

# 脱机系列通讯开发包开发手册

版本: 6.1

---

日期: 2008 年2月

2006 ZKSoftware Inc.。版权所有，翻印必究。

- 本说明文件中的信息如有更改，恕不另行通知。
- ZKSoftware Inc.为中控科技的司标，Biokey 为中控科技的核心技术徽标，ZKSoftware ,Biokey 已经在中国及美国注册。
- 本说明文件中述及其他商标和产品名称是指拥有相应商标和产品名称的公司或者制造的产品。ZKSoftware Inc.对其他公司的商标和产品名称不拥有任何所有权。
- 开发中使用问题，非开发包问题，请自行解决。
- 使用该开发包，由于人为或程序本身造成数据丢失，概不负责。

# 目 录

1. 脱机开发包描述 .....	1
2. SDK 安装.....	2
2.1 文件夹目录 .....	2
2.2 如何安装.....	2
3. 快速开始 .....	2
3.1 用户基本数据上传和下载及管理常用函数 .....	2
3.1.1 考勤记录下载 .....	2
1 ReadAllGLogData .....	3
2 GetGeneralLogDataStr.....	3
3 ClearGLog .....	3
3.1.2 用户数据的增、删、改、查.....	3
1 ReadAllUserID .....	3
2 ReadAllTemplate .....	3
3 SetUserInfo .....	3
4 SetUserTmp .....	3
5 CardNumber 属性.....	3
6 DeleteEnrollData.....	4
7 DelUserTmp.....	4
8 GetUserInfo .....	4
9 GetUserTmp.....	4
10 ModifyPrivilege .....	4
11 GetAllUserID .....	4
3.1.3 管理记录下载 .....	4
1 ReadAllSLogData .....	4
2 GetAllSLogData .....	4
3 ClearSLog .....	4
3.2 门禁操作常用函数 .....	4
1 GetTZInfo .....	5
2 SetTZInfo.....	5
3 SetUserTZStr .....	5
4 GetUserTZStr.....	5
5 SetUserGroup.....	5
6 GetUserGroup.....	5
7 SetGroupTZStr.....	5
8 GetGroupTZStr .....	5
9 SetUnlockGroups .....	5
10 GetUnlockGroups .....	5
11 ACUnlock .....	5
3.3 设备操作常用函数 .....	5

1	Beep .....	6
2	ClearLCD .....	6
3	Connect_Com .....	6
4	EnableDevice .....	6
5	GetDeviceInfo .....	6
6	RestartDevice .....	6
7	SetWiegandDefine .....	6
3.4	常用事件 .....	6
1	OnAttTransaction .....	6
2	OnAlarm .....	6
3	OnKeyPress .....	6
4	RegEvent .....	7
3.5	数据下载一般流程（考勤记录、用户信息、指纹） .....	7
4.	函数详细描述 .....	9
4.1	数据管理 .....	9
4.1.1	用户信息 .....	9
4.1.1.1	GetUserInfo .....	9
4.1.1.2	SetUserInfo .....	9
4.1.1.3	ClearAdministrators .....	10
4.1.1.4	DeleteEnrollData .....	10
4.1.1.5	GetUserInfoByCard .....	11
4.1.1.6	BeginBatchUpdate .....	12
4.1.1.7	BatchUpdate .....	13
4.1.1.8	ReadAllUserID .....	13
4.1.1.9	GetUserInfoByPIN2 .....	13
4.1.1.10	GetPIN2 .....	14
4.1.1.11	GetUserIDByPIN2 .....	15
4.1.1.12	GetSMS .....	15
4.1.1.13	SetSMS .....	16
4.1.1.18	GetAllUserInfo .....	17
4.1.1.19	GetAllUserID .....	17
4.1.1.20	GetBackupNumber .....	18
4.1.1.21	ModifyPrivilege .....	18
4.1.1.22	DeleteSMS .....	18
4.1.1.23	SetUserSMS .....	19
4.1.1.24	DeleteUserSMS .....	19
4.1.1.25	ClearSMS .....	20
4.1.1.26	ClearUserSMS .....	20
4.1.1.27	SetUserInfoEx .....	20
4.1.1.28	GetUserInfoEx .....	21
4.1.1.29	GetStrCardNumber .....	21
4.1.1.30	SetStrCardNumber .....	22
4.1.2	指纹管理 .....	23
4.1.2.1	GetUserTmpStr .....	23
4.1.2.2	SetUserTmpStr .....	23

4.1.2.3 DelUserTmp.....	24
4.1.2.4 GetEnrollDataStr.....	24
4.1.2.5 SetEnrollDataStr .....	25
4.1.2.6 SetUserTmp .....	26
4.1.2.7 GetUserTmp.....	26
4.1.2.8 GetEnrollData .....	27
4.1.2.9 SetEnrollData.....	28
4.1.2.10 ReadAllTemplate .....	28
4.1.2.11 FPTempConvertNew .....	29
4.1.2.12 FPTempConvertNewStr .....	29
4.1.2.13 FPTempConvertStr.....	30
4.1.2.14 FPTempConvert .....	31
4.1.2.15 GetFPTempLengthStr .....	31
4.1.2.16 GetFPTempLength.....	32
4.1.3 记录管理 .....	32
4.1.3.1 ReadSuperLogData.....	32
4.1.3.2 ReadGeneralLogData.....	32
4.1.3.3 GetGeneralLogDataStr.....	33
4.1.3.4 GetSuperLogData .....	34
4.1.3.5 GetAllSLogData .....	35
4.1.3.6 GetAllGLogData.....	36
4.1.3.7 ReadAllSLogData.....	37
4.1.3.8 ReadAllGLogData .....	37
4.1.3.9 ClearSLog .....	37
4.1.3.10 GetGeneralExtLogData.....	37
4.1.4 系统数据管理 .....	38
4.1.4.1 BackupData.....	38
4.1.4.2 RestoreData.....	38
4.1.4.3 ClearKeeperData.....	39
4.1.4.4 PINWidth .....	39
4.1.4.5 RefreshData .....	39
4.2 门禁.....	40
4.2.1 ACUnlock .....	40
4.2.2 EnableUser.....	40
4.2.3 GetUserTZs.....	41
4.2.4 SetUserTZs .....	41
4.2.5 GetUserGroup.....	42
4.2.6 SetUserGroup.....	42
4.2.7 GetACFun.....	43
4.2.8 GetTZInfo .....	43
4.2.9 SetTZInfo.....	44
4.2.10 GetUnlockGroups .....	44
4.2.11 SetUnlockGroups .....	44
4.2.12 GetGroupTZs.....	45
4.2.13 SetGroupTZStr.....	45

4.2.14 GetGroupTZStr .....	46
4.2.15 SetGroupTZStr.....	46
4.2.16 GetUserTZStr.....	47
4.2.17 SetUserTZStr .....	47
4.2.18 GetDoorState.....	48
4.2.19 UseGroupTimeZone.....	48
4.3 机器设置.....	49
4.3.1 Beep .....	49
4.3.2 ClearLCD.....	49
4.3.3 Connect_Com .....	49
4.3.4 Connect_Net .....	50
4.3.5 DisableDeviceWithTimeOut.....	51
4.3.6 Disconnect .....	51
4.3.7 EnableDevice .....	51
4.3.8 EnableClock.....	52
4.3.9 GetDeviceStatus.....	52
4.3.10 GetDeviceInfo.....	53
4.3.11 GetWiegandDefine.....	54
4.3.12 GetDeviceIP.....	55
4.3.13 SetDeviceIP .....	55
4.3.14 GetDeviceMAC .....	55
4.3.15 GetDeviceTime .....	56
4.3.16 GetSerialNumber .....	56
4.3.17 GetProductCode.....	57
4.3.18 GetFirmwareVersion .....	57
4.3.19 GetSDKVersion .....	57
4.3.20 PowerOnAllDevice .....	58
4.3.21 PowerOffDevice .....	58
4.3.22 PlayVoiceByIndex.....	58
4.3.23 QueryState.....	59
4.3.24 RestartDevice.....	59
4.3.25 SetDeviceInfo .....	59
4.3.26 SetDeviceTime.....	60
4.3.27 SetDeviceTime2.....	60
4.3.28 SetDeviceMAC .....	61
4.3.29 SetWiegandDefine .....	61
4.3.30 SetCommPassword .....	62
4.3.31 UpdateFirmware .....	62
4.3.32 WriteLCD.....	62
4.4 其他.....	63
4.4.1 CaptureImage .....	63
4.4.2 CancelOperation .....	64
4.4.3 GetLastError .....	64
4.4.4 StartVerify .....	64
4.4.5 StartEnroll .....	65

4.4.6 StartIdentify .....	65
4.4.7 GetSensorSN .....	65
4.4.8 WriteCard .....	66
4.4.9 EmptyCard .....	67
<b>5. 事件 .....</b>	<b>67</b>
5.1 OnAttTransaction .....	67
5.2 OnFinger .....	67
5.3 OnNewUser .....	67
5.4 OnEnrollFinger .....	67
5.5 OnKeyPress .....	68
5.6 OnVerify .....	68
4.7 OnFingerFeature .....	68
5.8 OnAlarm .....	68
5.9 OnHIDNum .....	68
5.10 OnWriteCard .....	69
5.11 OnEmptyCard .....	69
5.12 OnAttTransactionEx .....	69
5.13 ReadRTLog .....	69
5.14 GetRTLog .....	70
5.15 OnDoor .....	70
<b>6. 属性 .....</b>	<b>70</b>
6.1 AccGroup .....	70
6.2 AccTimeZones .....	70
6.3 BASE64 .....	70
6.4 CardNumber .....	71
6.5 CommPort .....	71
6.6 ConvertBIG5 .....	71
6.7 PINWidth .....	71
<b>7. 常见问题解答 .....</b>	<b>71</b>
7.1 如何下载考勤记录 .....	71
7.2 如何联机创建用户 .....	71
7.3 U 盘数据导入及下载 .....	71
7.4 BIOKEY 采集指纹模板写入脱机 .....	72
7.5 取得所有用户所有信息 .....	72
7.6 机器连接 .....	72
7.7 在 SetUserInfo 后密码不能用问题 .....	73
7.8 联机模板转换为脱机模板 .....	73
7.9 Demo 程序不能接机器问题 .....	73
7.10 脱机指纹机连接后处于工作状态问题 .....	73
7.11 将繁体文字写入脱机指机的时候需要注意的问题 .....	73
7.12 A5,K8 射频管理卡问题 .....	74
7.13 连接穿越防火墙或路由器问题 .....	74
7.14 关于指纹模板 .....	74

7.15 大容量指纹上传 .....	74
<b>8. 扩展函数 .....</b>	<b>75</b>
8.1 SSR_GetGeneralLogData .....	75
8.2 SSR_GetAllUserInfo .....	75
8.3 SSR_GetUserInfo .....	77
8.4 SSR_SetUserInfo .....	77
8.5 SSR_GetUserTmpStr .....	78
8.6 SSR_DeleteEnrollData .....	78
8.7 SSR_GetUserTmp .....	78
8.8 SSR_DelUserTmp .....	78
8.9 SSR_SetUserTmpStr.....	79
8.10 SSR_SetUserTmp .....	80
8.11 SetWorkCode .....	80
8.12 GetWorkCode .....	80
8.13 DeleteWorkCode.....	81
8.14 ClearWorkCode.....	81
8.15 IsTFTMachine .....	81
8.15 SSR_EnableUser.....	82
8.16 SSR_SetUserSMS.....	82
8.17 SSR_DeleteUserSMS .....	83
8.18 SSR_SetHoliday .....	83
8.19 SSR_GetHoliday.....	84
8.20 SSR_SetGroupTZ .....	84
8.21 SSR_GetGroupTZ .....	85
8.22 SSR_SetUnLockGroup .....	85
8.23 SSR_GetUnLockGroup .....	86
8.24SetDaylight .....	86
8.25GetDaylight.....	87



## Release history

Date	Revision	Change
2006.12	5.10	First release
2007.07	6.0	Add some new function
2008.02	6.1	Add RegEvent

# 1. 脱机开发包描述

脱机通讯开发包是一个与脱机指纹机进行数据通讯的接口。可以方便地对脱机指纹脱机指纹机进行用户信息、指纹的管理，下载打卡记录，对指纹机进行各种设置及门禁配置管理的开发包。主要功能有：

- 1、下载指纹机打卡记录。
- 2、下载、上传用户信息、卡信息及其指纹。
- 3、设置指纹门禁机的规则。
- 4、进行各种配置，如设置脱机指纹机时间，设置匹配阈值等的功能。
- 5、可实时激发脱机指纹机的各种事件，如按指纹、验证通过等。
- 6、可设置用户“短消息”（只能在标示有此功能的机种使用）。

## 2. SDK 安装

### 2.1 文件夹目录

- 1、 Demo
- 2、 zkemsdk
- 3、 脱机系列通讯开发包开发手册.pdf

### 2.2 如何安装

将所有 dll 文件拷贝到系统目录 System32 下，在开始一〉运行中执行 `regsvr32 zkemkeeper.dll`，提示注册成功，在开发工具中引入该控件(请参考具体工具使用方法)，简单拖放，即可使用。

#### 说明

以下都采用 C 语法说明函数协议，vb 语法举例说明函数功能。函数以功能分类。举例中 CZKEM1 控件名称。machineNumber 为机器号。SDK 默认为脱机通信开发包。

## 3. 快速开始

所有接口函数执行前，请首先连接设备，如果通过网络连接，请使用 `connect_net`(具体使用请查阅该函数说明)。如果使用串口或者 485 连接，请使用 `connect_com` 连接设备。所有连接参数如设备 IP、机器号、波特率可以在设备菜单中设置。

连接过程不需要具体协议如何使用，开发包中已经封装完善，调用控件接口即可。

**注意：**网络连接时，端口为 4370。

### 3.1 用户基本数据上传和下载及管理常用函数

#### 3.1.1 考勤记录下载

执行流程：首先使用 `ReadAllGLogData` 读取所有考勤记录到内存中（内存管理由开发包自动完成），然后，执行 `GetGeneralLogData` 函数读取每一条考勤记录。当 `GetGeneralLogData` 函数返回 `False` 的时候，读完所有考勤记录。在该过程中使用到的函数请参考下列函数，函数具体使用可参考函数具体使用部分说明。在操作完成后，执行 `DisConnect` 函数断开连接的设备，开发包自动释放所占用的资源。

## 1 ReadAllGLogData

读取所有考勤记录到内存中，为 GetGeneralLogData 等函数获取考勤记录做准备。和 ReadAllGLogData 函数功能相同。

## 2 GetGeneralLogDataStr

从内存中获取考勤记录，和函数 GetGeneralLogData 功能完全相同，只是返回时间参数格式有所不同。

## 3 ClearGLog

清除所有考勤记录。

注意：所有逐条获取考勤记录的函数，在返回 False 的时候，标示读完考勤记录。具体函数使用，请参考函数接口详细手册。在有 WorkCode 功能的机器中，可以用 GetGeneralExtLogData 函数从内存中读到考勤记录。

### 3.1.2 用户数据的增、删、改、查

用户增加暨用户上传，删除包含删除指纹、密码、用户。修改指修改用户名称、权限、密码等，指纹不能被修改。查询暨用户下载。

一般下载用户流程：和下载考勤记录相同，首先执行 ReadAllUserID、ReadAllTemplate 函数读取用户所有信息到内存中来（注意：即使读取所有用户信息，容量相对 PC 内存很小）。然后可以执行相关获取(get)函数，如 GetUserInfo、GetUserTmp 等，获取用户信息。

#### 1 ReadAllUserID

将所有用户信息（编号、姓名、密码不包含指纹）读取到内存中。

#### 2 ReadAllTemplate

将用户所有指纹模板读取到内存中。

#### 3 SetUserInfo

增加或修改用户信息，如果用户存在，则覆盖，否则创建。建议使用该函数。

#### 4 SetUserTmp

增加(上传)用户指纹模板。其它相同功能函数有：SetUserTmp、SetEnrollData、SetEnrollDataStr，请查阅其他章节描述。

## 5 CardNumber 属性

该属性用户上传或下载某一用户卡信息。

先设置 cardnumber[0]=要创建用户的卡号，然后用 setuserinfo,就能写入用户卡号信息。下载时在 getuserinfo 函数后，获取该属性，既可获得用户卡号。

请阅读：FAQ A5,K8 射频管理卡问题

## 6 DeleteEnrollData

删除用户本身或用户密码或指纹。

## 7 DelUserTmp

删除用户指纹模板。

## 8 GetUserInfo

按照编号查询用户信息，包含密码、姓名等。

## 9 GetUserTmp

查询(下载或获得)用户指纹模板，该函数指纹模板以字符串返回，可参考其它函数，如：GetEnrollDataStr、GetUserTmp、GetEnrollData 都可以获得指纹模板，很多函数为兼容存在，其内部实现相同。

## 10 ModifyPrivilege

修改用户权限。

## 11 GetAllUserID

从内存中获取某一用户信息，该函数每执行一次，指向内存用户记录下移到下一条记录，该函数返回 False 时，下载万所有用户信息，可根据该函数返回用户编号，下载指纹模板。配合 ReadAllUserID、ReadAllTemplate、GetUserTmp 函数，可以下载所有用户信息。

### 3.1.3 管理记录下载

下载管理记录过程和下载考勤记录过程完全相同，都是先读到内存中，然后获得每条记录。

#### 1 ReadAllSLogData

读所有管理记录到内存，为获取管理记录做准备。相同功能函数有 ReadSuperLogData。

#### 2 GetAllSLogData

从内存中获取管理记录。和 GetSuperLogData 函数功能相同。

#### 3 ClearSLog

清除管理记录。

## 3.2 门禁操作常用函数

主要包括：时间段、组使用时间段、用户使用时间段的上传、下载，开锁组合等。门禁机内置 50 个时间段，5 个组，10 个开锁组合。如不了解细节，请参考函数部分门禁函数说明。

## 1 GetTZInfo

下载时间段的设置。

## 2 SetTZInfo

上传时间段设置。

## 3 SetUserTZStr

上传用户使用的时间段。

## 4 GetUserTZStr

下载用户使用时间段。

## 5 SetUserGroup

设置用户属于某一个组。

## 6 GetUserGroup

查询用户所属于的组。

## 7 SetGroupTZStr

上传组使用时间段。

## 8 GetGroupTZStr

下载组使用时间段。

## 9 SetUnlockGroups

设置开锁组合，共 5 个组，可以上传 10 个组合。

## 10 GetUnlockGroups

返回指纹门禁机的开锁组合。

## 11 ACUnlock

开门命令。

## 3.3 设备操作常用函数

设备操作主要包含：机器参数，如时间、通信参数、电源管理等设置。

## 1 Beep

响铃。

## 2 ClearLCD

清空 LCD 显示内容，准备写入要写入的字符。WriteLCD 可以在 LCD 写入要显示的字符。

## 3 Connect\_Com

连接机器，Connect\_Net 函数用户网络连接机器。

## 4 EnableDevice

使机器处于屏蔽外围设备状态或取消屏蔽。EnableClock 用于停止秒表的闪烁。类似函数有 DisableDeviceWithTimeOut，该函数可以屏蔽外围设备且有超时限制。使用 EnableDevice 函数，程序员可以按照自己的程序自己来控制状态。

是否允许指纹机打卡，当有较大数据通讯时使用该函数不允许指纹机打卡可提高通讯的速度和效率

## 5 GetDeviceInfo

获取设备相关配置参数。功能类似函数有：GetDeviceStatus、SetDeviceInfo。还有其他函数，如修改机器时间函数 SetDeviceTime、SetDeviceTime2。修改机器 IP 函数 SetDeviceIP。GetSerialNumber 函数用户获取设备序列号，可以参阅开发手册中设备管理函数。一般情况下，设备参数有下载就有上传，函数名称也由 Get....改为 Set....，函数功能对称，当然有的参数修改不了，如设备名称、序列号等。

