumber As Long, dwEnrollNumber As Long, UserGrp As Long) As Boolean

[Parameters]

dwMachineNumber

Device ID

dwEnrollNumber

User ID

UserGrp

Group number of the user

[Return]

TRUE if success, FALSE else.

[Notes]

2.52 SetUserGroup

[Function]

Set group for the user..

[Protocol]

SetUserGroup(dwMachineNumber As Long, dwEnrollNumber As Long, UserGrp As Long) As Boolean

[Parameters]

dwMachineNumber

Device ID.

dwEnrollNumber

User ID

UserGrp

User group

[Return]

TRUE if success, FALSE else.

[Notes]

2.53 GetTZInfo

[Function]

Get timezone information.

[Protocol]

GetTZInfo(dwMachineNumber As Long, TZIndex As Long, TZ As String) As Boolean

[Parameters]

dwMachineNumber

Device ID.

TZIndex

Timezone index

TZ

Timezone string.

[Return]

TRUE if success, FALSE else.

[Notes]

2.54 SetTZInfo

[Function]

Set timezone information

[Protocol]

SetTZInfo(dwMachineNumber As Long, TZIndex As Long, TZ As String) As Boolean

[Parameters]

dwMachineNumber
Device ID.
TZIndex
Index of TimeZone
TZ
Timezone string

[Return]

TRUE if success, FALSE else.

[Notes]**2.55 GetUnlockGroups****[Function]**

Get unlock groups

[Protocol]

GetUnlockGroups(dwMachineNumber As Long, Grps As String) As Boolean

[Parameters]

dwMachineNumber
Device ID.
Grps
Groups

[Return]

TRUE if success, FALSE else.

[Notes]**2.56 SetUnlockGroups****[Function]**

Set unlock groups

[Protocol]

SetUnlockGroups(dwMachineNumber As Long, Grps As String) As Boolean

[Parameters]

dwMachineNumber
Device ID.
Grps
Groups string

[Return]

TRUE if success, FALSE else.

[Notes]

2.57 GetGroupTZStr

[Function]

Get Group timezones.

[Protocol]

GetGroupTZStr(dwMachineNumber As Long, GroupIndex As Long, TZs As String) As Boolean

[Parameters]

dwMachineNumber
Device ID.
GroupIndex
TZs
Timezones

[Return]

TRUE if success, FALSE else.

[Notes]

2.58 SetGroupTZStr

[Function]

Set group timezone.

[Protocol]

SetGroupTZStr(dwMachineNumber As Long, GroupIndex As Long, TZs As String) As Boolean

[Parameters]

dwMachineNumber
Device ID.
GroupIndex
TZs

[Return]

TRUE if success, FALSE else.

[Notes]

2.59 GetUserTZStr

[Function]

Get timezone of the user.

[Protocol]

GetUserTZStr(dwMachineNumber As Long, dwEnrollNumber As Long, TZs As String) As Boolean

[Parameters]

dwMachineNumber
Device ID.
dwEnrollNumber

User ID
TZs
Timezones

[Return]

TRUE if success, FALSE else.

[Notes]

2.60 SetUserTZStr

[Function]

Set user timezones

[Protocol]

SetUserTZStr(dwMachineNumber As Long, dwEnrollNumber As Long, TZs As String) As Boolean

[Parameters]

dwMachineNumber
Device ID.
dwEnrollNumber
User ID
TZs
Timezones

[Return]

TRUE if success, FALSE else.

[Notes]

2.61 GetLastError

[Function]

Get last error number.

[Protocol]

```
BOOL GetLastError(  
                long FAR*           dwErrorCode);
```

[Parameters]

dwErrorCode

The returned error number

Value	Description
1	SUCCEEDED
4	ERR_INVALID_PARAM
0	ERR_NO_DATA
-1	ERROR_NOT_INIT
-2	ERROR_IO
-3	ERROR_SIZE
-4	ERROR_NO_SPACE
-100	ERROR_UNSUPPORTED

[Return]

TRUE if success, FALSE else.

[Notes]

dwErrorCode = 6 caused by calling Method [GetGeneralLogData](#), and [GetAllGLogData](#).
If them return FALSE, please call GetLastError to determine if there is error or read over data.