## 6 RestartDevice

重新启动设备，相关函数有 PowerOffDevice，关闭机器电源。

## 7 SetWiegandDefine

设置门禁机违根输出格式。对应 Get 函数也可以获取到该格式。

# 3.4 常用事件

## 1 OnAttTransaction

当验证指纹通过时，触发该事件。监控程序可以很容易实时了解到出入情况。一般称该事件为实时事件。

## 2 OnAlarm

实时报警事件，可以监控到设备报警状态。

## 3 OnKeyPress

当按键盘时，触发该事件，可以监控到按键情况。

## 4 RegEvent

当连接成功后，调用该函数，激活实时事件。比如激活 OnAttTransaction、OnAlarm 事件。

RegEvent (LONG dwMachineNumber, LONG EventMask, VARIANT\_BOOL\* pVal)函数，

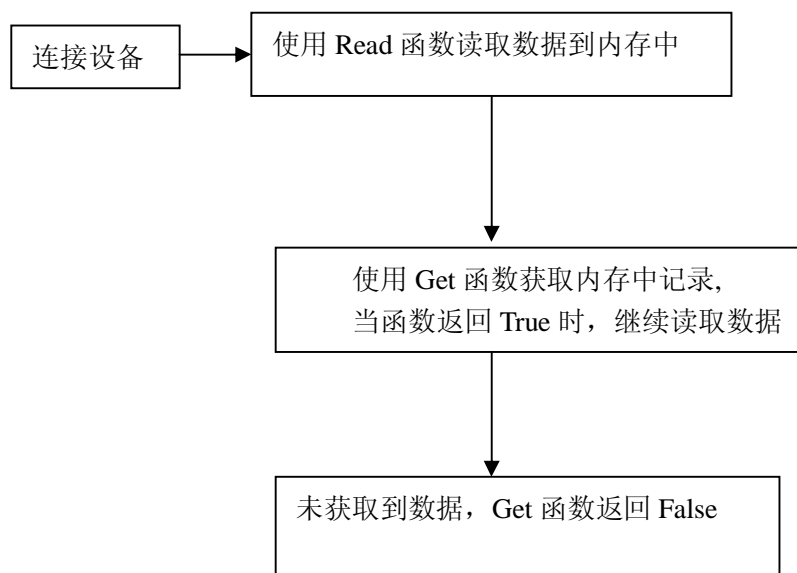
下面是注册事件的定义。注册所有事件是 EventMask=0x7FFF (按位注册事件)

```
#define EF_ATTLOG 1 //考勤登记事件 /OnAttTransaction
#define EF_FINGER (1<<1)
#define EF_ENROLLUSER (1<<2)
#define EF_ENROLLFINGER (1<<3)
#define EF_BUTTON (1<<4)
#define EF_UNLOCK (1<<5) //开锁
#define EF_STARTUP (1<<6) //起动系统
#define EF_VERIFY (1<<7) //验证指纹
#define EF_FPFTR (1<<8) //提取指纹特征点
#define EF_ALARM (1<<9) //报警信号
#define EF_HIDNUM (1<<10) //射频卡号码
#define EF_WRITECARD (1<<11) //写卡成功
#define EF_EMPTYCARD (1<<12) //清除卡成功
```

比如:IF CZKEM1.RegEvent(devID,1) then //注册事件 OnAttTransaction  
showmessage('regevent sucess!');

IF CZKEM1.RegEvent(devID,32767) then //注册所有事件, reg all event  
showmessage('regevent sucess!');

## 3.5 数据下载一般流程（考勤记录、用户信息、指纹）





下载完成

## 4. 函数详细描述

### 4.1 数据管理

#### 4.1.1 用户信息

##### 4.1.1.1 GetUserInfo

###### [函数]

VARIANT\_BOOL GetUserInfo([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] BSTR \* Name, [in] BSTR \* Password, [in] LONG \* Privilege, [in] VARIANT\_BOOL \* Enabled)

###### [功能]

通过用户登记号取得用户信息

###### [参数说明]

请参阅 SetUserInfo

###### [返回值]

成功则返回 True,否则返回 False。

###### [例子]

```
Dim dwEnrollNumber
Dim name As String
Dim password As String
Dim privileg As Integer
Dim enable As Boolean
Dim temp As Boolean
```

```
dwEnrollNumber = 55555
```

```
temp = CZKEM1.GetUserInfo(machineNumber, dwEnrollNumber, name, password, privileg, enable)
```

##### 4.1.1.2 SetUserInfo

###### [函数]

VARIANT\_BOOL SetUserInfo([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] BSTR Name, [in] BSTR Password, [in] LONG Privilege, [in] VARIANT\_BOOL Enabled)

###### [功能]

相当于登记一个用户，写入用户的个人信息。

###### [参数说明]

**dwEnrollNumber**

用户登记登记号。

**Name**

用户姓名。

### **Password**

用户密码。

### **Privilege**

用户权限；0，普通用户；1，登记员；2，管理员 3，超级管理员。

### **Enabled**

用户是否为禁止，false 为禁止。

### **[返回值]**

成功则返回 True,否则返回 False。

### **[例子]**

```
Dim dwEnrollNumber As Long
Dim name As String
Dim password As String
Dim privileg As Long
Dim enabled As Boolean
```

```
dwEnrollNumber = 1
name = "Henry"
password = "12"
privileg = 3
enabled = True
```

\CZKEM1.CardNumber(0) = 234 在这里设置用户卡号。

```
CZKEM1.SetUserInfo MACHINENUMBER, CInt(dwEnrollNumber), CStr(name), CStr(password),
CInt(privileg), CBool(enabled)
```

### **说明：**

该函数功能相当于登记用户，但还没有写入指纹模板，可用 SetUsertmpStr\SetUsertmp 函数上传指纹模板数据。

## **4.1.1.3 ClearAdministrators**

### **[函数]**

```
VARIANT_BOOL ClearAdministrators([in] LONG dwMachineNumber)
```

### **[功能]**

清除所有管理员权限。

### **[参数说明]**

#### **dwMachineNumber**

当前所要操作的机器编号。

### **[返回值]**

成功则返回 True,否则返回 False。

### **[例子]**

```
CZKEM1.ClearAdministrators MACHINENUMBER
```

### **[说明]**

此函数只是删除管理员的权限，不删除管理员对应的用户（管理员是特殊的用户）。

## **4.1.1.4 DeleteEnrollData**

### **[函数]**

```
VARIANT_BOOL DeleteEnrollData([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber,
```

[in] LONG dwEMachineNumber, [in] LONG dwBackupNumber)

**[功能]**

删除某一用户。

**[参数说明]**

**dwMachineNumber**

要操作的机器号。

**dwEnrollNumber**

用户登记号码。

**dwEMachineNumber**

要操作的机器号。

**dwBackupNumber**

用户备份号码，详情请看说明。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim dwEnrollNumber As Integer
Dim dwEmachineNumber As Integer
Dim dwBackupNumber As Integer
```

```
dwEnrollNumber = 1
dwEmachineNumber = 1
dwBackupNumber=0
```

```
CZKEM1.DeleteEnrollData MACHINENUMBER, dwEnrollNumber, dwEmachineNumber,
dwBackupNumber
```

```
CZKEM1.RefreshData MACHINENUMBER `刷新数据。
```

**说明**

备份号用于指定登记了多个指纹的用户指纹模板，如用户登记了两个指纹要删除第二个指纹模板，则指定 dwBackupNumber 为 1，以此类推；但是如果大于 9，意义为： 10，密码数据 11，所有指纹数据 12，所有指纹数据及密码数据。

### 4.1.1.5 GetUserInfoByCard

**[函数]**

VARIANT\_BOOL GetUserInfoByCard([in] LONG dwMachineNumber, [in] BSTR \* Name, [in] BSTR \* Password, [in] LONG \* Privilege, [in] VARIANT\_BOOL \* Enabled)

**[功能]**

根据卡号获得用户信息。

**[参数说明]**

请参考 GetUserInfoByPIN2 函数。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim name As String
Dim password As String
Dim privileg As Long
Dim enabled As Boolean
Dim MachineNumber
```

```
CZKEM1.CardNumber(0) = 234
MachineNumber = 1
CZKEM1.GetUserInfoByCard MachineNumber, name, password, privileg, enabled
```

#### 4.1.1.6 BeginBatchUpdate

##### [函数]

```
BeginBatchUpdate([in] LONG dwMachineNumber,[in] LONG UpdateFlag, [out,retval]
VARIANT_BOOL* pVal)
```

##### [功能]

开始批处理模式上传，目前，ZEM100 系列在 5.11 版本后具支持该功能。ZEM200 产品在 2006-9-15 以后产品具支持该功能。请按照实际情况检查。当上传用户信息、指纹模板之前，执行该函数，SDK 将缓存所有数据，直到执行所有上传后，有一对应函数 BatchUpdate 一次性将数据上传至机器。该函数相对 SetUserinfo\SetUserTmp 在循环中多次上传效率极大提高。

##### [参数说明]

UpdateFlag: 上传标志。1: 强行覆盖机器上存在的指纹。0: 不覆盖已经存在的指纹。  
pVal: 函数返回值。

##### [返回值]

成功则返回 True,否则返回 False。

##### [例子]

```
Dim sqlstr As String
Dim tempUserID As Long
Dim tempdata() As Byte
Dim tempdataSize As Long
Dim recCount As Long
Dim i As Long
Dim falg As Boolean

tempUserID = 0
i = 0
Set recFPTEST = New Recordset
sqlstr = "select USERID,FINGERID,TEMPLATE3 from TEMPLATE"
recFPTEST.Open sqlstr, connTEST, adOpenKeyset, adLockOptimistic, -1
'请自行定义连接 connTEST
'sqlstr = "select USERID,FINGERID from TEMPLATE"
recCount = recFPTEST.RecordCount

CZKEM1.BeginBatchUpdate 1, 1 '批处理上传。
While Not recFPTEST.EOF
    If tempUserID <> recFPTEST.Fields("USERID") Then
        falg = CZKEM1.SetUserInfo(CLng(devid), recFPTEST.Fields("USERID"),
"Henry", 0, 0, True) 'devid 为机器号。
        tempUserID = recFPTEST.Fields("USERID")
    End If
    tempdataSize = Len(recFPTEST.Fields("TEMPLATE3"))
    ReDim tempdata(tempdataSize) As Byte
    tempdata = recFPTEST.Fields("TEMPLATE3")
    'necessary condition:user must exist
    'if user have a template for this finger id,it will retrun false.do not upload
    template.
    flag = CZKEM1.SetUserTmp(CLng(devid), recFPTEST.Fields("USERID"),
recFPTEST.Fields("FINGERID"), tempdata(0))
```

```

        i = i + 1
        recFPTEST.MoveNext
        DoEvents
    Wend
    CZKEM1.BatchUpdate 1    '开始批处理。
    CZKEM1.RefreshData CLng(devid)
    sqlstr = ""
    recFPTEST.Close

```

#### 4.1.1.7 BatchUpdate

##### [函数]

BatchUpdate([in] LONG dwMachineNumber, [out,retval] VARIANT\_BOOL\* pVal)

##### [功能]

上传批处理缓存下来的数据。例子请参考 BeginBatchUpdate 相关代码。

##### [参数说明]

DwMachineNumber: 机器编号。

pVal: 函数返回值。

##### [返回值]

成功则返回 True, 否则返回 False。

#### 4.1.1.8 ReadAllUserID

##### [函数]

VARIANT\_BOOL ReadAllUserID([in] LONG dwMachineNumber)

##### [功能]

读取所有的用户信息到 PC 内存中, 包括用户编号, 密码, 姓名, 卡号等, 指纹模板除外。

##### [返回值]

成功则返回 True, 否则返回 False。

##### [例子]

```
Dim userID As Integer
```

```
CZKEM1.ReadAllUserID MACHINENUMBER
```

##### 说明:

在该函数执行完成后, 可调用函数 GetAllUserID 得到用户登记号。GetAllUserID 每执行一次, 指向用户信息的指针移动到下一记录, 在读到最后一条记录时, 返回 False。

#### 4.1.1.9 GetUserInfoByPIN2

##### [函数]

VARIANT\_BOOL GetUserInfoByPIN2([in] LONG dwMachineNumber, [in] BSTR \* Name, [in] BSTR \* Password, [in] LONG \* Privilege, [in] VARIANT\_BOOL \* Enabled)

##### [功能]

以 PIN2 属性值取得用户信息。

##### [参数说明]

Stand alone development for ZKEMSDK, Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.13

**dwMachineNumber**

机器号。

**Name**

用户姓名。

**Password**

用户密码。

**Privilege**

用户权限。

**Enabled**

用户禁止或可用。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim name As String
Dim password As String
Dim privilege As Integer
Dim enabled As Boolean
```

czkem1.PIN2=1 ‘该属性类型定义为 unsigned long 类型。故在 VB 中暂不支持。

CZKEM1.GetUserInfoByPIN2 MACHINENUMBER, name, password, privilege, enabled

**说明:**

该函数功能类似 GetUserInfo,一般使用 GetUserInfo 获取用户信息即可, PIN2 即用户编号。

#### 4.1.1.10 GetPIN2

**[函数]**

VARIANT\_BOOL GetPIN2([in] LONG UserID, [in] LONG\* PIN2)

**[功能]**

以用户的登记号取得用户的 PIN2 值。

**[参数说明]**

**UserID**

机器内部编码 (2Bytes)。

**PIN2**

超出 5 位用户 ID 标示。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim Pin
Dim pin2 As Long
Pin = 1 '用户编号为 1
```

```
CZKEM1.GetPIN2 Pin, pin2
```

**说明**

在机器内部处理用户编号(以后称 PIN)为 2 字节,故为 5 位十进制编码。在有些机器中,用户编号为 9 位编码(PIN2)或跟多 (4 字节), 如果机器用户编号超过 2 字节 (9 位码), 则 PIN 是用户看不见的编码。该函数功能使用 PIN 获

得 PIN2。

#### 4.1.1.11 GetUserIDByPIN2

##### [函数]

VARIANT\_BOOL GetUserIDByPIN2([in] LONG PIN2, [in] LONG\* UserID)

##### [功能]

以用户的登记号取得用户的 PIN 值。可参考 GetPIN2 函数说明。

##### [参数说明]

###### **UserID**

用户在登记的时候分配的号码(PIN)。

###### **PIN2**

超出 5 位用户 ID 标示。

##### [返回值]

成功则返回 True,否则返回 False。

##### [例子]

```
Dim pin2
Dim userID As Long
pin2 = 1
CZKEM1.GetUserIDByPIN2 pin2, userID
```

##### 说明:

如上例, 根据用户编号 1(PIN2)获得机器内部 2 字节编号 PIN。由于 PIN ,PIN2 只是机器内部或 SDK 处理来使用, 故在开发中一般使用不到, 可以不参考该函数, 越过阅读。

#### 4.1.1.12 GetSMS

##### [函数]

VARIANT\_BOOL GetSMS(LONG dwMachineNumber, LONG ID, LONG\* Tag, LONG\* ValidMinutes, BSTR\* StartTime, BSTR\*Content)

##### [功能]

根据编号取得短消息。

##### [参数说明]

###### **ID**

短消息编号。

###### **Tag**

短消息类型, 253 位公共短消息, 254 为用户短消息。

###### **StartTime**

短消息开始时间, 格式为:yyyy-mm-dd hh:ss:mm.

###### **Content**

短消息内容.

###### **ValidMinutes**

有效分钟数.其值的范围是 0-65535,如果设为 0 表示永久有效.SMS 的有效时间是: StartTime+ValidMinutes, 如果超过这个时间,系统将不显示这条 SMS

##### [返回值]



成功则返回 True,否则返回 False。

**[例子]**

```
Dim Tag As Long
Dim ValidMinutes As Long
Dim StartTime As String
Dim Content As String
Dim MachineNumber
Dim SMSID

MachineNumber = 1
SMSID = 1
CZKEM1.GetSMS MachineNumber, 1, Tag, ValidMinutes, StartTime, Content
```

### 4.1.1.13 SetSMS

**[函数]**

VARIANT\_BOOL SetSMS([in] LONG dwMachineNumber, [in] LONG ID, [in] LONG Tag, [in] LONG ValidMinutes [in] BSTR StartTime, [in] BSTR content)

**[功能]**

写入短消息。如果想设置个人的短消息,使用该函数后,必须还要使 SetUserSMS 函数,来建立用户与短消息的关联关系.

**[参数说明]**

**ID**

短消息编号。

**Tag**

短消息类型, 253 位公共短消息, 254 为用户短消息。

**StartTime**

短消息开始时间, 格式为:yyyy-mm-dd hh:ss:mm.

**Content**

短消息内容.

**ValidMinutes**

有效分钟数.其值的范围是 0-65535,如果设为 0 表示永久有效.SMS 的有效时间是: StartTime+ValidMinutes, 如果超过这个时间,系统将不显示这条 SMS

**[例子]**

```
Dim Tag
Dim ValidMinutes
Dim StartTime
Dim Content
Dim MachineNumber
Dim SMSID

MachineNumber = 1
SMSID = 1
Content = "Hello henry"
Tag = 253
StartTime = "2006-12-12 12:12:00"
ValidMinutes = 15
```

CZKEM1.SetSMS MachineNumber, SMSID, Tag, ValidMinutes, StartTime, Content

#### 说明:

短消息不依附于用户，是独立的，在上传短消息后，再把非公共短消息分配给用户。

#### 4.1.1.18 GetAllUserInfo

##### [函数]

VARIANT\_BOOL GetAllUserInfo([in] LONG dwMachineNumber, [in] LONG \*dwEnrollNumber, [in] BSTR \* Name, [in] BSTR \* Password, [in] LONG \* Privilege, [in] VARIANT\_BOOL \* Enabled)

##### [功能]

取得用户信息，把用户所有信息读取出来，可以在一个循环中顺序读取用户信息。

##### [参数说明]

dwMachineNumber: 机器编号,在网络连接的时候，该参数无效。

##### [返回值]

成功则返回 True,否则返回 False。

##### [例子]

```
Dim EnrollNumber As Long
Dim Name As String
Dim pass As String
Dim pri As Long
Dim en As Boolean
Dim MachineNumber
MachineNumber = 1
CZKEM1.ReadAllUserID 1
While CZKEM1.GetAllUserInfo(MachineNumber, EnrollNumber, Name, pass, pri, en)
    DoEvents
    '其他处理代码。
Wend
```

#### 4.1.1.19 GetAllUserID

##### [函数]

VARIANT\_BOOL GetAllUserID([in] LONG dwMachineNumber, [in] LONG\* dwEnrollNumber, [in] LONG\* dwEMachineNumber, [in] LONG\* dwBackupNumber, [in] LONG\* dwMachinePrivilege, [in] LONG\* dwEnable)

##### [功能]

取得所有用户信息。在该函数执行之前，可用 ReadAllUserID 读取到所有用户信息到内存，GetAllUserID 每执行一次，指向用户信息指针移到下一记录，当读完所有用户信息后，函数返回 False。

##### [参数说明]

##### dwMachinePrivilege

用户权限：0，普通用户；1，登记员；2，管理员；3 超级管理员。

##### [返回值]

成功则返回 True,否则返回 False。

##### [例子]

该函数和 GetAllUserInfo 实现相同，请参考 GetAllUserInfo 例子。

##### 说明

该函数可利用 ReadAllUserID 函数把所有用户信息读取到内存中，然后循环读出用户信息。

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.17

#### 4.1.1.20 GetBackupNumber

**[函数]**

LONG GetBackupNumber([in] LONG dwMachineNumber)

**说明:**

该函数已丢弃，无效。

#### 4.1.1.21 ModifyPrivilege

**[函数]**

VARIANT\_BOOL ModifyPrivilege([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG dwEMachineNumber, [in] LONG dwBackupNumber, [in] LONG dwMachinePrivilege)

**[功能]**

修改用户权限。

**[参数说明]**

**dwBackupNumber**

指纹备份号，暂不支持根据备份号修改用户某一指纹权限

**dwMachinePrivilege**

用户权限；0，用户；1，登记员；2，管理员；3，超级管理员。

**[返回值]**

成功则返回 True, 否则返回 False。

**[例子]**

```
Dim dwEnrollNumber ' user ID
Dim dwEmachineNumber
Dim dwBackupNumber
Dim dwMachinePrivilege
```

```
dwEnrollNumber = 1
dwEmachineNumber = 1
dwBackupNumber = 0
dwMachinePrivilege = 3
```

```
CZKEM1.ModifyPrivilege MACHINENUMBER, dwEnrollNumber, dwEmachineNumber, dwBackupNumber, dwMachinePrivilege
```

#### 4.1.1.22 DeleteSMS

**[函数]**

VARIANT\_BOOL DeleteSMS(LONG dwMachineNumber, [in] LONG ID);

**[功能]**

删除某一编号对应的的短消息。

**[参数说明]**

ID 短消息编号。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim b As Boolean

If CZKEM1.DeleteSMS(vMachineNumber, 1) Then
    MsgBox "DeleteSMS OK"
else
    MsbBox "DeleteSMS fail"
End If
```

### 4.1.1.23 SetUserSMS

#### [函数]

VARIANT\_BOOL SetUserSMS([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG SMSID);

#### [功能]

设置用户对应的短消息。

#### [参数说明]

dwEnrollNumber: 用户编号。

SMSID: 短消息编号。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim EnrollNumber
Dim MachineNumber
Dim SMSID

EnrollNumber=1
MachineNumber=1
SMSID=1
CZKEM1.SetUserSMS MachineNumber, EnrollNumber, SMSID
```

### 4.1.1.24 DeleteUserSMS

#### [函数]

VARIANT\_BOOL SetUserSMS([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG SMSID);

#### [功能]

删除用户对应的短消息。

#### [参数说明]

dwEnrollNumber: 用户编号。

SMSID: 短消息编号。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim bFlag As Boolean
Dim SMSID
```

```
SMSID=1
If CZKEM1.DeleteSMS(MachineNumber, SMSID) Then
    MsgBox "DeleteSMS OK"
Else
    MsgBox "DeleteSMS fail"
End If
```

#### 4.1.1.25 ClearSMS

##### [函数]

```
VARIANT_BOOL ClearSMS([in] LONG dwMachineNumber);
```

##### [功能]

清空短消息。

##### [参数说明]

dwMachineNumber: 机器号。

##### [返回值]

成功则返回 True, 否则返回 False。

##### [例子]

```
CZKEM1.ClearSMS MachineNumber
```

#### 4.1.1.26 ClearUserSMS

##### [函数]

```
VARIANT_BOOL ClearUserSMS([in] LONG dwMachineNumber);
```

##### [功能]

清空用户短消息。

##### [参数说明]

dwMachineNumber: 机器号。

##### [返回值]

成功则返回 True, 否则返回 False。

##### [例子]

```
CZKEM1.ClearUserSMS MachineNumber
```

#### 4.1.1.27 SetUserInfoEx

##### [函数]

```
SetUserInfoEx([in] LONG dwMachineNumber, [in] long dwEnrollNumber, [in] long
VerifyStyle, [in] BYTE* Reserved, [out,retval] VARIANT_BOOL* pVal)
```

##### [功能]

上传用户验证方式或组验证方式。只有有多种验证方式的机器可支持该函数。该类函数功能于 2006-08 左右发布开始使用。

##### [参数说明]

dwEnrollNumber: 用户编号。

VerifyStyle: 比对方式。设置用户验证方式值得范围为 1-15 共 15 种验证方式。如果使用组的验证方式，

则验证方式设置从 129-134。验证方式递增顺序和机器中验证方式菜单项顺序相同。

彩屏门禁指纹机的值为:0 表示使用组验证方式, 128(FP/PW/RF), 129(FP), 130(PIN), 131(PW), 132(RF), 133(FP&RF), 134(FP/PW), 135(FP/RF), 136(PW/RF), 137(PIN&FP), 138(FP&PW), 139(PW&RF), 140(FP&PW&RF), 141(PIN&FP&PW), 142(FP&RF/PIN)。

#### [返回值]

成功则返回 True,否则返回 False。

```
Dim aflag As Boolean
Dim reser As Byte
Dim EnrollNumber
Dim VerifyStyle
EnrollNumber=1
VerifyStyle=1
reser = 0
aflag = CZKEM1.SetUserInfoEx(MachineNumber, EnrollNumber, VerifyStyle, reser)
If aflag = 1 Then
    CZKEM1.RefreshData MachineNumber
    MsgBox "successfully", vbInformation + vbOKOnly, "info"
Else
    MsgBox "fail", vbInformation + vbOKOnly, "info"
End If
```

### 4.1.1.28 GetUserInfoEx

#### [函数]

GetUserInfoEx([in] long dwMachineNumber, [in] long dwEnrollNumber, [in,out] LONG\* VerifyStyle, [in,out] BYTE\* Reserved, [out,retval] VARIANT\_BOOL\* pVal)

#### [功能]

获取用户的验证方式。

#### [参数说明]

Reserved: 保留参数, 无效。

其他请参考 SetUserInfoEx 的说明。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim aflag As Boolean
Dim reser As Byte
Dim verifyType As Long
Dim UserID
UserID=1
aflag = CZKEM1.GetUserInfoEx(MachineNumber, UserID, verifyType, reser)
If aflag = 1 Then
    MsgBox "successfully", vbInformation + vbOKOnly, "info"
Else
    MsgBox "fail", vbInformation + vbOKOnly, "info"
End If
```

### 4.1.1.29 GetStrCardNumber

#### [函数]

GetStrCardNumber(BSTR\* ACardNumber)

#### [功能]

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.21

以字符串形式返回卡号。此功能只在 SDK 版本为 6.0 以上支持。

#### [参数说明]

AcardNumber: 卡号。

#### [返回值]

成功则返回 True, 否则返回 False。

#### [例子]

```
Dim MachineNumber As Long
Dim UserNo As Long
Dim Privilege As Long
Dim UserName As String
Dim Password As String
Dim CardNo As String
Dim Enabled As Boolean

MachineNumber = 1;
While CZKEM1.GetAllUserInfo(MachineNumber, UserNo, UserName, Password, Privilege, Enabled)
    ...
    CZKEM1.GetStrCardNumber(CardNo)
    ...
Wend
```

### 4.1.1.30 SetStrCardNumber

#### [函数]

SetStrCardNumber (BSTR ACardNumber )

#### [功能]

以字符串方式设置用户卡号。

#### [参数说明]

AcardNumber: 卡号。

#### [返回值]

成功则返回 True, 否则返回 False。

#### [例子]

```
Dim CardNo As String
Dim UserName As String
Dim Password As String
Dim Privilege As Long
Dim MachineNumber As Long
Dim Privilege As Long
Dim UserNo As Long
Dim Enabled As Boolean

CardNo = "123456"
UserName = "Tom"
Password = "123"
Privilege = 0
```

```
MachineNumber = 1
UserNo = 1
Enabled = True
CZKEM1. SetStrCardNumber(CardNo)
CZKEM1. SetUserInfo(MachineNumber, UserNo, UserName, Password, Privilege, Enabled)
...
...
```

## 4.1.2 指纹管理

### 4.1.2.1 GetUserTmpStr

#### [函数]

VARIANT\_BOOL GetUserTmpStr([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG dwFingerIndex, [out] BSTR\* TmpData, [out] LONG \* TmpLength)

#### [功能]

以字符串形式取得用户指纹模板信息。

#### [参数说明]

##### **dwFingerIndex**

指纹索引，范围为 0—9。

##### **TmpData**

指纹模板信息。

##### **TmpLength**

指纹模板长度。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim dwEnrollNumber
Dim dwFingerIndex
Dim tmpData As String 'Finger data of enroll
Dim tmpLength As long

dwEnrollNumber = 1
dwFingerIndex = 0
CZKEM1.GetUserTmpStr MACHINENUMBER, dwEnrollNumber,dwFingerIndex, tmpData,
tmpLength
```

#### 说明：

建议用此函数取得用户指纹模板信息，相对容易处理。字符串指纹只是对二进制指纹模板的 Base64 编码。

### 4.1.2.2 SetUserTmpStr

#### [函数]

VARIANT\_BOOL SetUserTmpStr([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG dwFingerIndex, [in] BSTR TmpData)

#### [功能]

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.23



以字符串形式写入用户指纹模板。意思即 把 dwEnrollNumber 对应的用户的第 dwFingerIndex 枚指纹模板 TmpData 上传到所连接的机器中。声明部分[out] 表示参数为传出的，一般为指针，[in]表示参数是传入的，一般为普通变量，非指针。

#### [参数说明]

请参阅 SetUserTmp 函数说明。

#### [例子]

```
Dim dwEnrollNumber
Dim dwFingerIndex
Dim tmpData

dwEnrollNumber = 1
dwFingerIndex = 0

tmpData="ocojg52rWoEOOqlégQwlrEtBFp4uRAESmkBLQRZ0wlLBB21BKUEM3EiuQTPmKGhBCCm8fE
kdw7MnQRE6QCXBC9DDVVEE3Kk3QR0iFjvBDRJACKEMz5VggQYbMn1BDy8uKwkNMItpyQ0VL0uBSJozS
4FQhR8/ARSDoTHBil0sIYEKQKYlghJDoxlBD02aKcERZJwaQRBbhioBKHkRS4EJhyUygVtEozPBPwi4
PsEQij5DQQL8HXQJDZtkLBOrMM8LEBHCAGQPFBgPBoHAwgKjrfxTfBfAwgIEDKLaiZwdwMF1pKzLrMu
qIcDBc6WZ693rmJrAwW+km87vzJmCwMFmoa3/DBWjy5qG"

CZKEM1.SetUserTmpStr MACHINENUMBER, dwEnrollNumber, dwFingerIndex, tmpData
```

### 4.1.2.3 DelUserTmp

#### [函数]

VARIANT\_BOOL DelUserTmp([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG dwFingerIndex)

#### [功能]

删除用户指纹模板信息。

#### [参数说明]

##### **dwFingerIndex**

用户登记指纹的指纹号，范围为 0—9，即十个手指。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim dwEnrollNumber As Integer
Dim dwFingerIndex As Integer

dwEnrollNumber = 1
dwFingerIndex = 0
`删除 1 号用户的第一枚指纹，用户指纹编号可以是 0-9.
CZKEM1.DelUserTmp MACHINENUMBER, dwEnrollNumber, dwFingerIndex
```

#### 说明

该函数功能和 DeleteEnrollData 删除指纹模板相同。

### 4.1.2.4 GetEnrollDataStr

#### [函数]

```
HRESULT GetEnrollDataStr( [in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in]
LONG dwEMachineNumber, [in] LONG dwBackupNumber, [out] LONG* dwMachinePrivilege, [out]
BSTR* dwEnrollData, [out] LONG* dwPassWord);
```

#### [功能]

取得用户信息，包含指纹模板数据。

#### [参数说明]

**dwEnrollNumber** 和 **dwEMachineNumber** 一样，都是指机器号。

**dwBackupNumber**

指纹备份号，即指纹索引。

**DwMachinePrivilege**

用户权限。0，普通用户，1，登记员，2，管理员，3，超级管理员

**DwEnrollData**

指纹数据，以字符串形式表示。

**DwPassWord**

用户密码。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim dwEnrollNumber
Dim dwEmachineNumber
Dim dwBackupNumber
Dim dwMachinePrivilege As Long
Dim dwEnrollData As String
Dim dwPassWord As Long
Dim MachineNumber
```

```
dwBackupNumber = 0
MachineNumber = 1
dwEnrollNumber = 1
dwEmachineNumber = 1
```

```
CZKEM1.GetEnrollDataStr MachineNumber, dwEnrollNumber, dwEmachineNumber,
dwBackupNumber, dwMachinePrivilege, dwEnrollData, dwPassWord
```

### 4.1.2.5 SetEnrollDataStr

#### [函数]

```
VARIANT_BOOL SetEnrollDataStr([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber,
[in] LONG dwEMachineNumber, [in] LONG dwBackupNumber, [in] LONG dwMachinePrivilege, [in]
BSTR dwEnrollData, [in] LONG dwPassWord)
```

#### [功能]

以字符串形式写入用户指纹模板。

#### [参数说明]

请参阅 SetUserTmp 函数说明。功能和 SetUserTmp 相同，内部实现一样，SetUserTmp 是最新的接口。

#### [例子]

```
Dim dwEnrollNumber
Dim dwEmachineNumber
Dim dwBackupNumber
Dim dwMachinePrivilege
Dim dwEnrollData
Dim dwPassWord
```

```
dwEnrollNumber = 1
dwEmachineNumber = 1
dwBackupNumber = 0
dwMachinePrivilege = 1
dwEnrollData =
"ocojg52rWoEOOqllegQwlrEtBFp4uRAESmkBLQRZ0wlLBB21BKUEM3EIuQTPmKGhBCCm8fEkdw7MnQR
E6QCXBC9DDVVEE3Kk3QR0iFjvBDRJAcKEMz5VggQYbMn1BDy8uKwkNMItPyQ0VL0uBSJozS4FQhR8/A
RSDoTHBil0sIYEKQKYlghJDoxlBD02aKcERZJwaQRBbhioBKHkRS4EJhyUygVtEozPBPwi4PseQi j5D
QQ18HXQJDZtkLBO rMM8LEBHCAgQPFBgPBoHAwgKjrfxTfBfAwgIEDKLaiZwdwMF1pKzLrMuqIcDBc6W
Z693rmJrAwW+km87vzJmCwMFmoa3/DBWjy5qG"
dwPassWord = 1
```

CZKEM1.SetEnrollDataStr MACHINENUMBER, dwEnrollNumber, dwEmachineNumber, dwBackupNumber, dwMachinePrivilege, dwEnrollData, dwPassWord

#### 4.1.2.6 SetUserTmp

##### [函数]

VARIANT\_BOOL SetUserTmp([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG dwFingerIndex, [in] BYTE\* TmpData)

##### [功能]

以二进制形式写入 (upload to device) 用户指纹模板。

##### [参数说明]

###### **dwFingerIndex**

指纹对应的索引号。

###### **TmpData**

指纹模板数据，二进制格式。

##### [返回值]

成功则返回 True, 否则返回 False。

##### [例子]

```
Dim dwEnrollNumber As Integer
Dim dwFingerIndex As Integer
Dim tmpData(1024) As Byte
```

```
dwEnrollNumber = 1
dwFingerIndex = 0
'tmpData 将二进制指纹模板传赋给该数组，如从数据库读取等。
```

CZKEM1.SetUserTmp MACHINENUMBER, dwEnrollNumber, dwFingerIndex, tmpData(0)

##### 说明:

在 vb6.0 中，二进制数组 tmpData(0) 意为传递地址。

#### 4.1.2.7 GetUserTmp

##### [函数]

VARIANT\_BOOL GetUserTmp([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG dwFingerIndex, [in,out] BYTE\* TmpData, [in,out] LONG \* TmpLength);

Stand alone development for ZKEMSDK, Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.26

**[功能]**

取得用户指纹模板信息。

**[参数说明]**

**dwEnrollNumber**

用户登记号。

**DwFingerIndex**

指纹索引，范围为 0—9。

**TmpData**

指纹模板字节数组。

**TmpLength**

模板长度。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim dwEnrollNumber
Dim dwFingerIndex
Dim tmpData(1024) As Byte
Dim tmpLength As Long

dwEnrollNumber = 1
dwFingerIndex = 0
CZKEM1.GetUserTmp MachineNumber, dwEnrollNumber, dwFingerIndex, tmpData(0),
tmpLength
```

#### 4.1.2.8 GetEnrollData

**[函数]**

VARIANT\_BOOL GetEnrollData( [in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG dwEMachineNumber, [in] LONG dwBackupNumber, [out] LONG\* dwMachinePrivilege, [out] LONG\* dwEnrollData, [out] LONG\* dwPassWord)

**[功能]**

取得用户指纹模板和密码。该函数以二进制方式取得指纹模板和用户密码。

**[参数说明]**

dwBackupNumber：指纹索引。

dwEnrollData：用户指纹模板。

其他请参考 GetEnrollDataStr 函数说明。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

请参阅 GetFPTempLength 函数例子。

**说明：**

该函数为旧版本 SDK 函数，可用 GetUserTmpStr\GetUserTmp 等代替获取指纹模板。

Dim dwmachineNumber

Dim dwEnrollNumber

Dim dwEmachineNumber

Dim dwBackupNumber

```
Dim pri As Long
Dim enrolldata(1024) As Byte
Dim pass As Long
dwmachineNumber = 1
dwEnrollNumber = 1
dwEmachineNumber = 1
dwBackupNumber = 0
```

CZKEM1.GetEnrollData dwmachineNumber, dwEnrollNumber, dwEmachineNumber, dwBackupNumber, pri, enrolldata(0), pass

#### 4.1.2.9 SetEnrollData

##### [函数]

```
VARIANT_BOOL SetEnrollData([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG dwEMachineNumber, [in] LONG dwBackupNumber, [in] LONG dwMachinePrivilege, [in] LONG* dwEnrollData, [in] LONG dwPassWord)
```

##### [功能]

对某一用户写入(上传)其指纹模板及密码。

##### [参数说明]

###### **dwMachineNumber**

要操作的机器号。

###### **DwEnrollNumber**

用户登记号。

###### **DwEMachineNumber**

和 dwMachineNumber 一样。

###### **DwBackupNumber**

指纹备份号，指指纹索引。

###### **dwMachinePrivilege**

用户权限。

###### **DwEnrollData**

指纹数据。

###### **DwPassWord**

用户密码。

##### [返回值]

成功则返回 True, 否则返回 False。

##### [例子]

该函数内部实现较 SetUserInfo 及 SetUserTmp 效率低，可用 SetUserInfo 及 SetUserTmp 替代。

#### 4.1.2.10 ReadAllTemplate

##### [函数]

```
VARIANT_BOOL ReadAllTemplate([in] LONG dwMachineNumber)
```

**[功能]**

读取所有的指纹模板到内存中。

**[参数说明]**

**dwMachineNumber**

要操作的机器号。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
CZKEM1.ReadAllTemplate MACHINENUMBER
```

**说明:**

该函数的主要功能是提高取指纹模板的速度，相对于一个一个从脱机指纹机取指纹模板，速度是快多了。所以获取用户、考勤记录、指纹模板信息，和机器通信次数很少。

### 4.1.2.11 FPTempConvertNew

**[函数]**

```
VARIANT_BOOL FPTempConvertNew([in] BYTE* TmpData1, [in] BYTE* TmpData2, [in] LONG *Size)
```

**[功能]**

把 Biokey SDK 采集到的指纹模板转换为脱机指纹机使用的指纹模板。

**[参数说明]**

**TmpData1**

需要转换的模板。

**TmpData2**

转换后的指纹模板。

**Size**

转换后的模板长度。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim tmpData1(1024) As Byte '该参数赋值 BiokeySDK 采集指纹模板。
Dim tmpdata2(1024) As Byte
Dim size As Integer
```

```
CZKEM1.GetUserTmp MACHINENUMBER, 1, 0, tmpData1(0), size
```

```
CZKEM1.FPTempConvertNew tmpData1(0), tmpdata2(0), size
```

**说明**

以上程序例子需要给参数 temData1 传递一个字节数组，tmpData1(0)意为传递地址，即传给二进制变量全部。不过二进制处理不是十分方便，可以使用 FPTempConvertNewStr 函数进行转换，方便于处理。还有对应函数 FPTempConvert 将脱机模板转换为 BIOKEY SDK 模板。

### 4.1.2.12 FPTempConvertNewStr

**[函数]**

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.29

```
VARIANT_BOOL FPTempConvertNewStr([in] BSTR TmpData1, [in] BSTR* TmpData2, [in] LONG
*Size)
```

#### [功能]

将 BIOKEY SDK 模板转换为脱机指纹机中指纹模板，以字符串形式转换。当然，将二进制指纹模板 Base64 编码，即可得到该字符串。（由于每次按压不同，每次采集到的指纹不尽相同，请注意）。

#### [参数说明]

##### **TmpData1**

需要转换的模板。

##### **TmpData2**

需要转换的模板。

##### **Size**

转换后的模板长度

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim tmpData1
Dim tmpdata2 As String
Dim size As Integer
```

```
tmpData1="re2df3fd4fsfsfsddfnCTRXERcIQnkA8ghQqLUTCDBikWAIJGdKhTggWdMsDECE32o1RI
qxm4wf3fwYfsMCNDFWdFDfJosIN"
```

```
CZKEM1.FPTempConvertNewStr tmpData1, tmpdata2, size
```

### 4.1.2.13 FPTempConvertStr

#### [函数]

```
VARIANT_BOOL FPTempConvertStr([in] BSTR TmpData1, [in] BSTR* TmpData2, [in] LONG
*Size)
```

#### [功能]

将脱机指纹机中指纹模板转换为 BIOKEY SDK 模板，以字符串形式转换。

#### [参数说明]

##### **TmpData1**

需要转换的模板。

##### **TmpData2**

需要转换的模板。

##### **Size**

转换后的模板长度。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim tmpData1
Dim tmpdata2 As String
Dim size As Integer
```

```
tmpData1 = "
ocojg52rWoEOOqlégQwlrEtBFp4uRAESmkBLQRZ0wlLBB21BKUEM3EIuQTPmKGhBCCm8fEkdw7MnQRE
6QCXBC9DDVVVEE3Kk3QR0iFjvBDRJACKEMz5VggQYbMn1BDy8uKwkNMItpyQ0VL0uBSJozS4FQhR8/AR
SDoTHBil0sIYEKQKYlghJDoxlBD02aKcERZJwaQRBbhioBKHkRS4EJhyUygVtEozPBPwi4PseQi j5DQ
```

Q18HXQJDZtkLBOrMM8LEBHCAgQPFBgPBoHAWgKjrfxTfBfAwgIEDKLaiZwdwMF1pKzLrMuqIcDBc6WZ  
693rmJrAwW+km87vzJmCwMFmoa3/DBWjy5qG "

CZKEM1.FPTempConvertStr tmpData1, tmpdata2, size

#### 4.1.2.14 FPTempConvert

##### [函数]

VARIANT\_BOOL FPTempConvert([in] BYTE\* TmpData1, [in] BYTE\* TmpData2, [in] LONG \*Size)

##### [功能]

将脱机指纹机中指纹模板转换为 BIOKEY SDK 模板。

##### [参数说明]

###### **TmpData1**

需要转换的模板。

###### **TmpData2**

转换后的指纹模板

###### **Size**

转换后的模板长度

##### [返回值]

成功则返回 True, 否则返回 False。

##### [例子]

请参考 FPTempConvertNew 函数例子。

#### 4.1.2.15 GetFPTempLengthStr

##### [函数]

LONG GetFPTempLengthStr([in] BSTR dwEnrollData)

##### [功能]

取得指纹模板长度。指纹模板以二进制方式来计算，字符串长度不能正确衡量指纹模板长度。

##### [参数说明]

###### **dwEnrollData**

某一指纹模板。

##### [返回值]

指纹模板的长度。

##### [例子]

Dim TmpData1

Dim templen As Long

TmpData1 =

"ocojg52rWoEOOqlégQw1rEtBFp4uRAESmkBLQRZ0wlLBB21BKUEM3EIuQTPmKGhBCCm8fEkdw7MnQRE6QC  
XBC9DDVVEE3Kk3QR0iFjvBDRJACKEMz5VggQYbMn1BDy8uKwkNMItPyQ0VL0uBSJozS4FQhR8/ARSDoTHBI  
10sIYEKQKYlghJDoxlBD02aKcERZJwaQRBbhioBKHkRS4EJhyUygVtEozPBPwi4PsEQij5DQQ18HXQJDZtk  
LBOrMM8LEBHCAgQPFBgPBoHAWgKjrfxTfBfAwgIEDKLaiZwdwMF1pKzLrMuqIcDBc6WZ693rmJrAwW+km87  
vzJmCwMFmoa3/DBWjy5qG"

templen = CZKEM1.GetFPTempLengthStr(TmpData1)



#### 4.1.2.16 GetFPTempLength

**[函数]**

LONG GetFPTempLength([in] BYTE\* dwEnrollData)

**[功能]**

得到指纹模板长度。

**[参数说明]**

**dwEnrollData**

指纹模板。

**[返回值]**

模板长度。

**[例子]**

```
Dim length as Long
Dim glngEnrollData(1024)    ‘这里需要给该变量赋值。

length = CZKEM1.GetFPTempLength(glngEnrollData(0))
```

#### 4.1.3 记录管理

记录主要包含考勤记录和管理记录，这些数据只能下载不能上传，考勤记录和管理记录下载时模式相同，都是先读取所有记录到内存中，然后，读出每条记录。

##### 4.1.3.1 ReadSuperLogData

**[函数]**

VARIANT\_BOOL ReadSuperLogData([in] LONG dwMachineNumber)

**[功能]**

读取所有的管理记录到内存中。

**[参数说明]**

**dwMachineNumber**

要操作的机器号。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
CZKEM1.ReadSuperLogData MACHINENUMBER
```

##### 4.1.3.2 ReadGeneralLogData

**[函数]**

VARIANT\_BOOL ReadGeneralLogData([in] LONG dwMachineNumber)

**[功能]**

读取所有的考勤记录到内存中。

**[参数说明]**

### **dwMachineNumber**

要操作的机器号。

### **[返回值]**

成功则返回 True,否则返回 False。

### **[例子]**

CZKEM1.ReadGeneralLogData MACHINENUMBER

## **4.1.3.3 GetGeneralLogDataStr**

### **[函数]**

VARIANT\_BOOL GetGeneralLogDataStr([in] LONG dwMachineNumber, [out] LONG\* dwEnrollNumber, [out] LONG\* dwVerifyMode, [out] LONG\* dwInOutMode, [out] BSTR \*TimeStr);

### **[功能]**

从计算机内存获得一个考勤记录数据。该函数每执行一次，内存中存储记录指针下移一位，在获取记录时，从机器中读取所有考勤记录，因为考勤记录数据量相对很少。读到后，使用该函数获取所有考勤记录，当内存考勤记录指针移动到最后一位的时候，该函数返回 False。

### **[参数说明]**

#### **dwVerifyMode**

比对模式，如下所描述：

**值 描述**

1 指纹

0 密码

#### **dwInOutMode**

考勤状态，如下所描述：

**值 描述**

0 上班

1 下班

2 外出

3 外出返回

4 加班签到

5 加班签退

### **[返回值]**

成功则返回 True,否则返回 False，标明记录读取完成。

### **[例子]**

```
Dim dwEnrollNumber As LONG
Dim dwVerifyMode As LONG
Dim dwInOutMode As LONG
Dim timeStr As String
Dim i As Long

CZKEM1.ReadAllLogData machineNumber
i = i + 1
while CZKEM1.GetGeneralLogDataStr(machineNumber, dwEnrollNumber, dwVerifyMode,
dwInOutMode, timeStr)
    MsgBox "get success " & i & " times"
    i = i + 1
Wend
```

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.33

### 4.1.3.4 GetSuperLogData

#### [函数]

```
VARIANT_BOOL GetSuperLogData([in] LONG dwMachineNumber, [in] LONG* dwTMachineNumber,
[in] LONG* dwSEnrollNumber, [in] LONG* Params4, [in] LONG* Params1, [in] LONG* Params2,
[in] LONG* dwManipulation, [in] LONG* Params3, [in] LONG* dwYear, [in] LONG* dwMonth,
[in] LONG* dwDay, [in] LONG* dwHour, [in] LONG* dwMinute)
```

#### [功能]

从计算机内存里依序一个个获取管理记录数据。类似读取考勤记录、所有用户，可用 ReadSuperLogData 函数先读取所有管理记录，该函数执行一次，指向管理记录指针指向下一记录。

#### [参数说明]

##### Params4

指向 long 型变量的指针，该变量接收进行管理操作的管理者的登记机器号码值。

##### Params1

指向 long 型变量的指针，该变量接收管理操作对象的登记号码值。若操作对象为终端机本身（即修改终端机系统信息的管理操作），该参数值为 0

##### Params2

指向 long 型变量的指针，该变量接收管理操作对象的登记机器号码值。若操作对象为终端机本身（即修改终端机系统信息的管理操作），该参数值为 0

##### Params3

保留字段，无效。

##### dwManipulation

指向 long 型变量的指针，该变量接收在终端机上进行的管理操作的类型值。管理操作类型值的说明如下：

值	描述
0	开机
1	关机
2	验证失败警告
3	拆机警告
4	进入菜单
5	更改设置
6	指纹备份登记
7	添加密码
8	登记HID卡
9	删除用户
10	删除指纹
11	删除密码
12	删除射频卡
13	清除数据
14	创建MF卡
15	登记MF卡
16	注册MF卡
17	删除MF卡注册
18	清除MF卡内容

- 19 把登记数据移到卡中
- 20 把卡中的数据复制到脱机指纹机中
- 21 设置脱机指纹机时间
- 22 恢复出厂设置
- 23 删除考勤(进出)记录
- 24 清除管理员权限
- 25 修改门禁组设置
- 26 修改用户门禁设置
- 27 修改门禁时间段
- 28 修改开锁组合设置
- 29 开锁
- 30 登记用户

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim dwTMachineNumber As Integer
Dim dwSenrollNumber As Integer
Dim params4 As Integer
Dim params1 As Integer
Dim params2 As Integer
Dim dwManipulation As Integer
Dim params3 As Integer
Dim dwYear As Integer
Dim dwMonth As Integer
Dim dwDay As Integer
Dim dwHour As Integer
Dim dwMinute As Integer
Dim i As Long
'dwTMachineNumber = 1
dwSenrollNumber = 1

CZKEM1.ReadAllSLogData machineNumber
i = 1
While CZKEM1.GetSuperLogData(machineNumber, dwTMachineNumber, dwSenrollNumber,
params4, params1, params2, dwManipulation, params3, dwYear, dwMonth, dwDay, dwHour,
dwMinute)
    MsgBox "Get super log successful " & i & " times"
    i = i + 1
Wend
```

### 4.1.3.5 GetAllSLogData

#### [函数]

```
VARIANT_BOOL GetAllSLogData([in] LONG dwMachineNumber, [in] LONG* dwTMachineNumber,
[in] LONG* dwSEnrollNumber, [in] LONG* Params4, [in] LONG* Params1, [in] LONG* Params2,
[in] LONG* dwManipulation, [in] LONG* Params3, [in] LONG* dwYear, [in] LONG* dwMonth,
[in] LONG* dwDay, [in] LONG* dwHour, [in] LONG* dwMinute)
```

#### [功能]

从计算机内从中读出管理记录,此内存中管理记录是由 ReadAllSLogData 函数读出的。和 GetSuperLogData 函数功能相同,实现完全一样,只是为了兼容,接口名称不同而已。

#### [参数说明]

请参考 GetSuperLogData 函数说明。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim dwTMachineNumber
Dim dwSenrollNumber As Long
Dim params4 As Long
Dim params1 As Long
Dim params2 As Long
Dim dwManipulation As Long
Dim params3 As Long
Dim dwYear As Long
Dim dwMonth As Long
Dim dwDay As Long
Dim dwHour As Long
Dim dwMinute As Long
Dim i As Long

CZKEM1.ReadAllSLogData MachineNumber
i = 1
While CZKEM1.GetAllSLogData(MachineNumber, dwTMachineNumber, dwSenrollNumber,
    params4, params1, params2, dwManipulation, params3, dwYear, dwMonth, dwDay,
    dwHour, dwMinute)
    DoEvents
    i = i + 1
Wend
```

### 4.1.3.6 GetAllGLogData

#### [函数]

VARIANT\_BOOL GetAllGLogData([in] LONG dwMachineNumber,[in] LONG\* dwTMachineNumber, [in] LONG\* dwEnrollNumber, [in] LONG\* dwEMachineNumber, [in] LONG\* dwVerifyMode, [in] LONG\* dwInOutMode, [in] LONG\* dwYear, [in] LONG\* dwMonth, [in] LONG\* dwDay, [in] LONG\* dwHour, [in] LONG\* dwMinute)

#### [功能]

从计算机内从中读出考勤记录,考勤记录为 ReadAllGLogData 函数读到内存中的考勤记录。该函数每执行一次,指向考勤记录指针移动到下一条记录,在读完记录后,函数返回 False。该函数功能和 GetGeneralLogDataStr 完全一样,实现相同。

#### [参数说明]

##### **dwEnrollNumber**

用户登记号

##### **dwEMachineNumber**

和 dwMachineNumber 一样

##### **dwVerifyMode**

比对模式, 0: 密码验证。1: 指纹验证。卡验证认为密码验证方式, 故在这里无区别。

##### **dwInOutMode**

考勤类型。可参考 GetGeneralLogDataStr 函数说明。

#### 4.1.3.7 ReadAllSLogData

**[函数]**

VARIANT\_BOOL ReadAllSLogData([in] LONG dwMachineNumber)

**[功能]**

读取所有的管理记录到内存中，和函数 ReadSuperLogData 一样的功能，请参阅。

#### 4.1.3.8 ReadAllGLogData

**[函数]**

VARIANT\_BOOL ReadAllGLogData([in] LONG dwMachineNumber)

**[功能]**

读取所有的考勤记录到内存中。和函数 ReadGeneralLogData 一样的功能，请参阅。为兼容存在。

#### 4.1.3.9 ClearSLog

**[函数]**

VARIANT\_BOOL ClearSLog([in] LONG dwMachineNumber)

**[功能]**

清除所有管理记录。

**[参数说明]**

**dwMachineNumber**

所要操作的机器号。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

CZKEM1.ClearSLog MACHINENUMBER

#### 4.1.3.10 GetGeneralExtLogData

**[函数]**

VARIANT\_BOOL GetGeneralExtLogData([in] LONG dwMachineNumber, [in,out] LONG\* dwEnrollNumber, [in,out] LONG\* dwVerifyMode, [in,out] LONG\* dwInOutMode, [in,out] LONG\* dwYear, [in,out] LONG\* dwMonth, [in,out] LONG\* dwDay, [in,out] LONG\* dwHour, [in,out] LONG\* dwMinute, [in,out] LONG\* dwSecond, [in,out] LONG\* dwWorkCode, [in,out] LONG\* dwReserved)

**[功能]**

下载考勤记录，是 GetGeneralLogData 函数的增强函数，可兼容 GetGeneralLogData。有的机器中有 WorkCode 功能，该函数可以获取到用户验证指纹时输入的 WorkCode。

**[参数说明]**

dwWorkCode: 用户按指纹时输入的 WorkCode 值。如果没有 WorkCode，则返回 0。

dwReserved: 为保留参数，无效。

其他参数请参考 GetGeneralLogData\GetGeneralLogData 函数说明。

#### [返回值]

成功则返回 True,否则返回 False。

### 4.1.4 系统数据管理

#### 4.1.4.1 BackupData

##### [函数]

VARIANT\_BOOL BackupData([in] BSTR DataFile)

##### [功能]

以文件的形式保存用户信息、指纹、记录等。该函数只支持 ZEM100 系列产品（A1,A2,A3,F4,F7 等）。

##### [参数说明]

###### DataFile

保存文件路径名，可以有盘符，如“c:\aaa.bak”。

##### [返回值]

成功则返回 True,否则返回 False。

##### [例子]

‘所有函数执行时，都需要连接上机器，请注意。

```
Dim dataFile
```

```
dataFile = "c:\AAA.bak"
```

```
CZKEM1.BackupData dataFile
```

##### [说明]

此函数可以和 RestoreData 配套使用。

#### 4.1.4.2 RestoreData

##### [函数]

VARIANT\_BOOL RestoreData([in] BSTR DataFile)

##### [功能]

从备份文件恢复机器数据。该函数只支持 ZEM100 系列产品（A1,A2,A3,F4,F7 等）。

##### [参数说明]

###### DataFile

数据文件，指由函数 BackupData 备份下的数据文件，如 c:\xxx.bak。

##### [返回值]

成功则返回 True,否则返回 False。

##### [例子]

```
Dim dataFile
```

```
dataFile = "c:\AAA.bat"
```

```
CZKEM1.BackupData dataFile
```

##### 说明：

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.38

如果进行串口通信，在函数恢复机器数据的时候，通常情况需要时间较多，需要耐心等待。

#### 4.1.4.3 ClearKeeperData

**[函数]**

VARIANT\_BOOL ClearKeeperData([in] LONG dwMachineNumber)

**[功能]**

清除脱机指纹机中的所有数据。如用户信息，指纹，考勤记录，管理记录。

**[参数说明]**

**dwMachineNumber**

当前所要操作的机器编号。

**[返回值]**

成功则返回 True, 否则返回 False。

**[例子]**

CZKEM1.ClearKeeperData MACHINENUMBER

**[说明]**

由于此函数能清空脱机指纹机中所有数据，所以使用要慎重。

#### 4.1.4.4 PINWidth

**[函数]**

LONG PINWidth()

**[功能]**

取得用户登记号最大长度。这是一个属性，最大用户编号一般为 5 个十进制长，或大于 5 位，如 9 位等。

**[返回值]**

返回用户登记号最大长度。

**[例子]**

Dim tmpWidth As Integer

tmpWidth = CZKEM1.PINWidth()

#### 4.1.4.5 RefreshData

**[函数]**

VARIANT\_BOOL RefreshData([in] LONG dwMachineNumber)

**[功能]**

刷新数据。

**[返回值]**

成功则返回 True, 否则返回 False。

**[例子]**

CZKEM1.RefreshData MACHINENUMBER

**说明:**

在上传用户信息或指纹后调用，这样能使所作的修改立即起作用，起到同步作用。



## 4.2 门禁

### 4.2.1 ACUnlock

**[函数]**

VARIANT\_BOOL ACUnlock([in] LONG dwMachineNumber, [in] LONG Delay)

**[功能]**

开门，使门锁控制器输出开门电平。

**[参数说明]**

**dwMachineNumber**

要操作的机器号。

**Delay**

开门延迟时间。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim delay As Integer 'Delay time
```

```
delay = 150
```

```
CZKEM1.ACUnlock MACHINENUMBER, delay
```

### 4.2.2 EnableUser

**[函数]**

VARIANT\_BOOL EnableUser([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG dwEMachineNumber, [in] LONG dwBackupNumber, VARIANT\_BOOL bFlag)

**[功能]**

设置用户为禁止或可用用户。

**[参数说明]**

**dwMachineNumber**

要操作的机器号。

**DwEnrollNumber**

用户登记号。

**DwEMachineNumber**

和 dwMachineNumber 一样，为机器号。

**DwBackupNumber**

指纹索引。

**BFlag**

用户是否可用，True 为可用，否则为禁用。

**[返回值]**

成功则返回 True,否则返回 False。

#### [例子]

```
Dim dwEnrollNumber
Dim dwEmachineNumber
Dim dwBackNumber
Dim bFlag
```

```
dwEnrollNumber = 1
dwEmachineNumber = 1
dwBackNumber = 0
bFlag = True
```

```
CZKEM1.EnableUser MACHINENUMBER, dwEnrollNumber, dwEmachineNumber, dwBackNumber,
bFlag
```

### 4.2.3 GetUserTZs

#### [函数]

```
VARIANT_BOOL GetUserTZs([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in]
LONG *TZs)
```

#### [功能]

取得用户时间段。

#### [参数说明]

**TZs**

用户开门时间段。TZs 指针中共有三个值，分别存储三个时间段。

#### [返回值]

成功则返回 True, 否则返回 False。

#### [例子]

```
Dim dwEnrollNumber
Dim tzs(2) As Long
```

```
dwEnrollNumber = 1
```

```
CZKEM1.GetUserTZs MachineNumber, dwEnrollNumber, tzs(0)
Debug.Print tzs(0), tzs(1), tzs(2)
```

#### 说明:

考勤机总共内置 50 个时间段，其值都默认为 00:00 到 23:59 分，以周(week)为周期。内定设置 5 个组，内定设置 10 个分组组合。组或用户可以使用定义好的时间段。组或用户可以使用三个时间段，三个时间段为或地关系。用户可以拥有自己的时间段，而用户在门禁机中必须属于某一个组，即用户和组都使用时间段，以用户时间段优先。当用户时间段设置使用组时间段或用户三个时间段设置为 0 时（空），使用组时间段。

### 4.2.4 SetUserTZs

#### [函数]

```
VARIANT_BOOL SetUserTZs([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in]
LONG *TZs)
```

#### [功能]

设置用户时间段。

#### [参数说明]

##### **dwEnrollNumber**

用户登记号。

##### **TZs**

时间段，为空的时候，使用组的设置，不为空时，则指定一个用户自己的设置。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim dwEnrollNumber
Dim tzs(2) As Long

dwEnrollNumber = 1
tzs(0) = 1
tzs(1) = 2
tzs(2) = 0
CZKEM1.SetUserTZs MachineNumber, dwEnrollNumber, tzs(0)
CZKEM1.RefreshData MachineNumber
```

## 4.2.5 GetUserGroup

#### [函数]

VARIANT\_BOOL GetUserGroup([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG \*UserGrp)

#### [功能]

取得用户所在的组。

#### [参数说明]

##### **UserGrp**

用户所在的组。

#### [例子]

```
Dim dwEnrollNumber
Dim userGrp As Long

dwEnrollNumber = 1
`取得用户 1 所在的组。
CZKEM1.GetUserGroup MACHINENUMBER, dwEnrollNumber, userGrp
```

## 4.2.6 SetUserGroup

#### [函数]

VARIANT\_BOOL SetUserGroup([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG UserGrp)

#### [功能]

设置用户属于某一个组。当进行 1:G 验证时，该函数设定用户分组对 1:G 验证也有效。当然，机器要支持 1:G 功能验证。具体请参考设备说明。

#### [参数说明]

##### **dwEnrollNumber**

用户登记号。

##### **UserGrp**

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.42

某一个组。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim dwEnrollNumber
Dim userGrp

dwEnrollNumber = 1
userGrp = 1

CZKEM1.SetUserGroup MACHINENUMBER, dwEnrollNumber, userGrp
```

### 4.2.7 GetACFun

#### [函数]

VARIANT\_BOOL GetACFun([in] LONG\* ACFun)

#### [功能]

取得开锁延迟时间

#### [参数说明]

**ACFun**

开锁时间长度。

#### [例子]

```
Dim acFun as Long

CZKEM1.GetACFun acFun
```

### 4.2.8 GetTZInfo

#### [函数]

VARIANT\_BOOL GetTZInfo([in] LONG dwMachineNumber, [in] LONG TZIndex, [in] BSTR \*TZ)

#### [功能]

取得时间段设置。

#### [参数说明]

dwMachineNumber: 指纹机编号。

TZIndex: 时间段索引。

TZ: 时间段字符串。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim tzIndex
Dim tz As String

tzIndex = 1

CZKEM1.GetTZInfo MACHINENUMBER, tzIndex, tz '可以观察一下该字符串的格式。
Debug.Print tz
```

‘如 010023590000235900002359000023590000235900002359

‘时间段中每个时间设置开始时间和结束时间。如从 09: 00 到 14: 00, 设置为 09001400, 共 8 个字符。

可以看出, 格式为 24 小时格式, 时间段开始时间为周末, 依次类推。所有每个时间段共有 8\*7 个字符组成。可根

**Stand alone development for ZKEMSDK,Reversion 6.1**

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.43

据该格式分析时间段。

#### 4.2.9 SetTZInfo

##### [函数]

VARIANT\_BOOL SetTZInfo([in] LONG dwMachineNumber, [in] LONG TZIndex, [in] BSTR TZ)

##### [功能]

给组设置时间段。

##### [参数说明]

**TZIndex**

组索引。

**TZ**

时间段，格式请参考 GetTZInfo 函数说明。

##### [返回值]

成功则返回 True,否则返回 False。

##### [例子]

```
Dim tzIndex
Dim tz

tzIndex = 1
tz = "010023590000235900002359000023590000235900002359"

CZKEM1.SetTZInfo MACHINENUMBER, tzIndex, tz
```

#### 4.2.10 GetUnlockGroups

##### [函数]

VARIANT\_BOOL GetUnlockGroups([in] LONG dwMachineNumber, [in] BSTR \*Grps)

##### [功能]

取得开锁组合信息。

##### [参数说明]

**dwMachineNumber**

机器编号，即所操作的机器。

**Grps**

开锁组信息。5 个组，10 个组合。一个组合最大可包含 5 个组每个组合以 ‘:’ 隔开

##### [返回值]

成功则返回 True,否则返回 False。

##### [例子]

```
Dim grps As String

CZKEM1.GetUnlockGroups MACHINENUMBER, grps
```

#### 4.2.11 SetUnlockGroups

##### [函数]

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.44

---

VARIANT\_BOOL SetUnlockGroups([in] LONG dwMachineNumber, [in] BSTR Grps)

**[功能]**

设置开锁组合

**[参数说明]**

**Grps**

组合开门组合字符串，需要设置 10 个组合，每个组合以 ‘:’ 隔开。

**[返回值]**

成功则返回 True, 否则返回 False。

**[例子]**

Dim grps

grps = "12:13:14:15:23:24:::"

CZKEM1.SetUnlockGroups MACHINENUMBER, grps

‘如格式 12:13:14:15:23:24:::’ 如果没有设置组合，‘:’ 后面为无字符。大体含义为：1，2 组组合，1，3 组组合。1，4 组组合。1，5 组组合。2，3 组组合。2，4 组组合。后面的组合为空，即没有设置组合。

## 4.2.12 GetGroupTZs

**[函数]**

VARIANT\_BOOL GetGroupTZs([in] LONG dwMachineNumber, [in] LONG GroupIndex, [in] LONG \*TZs)

**[功能]**

取得组的时间段。

**[参数说明]**

**GroupIndex**

组索引。

**TZs**

组的时间段。三个时间段。

**[返回值]**

成功则返回 True, 否则返回 False。

**[例子]**

Dim groupIndex

Dim tzs(2) As Long

groupIndex = 1

CZKEM1.GetGroupTZs MACHINENUMBER, groupIndex, tzs

## 4.2.13 SetGroupTZStr

**[函数]**

VARIANT\_BOOL SetGroupTZs([in] LONG dwMachineNumber, [in] LONG GroupIndex, [in] BSTR \*TZs)

**[功能]**

以字符串方式取得组使用的时间段。

**[参数说明]**

### **GroupIndex**

组索引。

### **TZs**

时间段字符串，没有个时间段一 ‘:’ 隔开。

#### **[返回值]**

成功则返回 True,否则返回 False。

#### **[例子]**

```
Dim groupIndex
Dim tzs

groupIndex = 1
tzs = "1:2:3" `组 1 使用时间 1, 2, 3。

CZKEM1.SetGroupTZs MACHINENUMBER, groupIndex, tzs
```

## **4.2.14 GetGroupTZStr**

#### **[函数]**

VARIANT\_BOOL GetGroupTZStr([in] LONG dwMachineNumber, [in] LONG GroupIndex, [in] BSTR \*TZs)

#### **[功能]**

取得组的时间段。

#### **[参数说明]**

### **GroupIndex**

组索引。

### **TZs**

组使用时间段，以 ‘:’ 隔开。

#### **[返回值]**

成功则返回 True,否则返回 False。

#### **[例子]**

```
Dim groupIndex
Dim tzs As String

groupIndex = 1

CZKEM1.GetGroupTZStr MACHINENUMBER, groupIndex, tzs
```

## **4.2.15 SetGroupTZStr**

#### **[函数]**

VARIANT\_BOOL SetGroupTZStr([in] LONG dwMachineNumber, [in] LONG GroupIndex, [in] BSTR TZs)

#### **[功能]**

设置组时间段。

#### **[参数说明]**

参阅 GetGroupTZStr, TZs 为时间段字符串，以 ‘:’ 隔开。

#### **[返回值]**

成功则返回 True,否则返回 False。

#### [例子]

```
Dim groupIndex
Dim tzs

groupIndex = 1
tzs = "1:2:3"

CZKEM1.SetGroupTZStr MACHINENUMBER, groupIndex, tzs
```

### 4.2.16 GetUserTZStr

#### [函数]

VARIANT\_BOOL GetUserTZStr([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] BSTR \*TZs)

#### [功能]

取得用户时间段。

#### [参数说明]

**dwEnrollNumber**

用户登记号。

**TZs**

用户开锁时间段。其格式为：

彩屏门禁：X1:X2:X3:X4, X1=1 表示是否使组自定义时间段，为空表示使用组时间段。X2、X3、X4 表示使用的时间段编号。例如用户 A 使用自定义的时间段 1、2，那么指纹机返回的值为是："1:1:2:"

黑白屏门禁：X1:X2:X3, X1、X2、X3 表示使用自定义的时间段编号。判断用户是否使用组时间段，请用 UseGroupTimeZone 函数的返回值来判断。例如用户 A 使用自定义的时间段 1、2、3，那么指纹机返回的值为是："1:2:3"。

#### [例子]

```
Dim dwEnrollNumber
Dim tzs As String

dwEnrollNumber = 1
CZKEM1.GetUserTZs machineNumber, dwEnrollNumber, tzs
```

### 4.2.17 SetUserTZStr

#### [函数]

VARIANT\_BOOL SetUserTZStr([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] BSTR TZs)

#### [功能]

设置用户时间段。TZs 每个时间段以 ‘:’ 隔开。

#### [参数说明]

参阅 SetUserTZ

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim dwEnrollNumber As Integer
```



```
Dim tzs

dwEnrollNumber = 1
tzs = "1:2:3" ' 可以从 50 个时间段中任意选择某一时间段给用户。

'Retrun true success else false
CZKEM1.SetUserTZStr MACHINENUMBER, dwEnrollNumber, tzs

//彩屏门禁示例
Dim MachineNumber As Integer
Dim strTzs As String
Dim UserNo As Integer

MachineNumber = 1
UserNo = 1
strTzs = "1:2:3:1"//设置用户使用自定义时间段 1、2、3。
//strTzs = "1:2:3:0"//设置用户的自定义时间段为 1、2、3，但该用户使用组时间段，不使用自定义的
//时间段
Czkem1.SetUserTZStr(MachineNumber, UserNo, strTzs)
```

## 4.2.18 GetDoorState

### [函数]

```
GetDoorState(LONG MachineNumber, [in,out] LONG* State, [out,retval] VARIANT_BOOL*
pVal)
```

### [功能]

获取门的状态。

### [参数说明]

State: 门状态。1: 门开。0: 门关。

### [返回值]

成功则返回 True, 否则返回 False。

### [例子]

```
Dim State As Long
CZKEM1.GetDoorState MACHINENUMBER, State
```

## 4.2.19 UseGroupTimeZone

### [函数]

```
UseGroupTimeZone(void)
```

### [功能]

判断用户是不是使用组时间。此功能必须和 GetUserTZs 函数或者 GetGroupTZStr 一起使用其返回值才正确。

### [返回值]

使用组时间则返回 True, 否则返回 False。

### [例子]

```
Dim UseGroupFlag As Boolean
Dim MachineNumber As Long
Dim GroupIndex As Long
Dim strTime As String
```

```
MachineNumber = 1
GroupIndex = 1
CZKEM1.GetGroupTZStr(MachineNumber, GroupIndex, strTime)
UseGroupFlag = CZKEM1.UseGroupTimeZone()
```

## 4.3 机器设置

### 4.3.1 Beep

**[函数]**

```
VARIANT_BOOL Beep([in] LONG DelayMS)
```

**[功能]**

使机器响铃。

**[参数说明]**

**DelayMS**

响铃延长时间。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim delayMs As Integer 'Delay time

delayMs = 150

CZKEM1.Beep delayMs
```

### 4.3.2 ClearLCD

**[函数]**

```
VARIANT_BOOL ClearLCD()
```

**[功能]**

清屏，清除屏幕上的字符。当然，清除屏幕后，秒点(‘:’)不能清除，可以用函数 EnableClock 停止或启动其闪烁，以便写入要写得字符。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
CZKEM1.ClearLCD 'Clear scr
```

### 4.3.3 Connect\_Com

**[函数]**

```
VARIANT_BOOL Connect_Com([in] LONG ComPort, [in] LONG MachineNumber, [in] LONG
BaudRate)
```

**[功能]**

从 PC 机串口连接机器。232 和 485 连接均使用该函数。

#### [参数说明]

##### **ComPort**

串口号。

##### **MachineNumber**

机器号。

##### **BaudRate**

波特率。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
CZKEM1.Connect_Com MACHINECOM, MACHINENUMBER, RATE 'Connect with machine
CZKEM1.Beep (150) 'Make machine beep
CZKEM1.EnableDevice MACHINENUMBER, False 'Make machine in working state
```

### 4.3.4 Connect\_Net

#### [函数]

```
VARIANT_BOOL Connect_Net([in] BSTR IPAdd, [in] LONG Port)
```

#### [功能]

指定 IP 地址连接机器，即网络联机机器。

#### [参数说明]

##### **IPAdd**

机器的 IP 地址，通过机器键盘可以设置，或接口函数 SetDeviceIP。

##### **Port**

端口号，一般为 4370。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim ipAdd
Dim port As Integer
Dim flag As Boolean

ipAdd = "192.168.1.159"
port = 4370

flag = CZKEM1.Connect_Net(ipAdd, port)
If flag = True Then
    CZKEM1.EnableDevice MACHINENUMBER, False
End If
```

连接函数执行后，返回错误号表示意义如下：

连接时错误代码的意义：

- 1 加载通讯支持库失败。
- 2 bind 端口失败。
- 5 创建读线程失败。
- 6 连接未通过认证，有可能有密码。
- 7 命令未得到响应。

### 4.3.5 DisableDeviceWithTimeOut

**[函数]**

VARIANT\_BOOL DisableDeviceWithTimeOut([in] LONG dwMachineNumber,[in] LONG TimeOutSec)

**[功能]**

设置机器处于工作状态下的时间长度(超时时间长度)。

**[参数说明]**

**TimeOutSec**

处于工作状态下的时间长度，单位为秒。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim timeOutSec As Integer
timeOutSec = 1200
```

```
CZKEM1.DisableDeviceWithTimeOut MACHINENUMBER, timeOutSec
```

**说明:**

在连接机器后，为了提高通讯效率，保持数据一致或避免出错的可能，使机器处于工作状态；当设置超时时长到达时，机器恢复到正常状态，取消 ”工作中...” 状态。

### 4.3.6 Disconnect

**[函数]**

Disconnect(void)

**[功能]**

断开连接，SDK 控件自动释放资源。

**[例子]**

```
CZKEM1.EnableDevice MACHINENUMBER, True
CZKEM1.Beep (150) 'Make machine beep
CZKEM1.Disconnect 'Disconnect with machine
```

### 4.3.7 EnableDevice

**[函数]**

EnableDevice ([in] LONG dwMachineNumber, [in] VARIANT\_BOOL Enabled)

**[功能]**

使得机器处于工作状态或非工作状态。

**[参数说明]**

**dwMachineNumber**

所操作的机器号。

**Enabled**

如果为 True,则处于非工作状态。否则处于工作状态，屏蔽外围键盘、指纹采集器等。

**[例子]**

```
CZKEM1.EnableDevice MACHINENUMBER, True
```

### 4.3.8 EnableClock

**[函数]**

VARIANT\_BOOL EnableClock([in] LONG Enabled)

**[功能]**

使得机器时钟停止或运行，在液晶屏显示为 ‘:’ 不会闪动。

**[参数说明]**

**Enabled**

如果为 True,则时钟处于工作状态，否则处于停滞（不闪烁）状态。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim enabled As Boolean
enabled = False

CZKEM1.EnableClock enabled
```

### 4.3.9 GetDeviceStatus

**[函数]**

VARIANT\_BOOL GetDeviceStatus([in] LONG dwMachineNumber, [in] LONG dwStatus, [in] LONG\* dwValue)

**[功能]**

取得机器的存储信息，如管理员个数，登记用户个数，模板个数等。

**[参数说明]**

**dwMachineNumber**

所操作的机器号。

**DwStatus**

要取得的机器状态种类，描述如下：

数值	描述
1	管理员数量。
2	登记用户数量。
3	指纹模板数量。
4	密码数量。
5	管理员进行管理的记录次数。
6	考勤记录次数。
7	指纹容量。
8	用户容量。
9	记录容量。

**DwValue**

取得的由 DwStatus 所描述的值。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim dwStatus As Integer
Dim dwValue As Integer

dwStatu = 1 'Count of administrators

CZKEM1.GetDeviceStatus MACHINENUMBER, dwStatus, dwValue
```

### 4.3.10 GetDeviceInfo

#### [函数]

VARIANT\_BOOL GetDeviceInfo([in] LONG dwMachineNumber, [in] LONG dwInfo, [in] LONG\* dwValue)

#### [功能]

取得机器设置信息，如语言设置、最大记录设置等。

#### [参数说明]

##### **dwMachineNumber**

所要操作的机器号。

##### **DwInfo**

要取得的机器设置信息的类型，描述如下：

值 描述

1. 最大管理员数量；
2. 机器号
3. 语言
4. 自动关闭机器的时间长度，单位为分钟。
5. 门锁信号控制输出。

值 描述

- 0 输出门锁控制信号。
- 1 不输出门锁控制信号。

6. 考勤记录最大数量警告。
7. 管理记录最大数量警告。
8. 两个考勤记录之间的最小间隔。
9. 波特率

值 描述

- 0 1200 bps
- 1 2400 bps
- 2 4800 bps
- 3 9600 bps
- 4 19200 bps
- 5 38400 bps
- 6 57600 bps
- 7 115200 bps

10. 奇偶较验

值 描述

- 0 无较验
- 1 偶较验
- 2 奇较验

11. 停止标志位  
值 描述  
0 1 Bit  
1 2 Bits
12. 数据分割符。
- 13.是否网络连接。
- 14.是否RS232通讯
- 15.是否RS485通讯
- 16.是否语音提示
- 17.验证速度。
- 18.空闲时间。
- 19.关闭机器时间。
- 20.开机时间
- 21.睡眠时间
- 22.自动响铃
- 23.Match threshold(比对阈值)
- 24.匹配伐值
- 25.1:1 匹配伐值
- 26.是否显示分数。
- 27 开锁人组合数量。
- 28.是否为刷卡验证
- 29.网络速度
- 30.必须登记卡号
- 31.机器临时状态保持时间。
- 32.输入数字保持时间
- 33.菜单保持时间
- 34.时间格式
- 35.是否为1: 1对比

#### [例子]

```
Dim dwInfo As Integer
Dim dwValue As Integer
dwInfo =1

CZKEM1.GetDeviceInfo MACHINENUMBER, dwInfo, dwValue
```

### 4.3.11 GetWiegandDefine

#### [函数]

VARIANT\_BOOL GetWiegandDefine([in] LONG dwMachineNumber, [in] BSTR \*sWiegandDefine)

#### [功能]

取得 Wiegand 格式。门禁机 Weigend 输出由机器编号+用户编号组成

#### [参数说明]

**dwMachineNumber**

要操作的机器号。

**SWiegandDefine**

Wiegand 格式。

**[例子]**

```
Dim sWiegandDefine As String
'weigend 26,p 校验位, E 机器号所占的位, O 用户编号所占的位。
'sWiegandDefine= "PEEEEEEEEEEEEEEOOOOOOOOOOOOP"

CZKEM1.GetWiegandDefine MACHINENUMBER, sWiegandDefine
```

### 4.3.12 GetDeviceIP

**[函数]**

```
VARIANT_BOOL GetDeviceIP([in] LONG dwMachineNumber, [in] BSTR *IPAddr)
```

**[功能]**

取得机器的 IP 地址。

**[参数说明]**

**IPAddr**

IP 地址。

**[例子]**

```
Dim ipAddr As String

CZKEM1.GetDeviceIP MACHINENUMBER, ipAddr
```

### 4.3.13 SetDeviceIP

**[函数]**

```
VARIANT_BOOL SetDeviceIP([in] LONG dwMachineNumber, [in] BSTR IPAddr)
```

**[功能]**

设置机器 IP 地址。

**[参数说明]**

请参阅 GetDeviceIP

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim ipAddr
Dim flag As Boolean

ipAddr = "192.168.100.173"

flag = CZKEM1.SetDeviceIP(MACHINENUMBER, ipAddr)
```

### 4.3.14 GetDeviceMAC

**[函数]**

```
VARIANT_BOOL GetDeviceMAC([in] LONG dwMachineNumber, [in] BSTR *sMAC)
```

**[功能]**

取得机器的 MAC 地址。

**[参数说明]**



## sMAC

MAC 地址。

### [返回值]

成功则返回 True,否则返回 False。

### [例子]

```
Dim sMac As String

CZKEM1.GetDeviceMAC MACHINENUMBER, sMac
```

## 4.3.15 GetDeviceTime

### [函数]

VARIANT\_BOOL GetDeviceTime([in] LONG dwMachineNumber, [in] LONG\* dwYear, [in] LONG\* dwMonth, [in] LONG\* dwDay, [in] LONG\* dwHour, [in] LONG\* dwMinute, [in] LONG\* dwSecond)

### [功能]

取得机器时间。

### [参数说明]

**dwYear**、**dwMonth**、**dwDay**、**dwHour**、**dwMinute**、**dwSecond**，年月日时分秒。

### [返回值]

成功则返回 True,否则返回 False。

### [例子]

```
Dim dwYear As Long
Dim dwMonth As Long
Dim dwHour As Long
Dim dwDay As Long
Dim dwMinute As Long
Dim dwSecond As Long

CZKEM1.GetDeviceTime MACHINENUMBER, dwYear, dwMonth, dwDay, dwHour, dwMinute,
dwSecond 'Get machine time
```

## 4.3.16 GetSerialNumber

### [函数]

VARIANT\_BOOL GetSerialNumber([in] LONG dwMachineNumber, BSTR\* dwSerialNumber)

### [功能]

取得产品序列号。

### [参数说明]

**dwMachineNumber**

要操作的机器。

**dwSerialNumber**

产品序列号。

### [返回值]

成功则返回 True,否则返回 False。

### [例子]

```
Dim dwSerialNumber As String
```

---

CZKEM1.GetSerialNumber MACHINENUMBER, dwSerialNumber

### 4.3.17 GetProductCode

**[函数]**

VARIANT\_BOOL GetProductCode([in] LONG dwMachineNumber, BSTR\* lpszProductCode)

**[功能]**

取得产品型号。

**[参数说明]**

**lpszProductCode**

产品型号。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim ipsxProductCode As String
```

```
CZKEM1.GetProductCode MACHINENUMBER, ipsxProductCode
```

### 4.3.18 GetFirmwareVersion

**[函数]**

VARIANT\_BOOL GetFirmwareVersion([in] LONG dwMachineNumber, [in] BSTR\* strVersion)

**[功能]**

得到固件版本号。

**[参数说明]**

**strVersion**

固件版本号。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim strVersion As String
```

```
CZKEM1.GetFirmwareVersion MACHINENUMBER, strVersion
```

### 4.3.19 GetSDKVersion

**[函数]**

VARIANT\_BOOL GetSDKVersion([in] BSTR\* strVersion)

**[功能]**

得到脱机 SDK 版本号。

**[参数说明]**

**strVersion**

SDK 版本号。

**[返回值]**

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.57

成功则返回 True,否则返回 False。

**[例子]**

```
Dim strVersion As String

CZKEM1.GetSDKVersion strVersion
```

### 4.3.20 PowerOnAllDevice

**[函数]**

```
PowerOnAllDevice()
```

**[功能]**

启动所有的机器。该函数无效。

**[返回值]**

成功则返回 True,否则返回 False。

### 4.3.21 PowerOffDevice

**[函数]**

```
VARIANT_BOOL PowerOffDevice([in] LONG dwMachineNumber)
```

**[功能]**

关闭机器。

**[参数说明]**

**dwMachineNumber**

机器号。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
CZKEM1.PowerOffDevice MACHINENUMBER
```

### 4.3.22 PlayVoiceByIndex

**[函数]**

```
VARIANT_BOOL PlayVoiceByIndex([in] LONG Index)
```

**[功能]**

根据索引播放某一声音。

**[参数说明]**

**Index**

声音索引。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim i As Integer

i = 1
```

### 4.3.23 QueryState

**[函数]**

VARIANT\_BOOL QueryState([in] LONG \*State)

**[功能]**

取得机器的状态。

**[返回值]**

成功则返回 True,否则返回 False。

**[参数说明]**

**State**

状态值。

**[返回值]**

成功则返回 True,否则返回 False。

值	描述
0	等待状态;
1	登记指纹状态;
2	识别指纹状态;
3	执行人机界面菜单
4	正忙于处理其他工作
5	等待写卡状态

**[例子]**

```
Dim state As Integer

CZKEM1.QueryState state
```

### 4.3.24 RestartDevice

**[函数]**

VARIANT\_BOOL RestartDevice([in] LONG dwMachineNumber)

**[功能]**

重新启动机器。

**[参数说明]**

**dwMachineNumber**

机器号。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
CZKEM1.RestartDevice MACHINENUMBER
```

### 4.3.25 SetDeviceInfo

**[函数]**

VARIANT\_BOOL SetDeviceInfo([in] LONG dwMachineNumber, [in] LONG dwInfo, [in] LONG dwValue)

**[功能]**

设置机器参数。

**[参数说明]**

请参阅 GetDeviceInfo,和该函数功能对称。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim dwInfo
Dim dwValue

dwInfo = 35
dwValue = 20

CZKEM1.SetDeviceInfo MACHINENUMBER, dwInfo, dwValue
```

### 4.3.26 SetDeviceTime

**[函数]**

VARIANT\_BOOL SetDeviceTime([in] LONG dwMachineNumber)

**[功能]**

设置机器\终端时间为 PC 的时间。

**[参数说明]**

**dwMachineNumber**

所要操作的机器号。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
CZKEM1.SetDeviceTime MACHINENUMBER 'Set machine time with sys
```

**注意:** SetDeviceTime2 设置用户自定义时间。

### 4.3.27 SetDeviceTime2

**[函数]**

VARIANT\_BOOL SetDeviceTime2([in] LONG dwMachineNumber, [in] LONG dwYear, [in] LONG dwMonth, [in] LONG dwDay, [in] LONG dwHour, [in] LONG dwMinute, [in] LONG dwSecond)

**[功能]**

设置的时间, 和 SetDeviceTime 不同在于本函数可以指定本地时间。

**[参数说明]**

分别为机器号、年、月、日、时、分、秒

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim dwYear
Dim dwMonth
Dim dwDay
```

```
Dim dwHour
Dim dwMinute
Dim dwSecond
```

```
dwYear = 2008
dwMonth = 8
dwDay = 8
dwHour = 8
dwMinute = 8
dwSecond = 8
```

```
CZKEM1.SetDeviceTime2 MACHINENUMBER, dwYear, dwMonth, dwDay, dwHour, dwMinute,
dwSecond
```

#### 4.3.28 SetDeviceMAC

##### [函数]

```
VARIANT_BOOL SetDeviceMAC([in] LONG dwMachineNumber, [in] BSTR sMAC)
```

##### [功能]

设置机器 MAC 值。

##### [参数说明]

请参阅 GetDeviceMAC

##### [返回值]

成功则返回 True,否则返回 False。

##### [例子]

```
Dim sMAC

SMAC = "00:50:54:00:0C:FC"

CZKEM1.SetDeviceMAC MACHINENUMBER, sMAC
```

#### 4.3.29 SetWiegandDefine

##### [函数]

```
VARIANT_BOOL SetWiegandDefine([in] LONG dwMachineNumber, [in] BSTR sWiegandDefine)
```

##### [功能]

设置机器 Wiegand 值。

##### [参数说明]

请参阅 GetWiegandDefine

##### [返回值]

成功则返回 True,否则返回 False。

##### [例子]

```
Dim sWiegandDefine

sWiegandDefine = "PEEEEEEEEEEEEEEOOOOOOOOOOOP"

CZKEM1.SetWiegandDefine MACHINENUMBER, sWiegandDefine
```

### 4.3.30 SetCommPassword

**[函数]**

VARIANT\_BOOL SetCommPassword([in] LONG CommKey)

**[功能]**

设置连接密码。

**[参数说明]**

**CommKey**

连接密码。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim commkey

commkey = 1234

CZKEM1.SetCommPassword commkey
```

### 4.3.31 UpdateFirmware

**[函数]**

VARIANT\_BOOL UpdateFirmware([in] BSTR FirmwareFile)

**[功能]**

升级固件文件。

**[参数说明]**

**FirmwareFile**

固件名称, 包含完整路径。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

```
Dim firmwareFile

firmwareFile = "c:\emfw.cfg.cfg"

CZKEM1.UpdateFirmware firmwareFile
```

### 4.3.32 WriteLCD

**[函数]**

VARIANT\_BOOL WriteLCD([in] LONG Row, LONG Col, BSTR Text)

**[功能]**

在液晶屏显示信息。

**[参数说明]**

**Row**

开始的行。

**Col**

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.62

开始的列。

## BSTR

显示的内容。

### [返回值]

成功则返回 True,否则返回 False。

### [例子]

```
Dim row As Integer
Dim col As Integer
Dim text

row = 1
col = 1
text = "Hello Beetfuxi pei"

CZKEM1.WriteLCD row, col, text
```

## 4.4 其他

### 4.4.1 CaptureImage

#### [函数]

VARIANT\_BOOL CaptureImage([in] VARIANT\_BOOL FullImage, [in] LONG \*Width, [in] LONG \*Height, [in] BYTE \*Image,[in] BSTR ImageFile)

#### [功能]

抓取脱机指纹机当前的指纹图像。

#### [参数说明]

##### FullImage

如果为 True, 则返回指纹图像全部, 否则返回指纹特征部分。

##### Width

指定图像的宽度, 由于获得图像是定比例图像, 故只需要指定宽度即可。

##### Height

指定图像的高度

##### Image

存放所采集来的图像, 以二进制形式存放于该变量中。存储位图全部数据。

##### ImageFile

以文件形式保存所采集的指纹, 可以包含文件路径, 如 c:\ABMP.bmp。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim fullImage
Dim width As Integer
Dim height As Integer
Dim image(1024 * 8) As Byte
Dim imageFile
fullImage = False
width = 88
height = 88
imageFile = "c:\fp1.bmp"
```



---

CZKEM1.CaptureImage fullImage, width, height, image(0), imageFile

**[说明]**

该函数只支持 ZEM100 系列机器，如 A1,A2,A3,F4,F7 等。

#### 4.4.2 CancelOperation

**[函数]**

VARIANT\_BOOL CancelOperation() Read only

**[功能]**

取消当前操作，如正在登记用户，在该函数执行时，则取消登记。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

CZKEM1.CancelOperation

#### 4.4.3 GetLastError

**[函数]**

GetLastError([in] LONG\* dwErrorCode)

**[功能]**

获取错误信息。

**[参数说明]**

**dwErrorCode**

取得的错误号，具体描述如下。

值	描述
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

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

Dim errorCode As Long

CZKEM1.GetLastError errorCode

#### 4.4.4 StartVerify

**[函数]**

VARIANT\_BOOL StartVerify([in] LONG UserID, [in] LONG FingerID)

**[功能]**

开始 1:1 比对指纹。

**[参数说明]**

**UserID**

用户登记号。

**FingerID**

指纹索引。

**[例子]**

```
Dim UserID As Integer
Dim fingerID As Integer

UserID = 1
fingerID = 1

CZKEM1.StartVerify UserID, fingered
```

### 4.4.5 StartEnroll

**[函数]**

VARIANT\_BOOL StartEnroll([in] LONG UserID, [in] LONG FingerID)

**[功能]**

开始登记用户。

**[参数说明]**

请参阅 StartVerify 函数。

**[例子]**

```
Dim userID As Integer
Dim fingerID As Integer

userID = 6
fingerID = 0

CZKEM1.StartEnroll userID, fingered
```

### 4.4.6 StartIdentify

**[函数]**

VARIANT\_BOOL StartIdentify()

**[功能]**

开始 1: N 比对。

**[返回值]**

成功则返回 True, 否则返回 False。

**[例子]**

```
CZKEM1.StartIdentify
```

### 4.4.7 GetSensorSN

**[函数]**

```
GetSensorSN([in] LONG dwMachineNumber, [in,out] BSTR* SensorSN, [out,retval]
VARIANT_BOOL* pVal)
```

#### [功能]

获得指纹采集器序列号。ZEM200 系列采用 URU 指纹仪机器可支持该功能。

#### [参数说明]

SensorSN: 指纹采集器序列号。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim SensorSN As String
CZKEM1.GetSensorSN MACHINENUMBER,SensorSN
```

### 4.4.8 WriteCard

#### [函数]

```
WriteCard([in] LONG dwMachineNumber, [in] LONG dwEnrollNumber, [in] LONG
dwFingerIndex1, [in] BYTE* TmpData1, [in] LONG dwFingerIndex2, [in] BYTE* TmpData2, [in]
LONG dwFingerIndex3, [in] BYTE* TmpData3, [in] LONG dwFingerIndex4, [in] BYTE* TmpData4,
[out,retval] VARIANT_BOOL* pVal)
```

#### [功能]

通知机器写 Mifare card.将某个用户的某几枚指纹模板写入 Mifare,当该命令执行成功后,机器 LCD 提示用户刷卡。

#### [参数说明]

dwMachineNumber: 机器编号。

dwEnrollNumber: 用户编号。

dwFingerIndex1: 指纹索引 1。

TmpData1: 指纹模板 2。

参数传入内容为某个用户的某几枚指纹模板, TmpData1 不能为空, 即第一枚指纹不能为空。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim BWrite As Boolean
Dim UserID
Dim temp1() As Byte 'template 1
Dim temp2() As Byte 'template 2
Dim temp3() As Byte 'template 3
Dim temp4() As Byte 'template 4
UserID=1
CZKEM1.CancelOperation `在有的固件版本中, 不需要执行该句, 请注意。
`需要给指纹模板赋值。
BWrite=CZKEM1.WriteCard(MACHINENUMBER,UserID, 0, temp1(0), _
                        1, temp2(0), _
                        2, temp3(0), _
                        3, temp4(0))
```

CZKEM1.StartIdentify 恢复机器到默认验证状态。

#### 4.4.9 EmptyCard

##### [函数]

EmptyCard([in]LONG dwMachineNumber,[out,retval] VARIANT\_BOOL\* pVal)

##### [功能]

清空 Mifare 卡。

##### [例子]

CZKEM1.EmptyCard MACHINENUMBER

## 5. 事件

SDK 中有较丰富的事件实时反映机器状态,如验证通过时、报警、按键等。

### 5.1 OnAttTransaction

事件原型如下:

void OnAttTransaction([in] LONG EnrollNumber, [in] LONG IsInvalid, [in] LONG AttState, [in] LONG VerifyMethod, [in] LONG Year, [in] LONG Month, [in] LONG Day, [in] LONG Hour, [in] LONG Minute, [in] LONG Second)

功能:

当指纹验证通时,触发该事件。

变量含义:

EnrollNumber:用户编号

IsInvalid: 0 记录无效, 1 记录有效。在指纹门禁机组拒绝或时间段原因拒绝开门时, 该变量会返回无效值。

AttState:考勤状态, 表示 Checkin checkOut 等, 值的范围为 0-5。超出无效。

VerifyMethod:比对方式, 0, 密码。1, 指纹验证。

Year, Month, Day, Hour, Minute, Second:年月日时分秒。

### 5.2 OnFinger

事件原型如下:

void OnFinger(void);

功能:

当按指纹时,触发该事件。

### 5.3 OnNewUser

事件原型如下:

void OnNewUser([in] LONG EnrollNumber);

功能:

当按指纹时,触发该事件。

变量含义:

EnrollNumber: 用户编号。

### 5.4 OnEnrollFinger

事件原型如下:

void OnEnrollFinger([in] LONG EnrollNumber, [in] LONG FingerIndex, [in] LONG ActionResult, [in] LONG

TemplateLength);

**功能:**

当登记指纹成功时, 触发该事件。

**变量含义:**

EnrollNumber: 用户编号。

FingerIndex: 指纹索引。

ActionResult: 0, 登记正常。3, 保存数据失败。4, 指纹登记失败。5, 指纹重复。6, 操作被取消。

TemplateLength: 指纹模板长度。

## 5.5 OnKeyPress

**事件原型如下:**

```
void OnKeyPress(LONG Key);
```

**功能:**

当按键盘时, 触发该事件。

**变量含义:**

Key: 键盘值。

## 5.6 OnVerify

**事件原型如下:**

```
HRESULT OnVerify([in] LONG UserID);
```

**功能:**

当指纹验证通过时, 触发该事件。

**变量含义:**

UserID: 用户编号。如果 UserID 的值小于 0, 表示指纹机没有所验的指纹, 指纹验证失败。

## 4.7 OnFingerFeature

**事件原型如下:**

```
HRESULT OnFingerFeature([in] LONG Score);
```

**功能:**

当指纹验证时, 触发该事件。

**变量含义:**

Score:: 验证指纹后指纹匹配分数。

## 5.8 OnAlarm

**事件原型如下:**

```
HRESULT OnAlarm([in] LONG AlarmType,[in] LONG EnrollNumber,[in] LONG Verified);
```

**功能:**

在发生拆机或胁迫报警时触发。

**变量含义:**

AlarmType: 报警类型。值为 55: 拆机报警; 58 为错按报警。其它值为胁迫报警。

EnrollNumber: 为 0, 在该处无效。

Verified: 为 0, 在该处无效。

## 5.9 OnHIDNum

**事件原型如下:**

```
HRESULT OnHIDNum([in] LONG CardNumber);
```

**功能:**

在刷卡时 (号码卡) 时, 触发该事件。

**变量含义:**

CardNumber: 卡号。

## 5.10 OnWriteCard

事件原型如下：

```
HRESULT OnWriteCard([in] LONG EnrollNumber, [in] LONG ActionResult, [in] LONG Length);
```

功能：

当写 Mifare card 成功或失败时触发该函数。

变量含义：

EnrollNumber：用户编号。

ActionResult：写入成功，返回 0，失败为负值。

Length：写入指纹模板长度。

## 5.11 OnEmptyCard

事件原型如下：

```
HRESULT OnEmptyCard([in] LONG ActionResult);
```

功能：

在清空卡（Mifare card）时触发该事件。

变量含义：

ActionResult:返回 0 时清卡成功。

## 5.12 OnAttTransactionEx

事件原型如下：

```
void OnAttTransactionEx([in] BSTR EnrollNumber, [in] LONG IsValid, [in] LONG AttState, [in] LONG VerifyMethod,
[in] LONG Year, [in] LONG Month, [in] LONG Day, [in] LONG Hour, [in] LONG Minute, [in] LONG Second, [in] LONG
WorkCode);
```

功能：

当指纹验证通过时，触发该事件。此功能只在支持 Work Code 功能的机器上并且机器的固件版为 6.0 以上有效。此事件支持 Work Code 功能的同时也支持彩屏门禁机器。

变量含义：

EnrollNumber:用户编号

IsValid: 0 记录无效，1 记录有效。在指纹门禁机组拒绝或时间段原因拒绝开门时，该变量会返回无效值。

AttState:考勤状态，表示 Checkin checkOut 等，值的范围为 0-5。超出无效。

VerifyMethod:比对方式。0 密码，1 指纹验证。对于彩屏指纹门禁机系列，该变量返回值是 0-14。

0(FP/PW/RF), 1(FP), 2(PIN), 3(PW), 4(RF), 5(FP&RF), 6(FP/PW), 7(FP/RF), 8(PW/RF), 9(PIN&FP), 10(FP&PW), 11(PW&RF), 12(FP&PW&RF), 13(PIN&FP&PW), 14(FP&RF/PIN)。

Year, Month, Day, Hour, Minute, Second:年月日时分秒。

WorkCode: 工作编号。

## 5.13 ReadRTLog

事件原型如下：

```
HRESULT ReadRTLog([in] LONG dwMachineNumber);
```

功能：

从机器获取实时事件，等待使用 GetRTLog 函数触发实时事件。如：OnAttTransaction、OnVerify 等等。此功能必须和 GetRTLog 函数配合使用。在 SDK 的 6.0 版本以前，对于实时事件，机器会主动发给 SDK，这种触发机制会造成通讯数据错误或者丢失。为了改善这种情况，在 SDK6.0 版本以后，机器不会主动发出。如果要想必须对实时事件的监控，必须通过 ReadRTLog 函数和 GetRTLog 函数配合使用才能触发实时事件。

对于此功能支持的机器的固件版本必须是 6.0 以上的版本。

变量含义：

dwMachineNumber: 机器号

例子：

```
Dim MachineNumber As Long
MachineNumber = 1
If CZKEM1.ReadRTLog(MachineNumber) then
    While CZKEM1.GetRTLog(MachineNumber)
        Wend
    End If
```

当调用 GetRTLog 函数后，SDK 就会触发实时事件。

### 5.14 GetRTLog

事件原型如下：

```
HRESULT GetRTLog(LONG dwMachineNumber);
```

功能：

触发实时事件。在触发前必须使用 ReadRTLog 函数获取实时事件。此事件必须与 ReadRTLog 配合使用。

变量含义：

dwMachineNumber：机器号

### 5.15 OnDoor

事件原型如下：

```
HRESULT OnDoor([in] LONG EventType);
```

功能：

门磁事件

变量含义：

EventType: 4 表示门未关好或者门已打开,53 表示出门按钮,5 表示门已关闭,1 表示门被意外打开

## 6. 属性

### 6.1 AccGroup

功能：设置或获取用户所属的组。

在上传用户前如果设置了该属性，则在 SetUserInfo 等函数上传用户时，设置该用户所属的组，否则默认为 1 组。

类型：LONG，可读写。

### 6.2 AccTimeZones

功能：设置用户使用时间段。

在上传用户前如果设置了该属性，则在 SetUserInfo 等函数上传用户时，设置该用户所属的时间段。

类型：LONG，可读写。AccTimeZones[1]、AccTimeZones[2]、AccTimeZones[3] 分别表示设置或者读取时间段 1、时间段 2、时间段 3 的时间段编号值。AccTimeZones[4] 只对于彩屏门禁起作用，如果该值为 0 表示用户使用组时间段，该值为 1 表示使用自定义的时间段。

### 6.3 BASE64

功能：当设置该属性为真值时，开发包在输出字符串模板时输出为 Base64 编码，否则是十六进制编码。

类型：LONG，可读写。

## 6.4 CardNumber

**功能：**设置或读取用户的卡号。

**类型：**LONG, 可读写。

## 6.5 CommPort

**功能：**设置串口或 485 连接时端口号。

**类型：**LONG, 可读写。

## 6.6 ConvertBIG5

**功能：**当设置该属性为真值时，开发包自动把简体字符转换为繁体字符，用户繁体版脱机开发，但是在多国语言系列产品中，该函数无效，请不要设置该属性

**类型：**LONG, 可读写。

**注意：**在有多国语言系列版本以后的版本中，请不要设置该属性，该属性无效。ZEM100 5.22, ZEM200 5.30 后的版本不需要修改该属性。

## 6.7 PINWidth

**功能：**标示用户编号（阿拉伯数字）最大长度。

**类型：**LONG, 只读。

# 7. 常见问题解答

## 7.1 如何下载考勤记录

首先用 ReadGeneralLogData 函数可以把所有的考勤记录读到内存中，然后使用 GetGeneralLogData 函数循环取得考勤记录，当函数 GetGeneralLogData 返回 False 的时候，说明读取考勤记录完成，把读取到的记录写到数据库或以其他形式显示，即可完成考勤记录的下载；下载管理记录和该方式一样。

## 7.2 如何联机创建用户

首先使用函数 SetUserInfo 给机器写入用户相关记录，如登记号、密码、姓名等信息，然后用 SetUserTmpStr/SetUserTmp/SetEnrollDataStr/SetEnrollData 函数给该用户写入指纹模板；这种方法适合已经采集到用户信息，并以诸如数据库等存储了用户信息，不需要用户脱机再登记，提高登记的效率。

## 7.3 U 盘数据导入及下载

在现有的脱机产品中，如 A5 可以提供 U 盘数据下载；很多客户关心 U 盘数据格式，由于下载到的数据格式较复杂，故我们有把 U 盘数据导入到数据库中的工具软件，该数据库开放，客户可以从该数据库中读到下载数据。

U 盘数据可参考如下描述。

用户数据结构：

```
typedef struct _User_{
```

Stand alone development for ZKEMSDK, Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.71



```

    U16 PIN;
    U8 Privilege;
    char Password[5];
    char Name[8];
    U8 Card[5];      //ID No which used for store the relevant ID No
    U8 Group;        //the Group user belongs to
    U16 TimeZones;    //user can use time zone
    U32 PIN2;        //32Bit PIN2
}GCC_PACKED TUser, *PUser;

```

指纹模板数据结构:

```

typedef struct _Template_{
    U16 Size; //指纹模板长度
    U16 PIN; //用户内部编号, 可和用户表中的 PIN2 对照。
    BYTE FingerID; // 指纹备份号
    BYTE Valid;
    BYTE Template[MAXTEMPLATESIZE]; //maximize template length
}GCC_PACKED TTemplate, *PTemplate;

```

考勤记录数据结构:

attlog.dat format explanation:

segment:

BadgeNumber(employee number),  
 checktime, DeviceID,  
 checktype(check status),  
 VerifyCode(verification ways: password or fingerprint)

There is an Ascii code #9(Tab) between each segment. When development, move to the segment value you want to choose by "Tab".

## 7.4 BIOKEY 采集指纹模板写入脱机

当用 BIOKEY 采集到指纹的时候, 一般在登记的时候取得指纹模板, 如 OnEnroll 事件中就可以取得当前登记指纹模板, 在取得指纹模板后, 即可写入模板。写入过程请参阅 4.2。

## 7.5 取得所有用户所有信息

使用 ReadAllUserID 把所有的用户 ID 号读取到内存中, 然后用 GetAllUserID 循环得到用户 EnrollNumber, 接着用 GetUserInfo 函数就可得到用户信息。如果还需要得到指纹模板数据, 可用 GetUserTmpStr 函数取得字符串类型的指纹模板。

## 7.6 机器连接

在连接过程中可以把机器看成一个独立的 PC, 进行连接。但是要注意的是, 机器的 IP 地址要和所连接的 IP 地址一一对应, 在有的机器中, 如 F4 有串口连接和网络连接, 在不同的连接过程中要对机器进行不同的设置, 需

要修改通讯方式，把所带控制器开关拨为 TCP/IP 或 232/485，否则不能连接。有时候也因为串口繁忙而不能连接机器，重新启动程序即可连接；有时候因为应用软件连接机器而没有人断开，可用 `DisableDeviceWithTimeOut` 函数设置机器自动断开时间；在有的连接中，由于通过串口或网络下载、修改数据等，和数据库一样的道理，为了保持数据的一致性，避免未知错误的发生，可用 `EnableDevice` 使机器处于工作状态下，在通讯完成后再恢复。

## 7.7 在 SetUserInfo 后密码不能用问题

在执行该函数后，有可能把 Password 设置为空，所以使用密码的时候就不能验证。在 `SetUserInfo` 之前，需要用 `GetUserInfo` 得到用户密码，把密码值传输给 `SetUserInfo` 的 Password 参数，在写入用户信息的时候可保持密码不变。

## 7.8 联机模板转换为脱机模板

可以使用 `FPTempConvertNew` 将所采集到的模板转换为脱机指纹模板。如何得到 BIOKEY 采集的模板，请参阅 4.4/3.1.9 的说明，该函数对二进制的指纹模板转换。其参数 `temp1` 和 `temp2` 都为二进制类型。也可以使用函数 `FPTempConvertNewStr` 将字符串格式的 BIOKEY 指纹模板转换为脱机指纹模板。

## 7.9 Demo 程序不能接机器问题

在有的时候，用户安装了考勤管理程序，用考勤管理程序是可以连接机器的，但是使用 Demo 就不能连接机器，其原因是在安装考勤管理软件的时候，动态链接库拷贝到考勤管理程序目录下，注册于安装目录，当开发时，一般情况，开发工具于系统目录下引用控件，故如果系统目录开发包和考勤软件目录下开发包版本不一致，容易造成冲突（不同版本动态连接库函数地址不同，但是写程序时 OCX 函数接口相同，所以在运行时才能表现出）。

**注意,开发包在系统中注册的一般步骤为**

- 1, 如果系统中已经存在开发包注册（注册过开发包），请执行 `regsvr32 /u zkemkeeper.dll`，反注册旧的开发包。
- 2, 拷贝所有动态连接库到系统目录下，如 win2000 位于 `winnt\system32`。
- 3, 运行下执行 `regsvr32 “注册路径\zkemkeeper.dll”` 注册开发包。
- 4, 在开发环境下正确引用控件（请自行了解开发工具使用，这里不做介绍）。
- 5, 在开发环境或运行环境下，请尽量使用同一版本开发包。

## 7.10 脱机指纹机连接后处于工作状态问题

在和脱机指纹机进行通讯的时候，为了保持数据的一致性，避免意外情况的发生，在连接脱机指纹机后，常用函数 `EnableDevice` 使脱机指纹机处于工作状态（请查阅 `EnableDevice` 函数说明），处于工作状态后，脱机指纹机键盘、指纹读头停止正常工作，即暂时处于不可用状态，在通讯完成后，可断开连接机器或再次使用 `EnableDevice`，使脱机指纹机恢复到正常状态。

推荐使用 `DisableDeviceWithTimeOut` 函数。

## 7.11 将繁体文字写入脱机指机的时候需要注意的问题

当把繁体文字写入脱机指纹机的时候，需要修改 `ConvertBIG5` 属性为 1，如 `CZKEM1.ConvertBIG5 = 1`，以免在繁体显示的时候显示的是乱码。注意：在多国语言系列版本，如：ZEM100 Ver 5.22，ZEM200 Ver 5.30 后的版本不

需要设置该属性。

因为繁体字库的不同，在上传姓名等字符后先是乱码，请修改 PC 语言选项。

## 7.12 A5,K8 射频管理卡问题

如何在 A5 、K8 中写入、取得用户卡号？

**A5**,自身具有指纹验证功能，一般上传用户分为两步走，上传用户消息 和 指纹模板。在脱机开发包中，有一属性，名称为 `cardnumber`，在上传（创建）用户的时候，假设用户现在有一开发包对应控件为 `czkem1`，首先设置 `cardnumber[0]`=要创建用户的卡号，然后用 `setuserinfo`，就能写入用户卡号信息，当然，**K8** 没有指纹验证，固 `setuserinfo` 之后不用再上传指纹。下载用户卡号，在取得某一用户基本信息（`getuserinfo`）之后，得到 `czkem1` 的 `cardnumber[0]`值即可得到用户卡号。

注意：射频卡后有喷码(10 进制)，写入卡号时只需要喷码的后三个字节。在一般可开发中，如 **PB** 中，用 `czkem1(0)`这样来写，在编译时会出错，故请注意写成 `czkem1[0]`，具体请参照脱机开发包中 **PB** 例子。

说明：**A5,K8** 卡号在机器内部定义为无符号 4 个字节，接口属性为 **Long** 型，如在 **VB** 中不能达到无符号 4 字节，故传入卡号后三个字节也可进行验证（如果在不重复的情况下）。

## 7.13 连接穿越防火墙或路由器问题

在很多时候，要连接的机器要穿越防火墙或路由器等，开发包通信 **Socket** 也使用 **UDP** 协议、4370 端口。所以在防火墙设置或路由器中要开放 **UDP** 协议及 4370 端口。如果通过端口映射来穿越网关，可通过访问路由器等的端口+IP 访问某台机器。在一般情况下，如果开放了 **UDP** 及 4370，能 **PING** 通，即可连通。当然，下载数据时要考虑网络情况。在有的机器中，可以支持 **SOAP** 接口，可以通过机器内置 **WebServer** +**SOAP** 访问机器。

注意：**zem100** 系列产品需要端口映射穿越 **internet**，对于 **zem200**，如果本地网络环境可以支持网关通信，由于其运行于 **linux** 上，设置网关既可访问到设备。当然还有其他一些方法访问到设备，如：**VPN**(虚拟局域网)、**IP** 对映。连接方案取决于具体网络环境。

## 7.14 关于指纹模板

**Biokey** SDK 采集到的指纹模板最大不超过 2048 字节。故指纹模板相对容量较小，很容易存储于数据库，如 **Access,MySQL,MSSQL,Orical** 等。

在 **zem100** 系列产品中，指纹模板只有 400（限二进制）字节左右。**Zem200** 指纹模板为 608 字节（限二进制）。

## 7.15 大容量指纹上传

大容量指纹一般指超过 1500 枚指纹，在一些设备中，最大能够容纳 8000 枚指纹，在上传时，必须使用缓冲模式上传指纹模板。既批处理模式上传，在该模式下，上传速度极大提高。如何使用批处理模式上传指纹模板，请参考批处理函数说明。

## 8. 扩展函数

开发包函数接口除去以上描述的函数接口外，还有一些在特殊情况下所加的接口，可能在有的开发包中不包含下列函数接口。现描述如下，请参考。

### 8.1 SSR\_GetGeneralLogData

#### [函数]

```
SSR_GetGeneralLogData([in] LONG dwMachineNumber, [out] BSTR* dwEnrollNumber, [out] LONG* dwVerifyMode, [out] LONG* dwInOutMode, [out] LONG* dwYear, [out] LONG* dwMonth, [out] LONG* dwDay, [out] LONG* dwHour, [out] LONG* dwMinute, [out] LONG* dwSecond, [out] LONG* dwWorkcode, [out,retval] VARIANT_BOOL* pVal);
```

#### [功能]

获得考勤记录，和 GetGeneralLogData 功能类似。使用该函数方式和 GetGeneralLogData 相同，只是用户编号和姓名可支持 24 位以内编码，当然，需要和特殊机器(**SSR** 等)配合使用。该类函数于 2006-09-15 完善。于 2006-11-15 日发布使用。于 2006-12-1 配合 SSR 类产品上市使用。

#### [参数说明]

dwEnrollNumber: 用户编号。其他参数在前面很多描述中具有描述，请参考。

#### [返回值]

成功则返回 True, 否则返回 False。

#### [例子]

```
Dim dwEnrollNumber As String
Dim dwVerifyMode As Long
Dim dwInOutMode As Long
Dim timeStr As String
Dim i As Long
Dim dwMachineNum, dwEMachineNum, dwYear, dwMonth, dwDay, dwHour, dwMinute, dwSecond, dwWorkcode, dwReserved, dwWorkcode As Long
```

```
If CZKEM1.ReadGeneralLogData(MachineNumber) Then
    While CZKEM1.SSR_GetGeneralLogData(MachineNumber, dwEnrollNumber, dwVerifyMode, dwInOutMode, dwYear, dwMonth, dwDay, dwHour, dwMinute, dwSecond, dwWorkcode)
        DoEvents
        i = i + 1
        Debug.Print dwEnrollNumber
    Wend
End If
```

#### 说明:

所有以 SSR\_开头的函数用户编号都为字符串类型，以下将不在描述。

### 8.2 SSR\_GetAllUserInfo

#### [函数]

Stand alone development for ZKEMSDK, Reversion 6.1  
Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.75

```
SSR_GetAllUserInfo([in] LONG dwMachineNumber, [out] BSTR * dwEnrollNumber, [out]
BSTR * Name, [out] BSTR * Password, [out] LONG * Privilege, [out] VARIANT_BOOL * Enabled,
[out,retval] VARIANT_BOOL* pVal);
```

#### [功能]

获取用户信息，该函数每执行一次，指向读到内存中的用户信息的指针移动到下一记录，当读完所有用户信息时，返回 False。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

```
Dim dwEnrollNmber As String
Dim dwEnrollNumber1
Dim Name As String
Dim password As String
Dim privilege As Integer
Dim enabled As Boolean
Dim tmpData As String
Dim tmpLength As Integer
Dim sqlstr As String
Dim TmpData1
Dim k As Long
Dim tmplateBinary(1024) As Byte
Dim tempstr As String
Dim i As Integer
Dim test As Boolean
machineNum = 1

If CZKEM1.ReadAllUserID(MachineNumber) Then
If CZKEM1.ReadAllTemplate(1) Then
While CZKEM1.SSR_GetAllUserInfo MachineNumber), dwEnrollNmber, Name, password, privilege,
enabled)
dwEnrollNumber1 = dwEnrollNmber
For k = 0 To 9
test = CZKEM1.SSR_GetUserTmpStr(MachineNumber), dwEnrollNumber1, CLng(k), tempstr,
tmpLength) ‘也可以以二进制方式取得指纹模板。
If test Then
Debug.Print tempstr
End If
Next
Wend
End If
End If
```

## 8.3 SSR\_GetUserInfo

### [函数]

```
SSR_GetUserInfo([in] LONG dwMachineNumber, [in] BSTR dwEnrollNumber, [out] BSTR*
Name, [out] BSTR* Password, [out] LONG* Privilege, [out] VARIANT_BOOL* Enabled,
[out,retval] VARIANT_BOOL* pVal)
```

### [功能]

根据用户编号获取用户消息，功能和 GetUserInfo 完全一样，只是用户编号可用字符串类型。只支持 SSR 系列产品。

### [返回值]

成功则返回 True,否则返回 False。

### [例子]

```
Dim Name As String
Dim pas As String
Dim pri As Integer
Dim en As Boolean
```

```
CZKEM1.SSR_GetUserInfo 1, "12345678901234", Name, pas, pri, en
```

## 8.4 SSR\_SetUserInfo

### [函数]

```
SSR_SetUserInfo([in] LONG dwMachineNumber, [in] BSTR dwEnrollNumber, [in] BSTR Name,
[in] BSTR Password, [in] LONG Privilege, [in] VARIANT_BOOL Enabled, [out,retval]
VARIANT_BOOL* pVal)
```

### [功能]

上传用户信息，功能和 SetUserInfo 完全一样，只是用户编号可用字符串类型。只支持 SSR 系列产品。

### [返回值]

成功则返回 True,否则返回 False。

### [例子]

```
Dim Name
Dim pas
Dim pri
Dim en
pas = ""
pri = 0
en = True
Name = "KKKKAAAADDDDDDDDEEEEEIIII"
```

```
CZKEM1.SSR_SetUserInfo 1, "12345678901234", Name, pas, pri, en
```

## 8.5 SSR\_GetUserTmpStr

### [函数]

```
SSR_GetUserTmpStr([in] LONG dwMachineNumber, [in] BSTR dwEnrollNumber, [in] LONG dwFingerIndex, [out] BSTR* TmpData, [out] LONG* TmpLength, [out,retval] VARIANT_BOOL* pVal);
```

### [功能]

以字符串方式获得用户指纹模板，和 GetUserTmpStr 功能相同。请参考 GetUserTmpStr 函数。

### [返回值]

成功则返回 True, 否则返回 False。

### [例子]

请参考 GetAllUserInfo 例子中该函数的使用。

## 8.6 SSR\_DeleteEnrollData

### [函数]

```
SSR_DeleteEnrollData([in] LONG dwMachineNumber, [in] BSTR dwEnrollNumber, [in] LONG dwBackupNumber, [out,retval] VARIANT_BOOL* pVal)
```

### [功能]

删除用户指纹、密码、用户本身等。和 DeleteEnrollData 函数功能相同。于 DeleteEnrollData 相比，参数唯一的区别是：该函数用户编号采用字符串，其他参数均一样。以下不做描述。执行成功返回 True, 否则返回 False。

### [例子]

删除用户编号为 12345678901234 的第一枚指纹模板。pVal 为函数返回参数，在开发中不可见。  
CZKEM1.SSR\_DelUserTmp 1, "12345678901234", 0

## 8.7 SSR\_GetUserTmp

### [函数]

```
SSR_GetUserTmp([in] LONG dwMachineNumber, [in] BSTR dwEnrollNumber, [in] LONG dwFingerIndex, [out] BYTE* TmpData, [out] LONG* TmpLength, [out,retval] VARIANT_BOOL* pVal)
```

### [功能]

以二进制方式获得用户指纹模板。参数 dwEnrollNumber 为用户编号，采用字符串方式，目前只支持 SSR 系列产品。功能和 GetUserTmp 函数相同，可参考 GetUserTmp 函数。例子请参考前面 SSR\_GetUserTmpStr 函数相关代码。

## 8.8 SSR\_DelUserTmp

### [函数]

```
SSR_DelUserTmp([in] LONG dwMachineNumber, [in] BSTR dwEnrollNumber, [in] LONG dwFingerIndex, [out,retval] VARIANT_BOOL* pVal)
```

#### [功能]

删除用户某一指纹模板，和 DeluserTmp 函数功能相同。dwEnrollNumber 为字符串用户编号。dwFingerIndex 为指纹索引。

#### [例子]

```
CZKEM1.SSR_DelUserTmp 1, "12345678901234", 0
```

## 8.9 SSR\_SetUserTmpStr

#### [函数]

```
SSR_SetUserTmpStr([in] LONG dwMachineNumber, [in] BSTR dwEnrollNumber, [in] LONG dwFingerIndex, [in] BSTR TmpData, [out,retval] VARIANT_BOOL* pVal)
```

#### [功能]

以字符串方式上传指纹模板。

#### [参数说明]

dwEnrollNumber: 用户编号。其他参数在前面很多描述中具有描述，请参考。

TmpData: 字符串类型指纹模板。

#### [返回值]

成功则返回 True, 否则返回 False。

#### [例子]

```
Dim dwEnrollNumber
Dim tempstr
Dim Name
Dim pas
Dim pri
Dim en

tempstr=
"ocojg5gvYQENNjFnwQ02r1lBDaCwTcENoClCgSoqK0SBFCk1VIFPlDdUgVyNyF6BBdqaOIEUZx0
dwQ1VmUgBDRcuc0EJKsZ4QQrPvUiBEPFBR4ELgLY0gRU8MTqBDzKuMIEKQSGyQQ1BJCoBCK4cKoE
PWSBLwRaIIjvBilOnPcEzQsMwwQ7SyztBCwLQdUEH2aUTwQfKTl1BBWyde8ENVtJdAQbjyU6BDnf
AHAEHxQsqQQx0JqRJEJnEjRUL0K8OERXAwGkmZzZmrrAwnKly6rOm6ubwMJupc3Lzquqmh7AwWm
mqs3t+8qZmsDBZaG6vAIKEqLMmJkfWMFfocqtdgkVouqYl8DBWKG5vXAKGaH5iCLAwVShmJxmCxy
h6qglwMFpZeITiWiTKqHwMFMoZZVOy6iC6poKMDBSqGGQjOje82lVsDBSElIRDswKSQrNqGjRSr
AwUhJSEU7LiQeJkKiYCZXwUhKSkc9LSAYCFFIP6ElVsJMTUtGLRYOcllRoQNFOcJQUVJVbAkGb2R
boRN3wMJUVVhfbQEDb2ahNGdZwMNaX2dwdnZtohzmeMDEZKTetzN4Z2HAYWYhR4dl4A=="
pas = ""
pri = 0
en = True
machineNum = 1

Name = "TEST"
CZKEM1.SSR_SetUserInfo 1, "12345678901234", Name, pas, pri, en
CZKEM1.SSR_SetUserTmpStr CLng(machineNum), "12345678901234", 0, tempstr
```



## 8.10 SSR\_SetUserTmp

### [函数]

SSR\_SetUserTmp([in] LONG dwMachineNumber, [in] BSTR dwEnrollNumber, [in] LONG dwFingerIndex, [in] BYTE\* TmpData, [out,retval] VARIANT\_BOOL\* pVal)

### [功能]

以二进制方式上传指纹模板。和 SetUserTmp 功能完全相同，只是该函数用户编号为字符串类型。当然，以 SSR 开头的函数需要相关机器支持（只能用于 SSR 类似产品）。例子请参考 SetUesrTmp 相关代码或参考 SSR\_SetUserTmpStr 例程代码。

### [参数说明]

dwEnrollNumber: 用户编号。  
dwFingerIndex: 指纹索引。  
TmpData: 用户指纹模板。

## 8.11 SetWorkCode

### [函数]

SetWorkCode([in] LONG WorkCodeID, [in] LONG AWorkCode, [out,retval] VARIANT\_BOOL\* pVal)

### [功能]

上传 WorkCode。该函数目前只支持一些对 WorkCode 功能特殊定制的机器。带有普通 WorkCode 功能机器不支持该函数接口。

### [参数说明]

WorkCodeID: WorkCode 编号。支持无符号 2 字节，即最大为 65535。  
AworkCode: WorkCode 值。可支持无符号 4 字节。  
pVal: 函数返回值，在开发时不可见。

### [例子]

```
Dim aflag As Boolean
Dim AWorkCode
Dim AworkCodeValue
AWorkCode=1
AworkCodeValue=1
aflag = CZKEM1.SetWorkCode(AWorkCode, AworkCodeValue)
CZKEM1.RefreshData 1
```

## 8.12 GetWorkCode

### [函数]

GetWorkCode([in] LONG WorkCodeID, [out] LONG\* AWorkCode, [out,retval] VARIANT\_BOOL\* pVal)

### [功能]

根据 WorkCode 编号获取 WorkCode 值。该函数目前只支持一些对 WorkCode 功能特殊定制的机器。带有普通 WorkCode 功能机器不支持该函数接口。

#### [参数说明]

WorkCodeID: WorkCode 编号。

AworkCode: WorkCode 值。

#### [例子]

```
Dim AWorkCode
Dim aflag As Boolean
Dim AworkCodeID
AworkCodeID=1

aflag = CZKEM1.GetWorkCode(AworkCodeID, AWorkCode)
CZKEM1.RefreshData 1
```

### 8.13 DeleteWorkCode

#### [函数]

DeleteWorkCode([in] LONG WorkCodeID, [out,retval] VARIANT\_BOOL\* pVal)

#### [功能]

根据 WorkCode 编号删除某一 WorkCode。该函数目前只支持一些对 WorkCode 功能特殊定制的机器。带有普通 WorkCode 功能机器不支持该函数接口。

#### [参数说明]

WorkCodeID:WorkCode 编号。

pVal: 函数返回值。

#### [例子]

```
Dim aflag As Boolean
Dim AworkCodeID
AworkCodeID=1
aflag = CZKEM1.DeleteWorkCode(AWorkCodeID)
CZKEM1.RefreshData 1
```

### 8.14 ClearWorkCode

#### [函数]

ClearWorkCode([out,retval] VARIANT\_BOOL\* pVal)

#### [功能]

清空上传的 WorkCode，当然，用户考勤时输入的 WorkCode 保留，可继续下载。

#### [例子]

```
Dim aflag As Boolean
aflag = CZKEM1.ClearWorkCode()
CZKEM1.RefreshData 1
```

### 8.15 IsTFTMachine

#### [函数]

IsTFTMachine(LONG dwMachineNumber)

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.81

**[功能]**

判断是否彩屏指纹机

**[参数说明]**

dwMachineNumber

指纹机编号

**[返回值]**

True 彩屏指纹机，反之则否。

**[例子]**

## 8.15 SSR\_EnableUser

**[函数]**

SSR\_EnableUser(LONG dwMachineNumber, BSTR dwEnrollNumber, VARIANT\_BOOL bFlag)

**[功能]**

设置用户为禁止或可用用户。该函数只支持彩屏指纹机系列。

**[参数说明]**

dwMachineNumber

指纹机编号

dwEnrollNumber

用户编号

bFlag

True:可用用户， False:禁止用户。

**[返回值]**

True:设置成功，反之则否。

**[例子]**

## 8.16 SSR\_SetUserSMS

**[函数]**

SSR\_SetUserSMS(LONG dwMachineNumber, BSTR dwEnrollNumber, LONG SMSID)

**[功能]**

设置用户的短消息。该函数只支持彩屏指纹机系列。要想设置个人的短消息，必须先通过 SetSMS 函数创建一个短消息，然后再使用此函数设置个短消息。

**[参数说明]**

dwMachineNumber

指纹机编号

dwEnrollNumber

用户编号

bFlag

True:可用用户， False:禁止用户。

**[返回值]**

True:设置成功，反之则否。

**[例子]**

Stand alone development for ZKEMSDK,Reversion 6.1

Copyright© 2005-2008 ZKSoftware Co., Ltd. All rights reserved.82

```
MachineNumber = 1;
SmsID = 1;
Tag = 253;
SmsContent = "This is a personal message";
StartTime = "2007-09-01 14:25:00"
EnrollNumber = 5;
ValidMinutes = 60; //min
czkem.SetSMS(MachineNumber, SmsID, Tag, ValidMinutes, StartTime, SmsContent);
czkem.SSR_SetUserSMS(MachineNumber, EnrollNumber, SmsID);
```

## 8.17 SSR\_DeleteUserSMS

### [函数]

```
SSR_DeleteUserSMS(LONG dwMachineNumber, BSTR dwEnrollNumber, LONG SMSID)
```

### [功能]

删除用户对应的短消息。该函数只支持彩屏指纹机系列

### [参数说明]

dwMachineNumber  
指纹机机号  
dwEnrollNumber  
用户编号。  
SMSID  
短消息编号。

### [返回值]

成功则返回 True, 否则返回 False。

### [例子]

## 8.18 SSR\_SetHoliday

### [函数]

```
SSR_SetHoliday(LONG dwMachineNumber, LONG HolidayID, LONG BeginMonth, LONG BeginDay, LONG EndMonth, LONG EndDay,
LONG TimeZoneID)
```

### [功能]

设置指纹机节假日，该函数只支持彩屏指纹机系列。最多只能设置 24 个节假日

### [参数说明]

dwMachineNumber  
指纹机机号  
HolidayID  
节假日编号，该值的范围是 1-24。  
BeginMonth  
节假日开始月份  
BeginDay  
节假日开始日  
EndMonth  
节假日结束月份  
EndDay  
节假日结束日  
TimeZoneID  
时间段编号，该值的范围是 1-50。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

## 8.19 SSR\_GetHoliday

#### [函数]

SSR\_GetHoliday(LONG dwMachineNumber, LONG HolidayID, LONG\* BeginMonth, LONG\* BeginDay, LONG\* EndMonth, LONG\* EndDay, LONG\* TimeZoneID)

#### [功能]

返回指纹机已设置节假日时间，该函数只支持彩屏指纹机系列。

#### [参数说明]

dwMachineNumber

指纹机机号

HolidayID

节假日编号，该值的范围是 1-24。

BeginMonth

节假日开始月份

BeginDay

节假日开始日

EndMonth

节假日结束月份

EndDay

节假日结束日

TimeZoneID

时间段编号。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

## 8.20 SSR\_SetGroupTZ

#### [函数]

SSR\_SetGroupTZ(LONG dwMachineNumber, LONG GroupNo, LONG Tz1, LONG Tz2, LONG Tz3, LONG VaildHoliday, LONG VerifyStyle)

#### [功能]

设置指纹机的组时间段，该函数只支持彩屏指纹机系列。

#### [参数说明]

dwMachineNumber

指纹机机号

GroupNo

组编号，该值的范围是 1-99。

Tz1

时间段 1，该值的范围是 1-50。

Tz2

时间段 2，该值的范围是 1-50。

Tz3

时间段 3，该值的范围是 1-50。

VaildHoliday

节假日是否有效。1: 有效, 0 无效。

VerifyStyle

指纹机验证方式。其值是：0(FP/PW/RF)，1(FP)，2(PIN)，3(PW)，4(RF)，5(FP&RF)，6(FP/PW)，7(FP/RF)，8(PW/RF)，9(PIN&FP)，10(FP&PW)，11(PW&RF)，12(FP&PW&RF)，13(PIN&FP&PW)，14(FP&RF/PIN)

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

## 8.21 SSR\_GetGroupTZ

#### [函数]

SSR\_GetHoliday(LONG dwMachineNumber, LONG GroupNo, LONG\* Tz1, LONG\* Tz2, LONG\* Tz3, LONG\* VaildHoliday, LONG\* VerifyStyle)

#### [功能]

返回指纹机的组时间段，该函数只支持彩屏指纹机系列。

#### [参数说明]

dwMachineNumber

指纹机机号

GroupNo

组编号，该值的范围是 1-99。

Tz1

时间段 1，该值的范围是 1-50。

Tz2

时间段 2，该值的范围是 1-50。

Tz3

时间段 3，该值的范围是 1-50。

VaildHoliday

节假日是否有效。1：有效，0 无效。

VerifyStyle

指纹机验证方式。其值是：0(FP/PW/RF)，1(FP)，2(PIN)，3(PW)，4(RF)，5(FP&RF)，6(FP/PW)，7(FP/RF)，8(PW/RF)，9(PIN&FP)，10(FP&PW)，11(PW&RF)，12(FP&PW&RF)，13(PIN&FP&PW)，14(FP&RF/PIN)。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

## 8.22 SSR\_SetUnLockGroup

#### [函数]

SSR\_SetUnLockGroup(LONG dwMachineNumber, LONG CombNo, LONG Group1, LONG Group2, LONG Group3, LONG Group4, LONG Group5)

#### [功能]

设置指纹机开锁组合，该函数只支持彩屏指纹机系列。

#### [参数说明]

dwMachineNumber

指纹机机号

CombNo

开锁组合编号，该值的范围是 1-10

Group1

组编号 1，该值的范围是 1-99。

Group2

组编号 2，该值的范围是 1-99。

Group3

组编号 3，该值的范围是 1-99。

Group4

组编号 4，该值的范围是 1-99。

Group5

组编号 5，该值的范围是 1-99。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

//设置开锁必须是所属组成 1 的用户和所属组 2 的用户先后按指纹验证通过以后，才可以开门。

```
dwMachineNumber = 1;
```

```
CombNo = 1;
```

```
Group1 = 1;
```

```
Group2 = 2;
```

```
Group3 = 0;
```

```
Group4 = 0;
```

```
Group5 = 0;
```

```
Czkem.SSR_SetUnLockGroup(dwMachineNumber, CombNo, Group1, Group2, Group3, Group4, Group5)
```

## 8.23 SSR\_GetUnLockGroup

#### [函数]

```
SSR_GetUnLockGroup(LONG dwMachineNumber, LONG CombNo, LONG* Group1, LONG* Group2, LONG* Group3, LONG* Group4, LONG* Group5)
```

#### [功能]

返回指纹机的开锁组合，该函数只支持彩屏指纹机系列。

#### [参数说明]

dwMachineNumber

指纹机机号

CombNo

开锁组合编号，该值的范围是 1-10

Group1

组编号 1，该值的范围是 1-99。

Group2

组编号 2，该值的范围是 1-99。

Group3

组编号 3，该值的范围是 1-99。

Group4

组编号 4，该值的范围是 1-99。

Group5

组编号 5，该值的范围是 1-99。

#### [返回值]

成功则返回 True,否则返回 False。

#### [例子]

## 8.24 SetDaylight

#### [函数]

```
SetDaylight(LONG dwMachineNumber, LONG Support, BSTR BeginTime, BSTR EndTime)
```

**[功能]**

设置夏令时开始时间与结束时间

**[参数说明]**

dwMachineNumber

指纹机机号

Support

设置夏令时是否有效，1 有效，0 无效。

BeginTime

夏令时开始时间。时间格式为：mm-dd hh:ss。例如：06-01 04:00

EndTime

夏令时结束时间。时间格式为：mm-dd hh:ss。例如：09-01 04:00

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**

## 8.25GetDaylight

**[函数]**

GetDaylight(LONG dwMachineNumber, LONG\* Support, BSTR\* BeginTime, BSTR\* EndTim)

**[功能]**

返回夏令时开始时间与结束时间

**[参数说明]**

dwMachineNumber

指纹机机号

Support

夏令时是否有效，1 有效，0 无效。

BeginTime

夏令时开始时间。时间格式为：mm-dd hh:ss。

EndTime

夏令时结束时间。时间格式为：mm-dd hh:ss。

**[返回值]**

成功则返回 True,否则返回 False。

**[例子]**



感谢您对本产品的关注，为提供给您完善的服务，请登录本公司技术论坛，并填写完整的注册信息，以便我们及时与您联系。

本公司周一至周五，上午九点到下午六点上班，周六值班，法定假日及、周日不上班。

我们随时恭候您的来电，为您迅速解答问题。

当您准备来电询问前，请先确定已经依照手册的步骤操作，且确认已经关闭所使用的其它应用程序。

来函请寄：北京中关村知春路 106 号太平洋国际大厦 1008 室

邮政编码：100086

来电请拨：010-51518010, 51518011, 51518012, 51518013, 51518014

传真服务：010-51518015

E-mail: [support@zksoftware.com](mailto:support@zksoftware.com), [pyh@zksoftware.com](mailto:pyh@zksoftware.com)

若您有关于本套产品任何技术上的问题，请详尽备妥下述资料，以便我们在最短的时间内为您解决问题，提供服务：

- 1、软件名称
- 2、您的电脑配备（包括厂牌、机型、CPU、内存、光驱、主板厂牌等）
- 3、Windows 95/98/NT4.0/2000/XP 或其它环境
- 4、任何您所使用的应用程序
- 5、详细描述问题的情况

您也可以登录我们公司主页 [www.zksoftware.com](http://www.zksoftware.com)，访问技术论坛并提出您的问题和建议，我们将尽快给您满意的答