

居民身份证验证读卡器接口 **API**使用手册

哈尔滨新中新电子股份有限公司
2004年12月

1. 端口类API

`int Syn_OpenPort(int iPortID);`

说明：打开串口/USB口

参数：iPortID [in] 整数，表示端口号。1-16（十进制）为串口，1001-1016（十进制）为USB口，缺省的一个USB设备端口号是1001。

返回值：成功返回0，错误返回见6。

`int Syn_ClosePort(int iPortID);`

说明：关闭串口/USB口

参数：iPortID [in] 整数，表示端口号。

返回值：成功返回0，错误返回见6。

`int Syn_GetCOMBaud(int iComID, unsigned int *puiBaudRate);`

说明：查看串口的波特率

参数：iPort [in] 整数，表示端口号。此处端口号必须为1-16，表示串口，参见7.1。
puiBaudRate [out] 无符号整数指针，指向普通串口当前波特率，默认情况下为115200。

返回值：成功返回0，错误返回见6。

`int Syn_SetCOMBaud(int iComID, unsigned int uiCurrBaud, unsigned int uiSetBaud);`

说明：设置串口的波特率

参数：iPort [in] 整数，表示端口号。此处端口号必须为1-16，表示串口。
uiCurrBaud [in] 无符号整数，调用该API前已设置的业务终端与SAM_V通信的波特率（SAM_V出厂时默认，业务终端与SAM_V通信的波特率为115200）。业务终端以该波特率与SAM_V通信，发出设置SAM_V新波特率的命令。uiCurrBaud只能为下列数值之一：115200，57600，38400，19200，9600。如果uiCurrBaud数值不是这些值之一，函数返回0x21；如果已设置的波特率与uiCurrBaud不一致，则函数返回非零，表示不能设置，调用API不成功。
uiSetBaud [in] 符号整数，将要设置的SAM_V与业务终端通信波特率。uiSetBaud只能取下列值之一：115200，57600，38400，19200，9600，如果输入uiSetBaud参数不是这些数值之一，函数返回非零，设置不成功，保持原来的波特率不变。

返回值：成功返回0，错误返回见6。

2. SAM类API

`int Syn_GetSAMStatus(int iPortID, int iIfOpen);`

说明：对SAM_V进行状态检测

参数：iPort [in] 整数，表示端口号。参见7.1。
iIfOpen [in] 整数，参见7.2。

返回值：成功返回0，错误返回见6。

`int Syn_ResetSAM(int iPortID, int iIfOpen);`

说明：对SAM_V复位

参数：iPort [in] 整数，表示端口号。参见7.1。
iIfOpen [in] 整数，参见7.2。

返回值：成功返回0，错误返回见6。

`int Syn_GetSAMID(int iPortID, unsigned char *pucSAMID, int iIfOpen);`

说明：读取SAM_V的编号（十六进制）

参数：iPort [in] 整数，表示端口号。参见7.1。
pucSAMID [out] 无符号字符串指针，SAM_V编号，16字节。
iIfOpen [in] 整数，参见7.2。

返回值：成功返回0，错误返回见6。

`int Syn_GetSAMIDToStr(int iPortID, char *pcSAMID, int iIfOpen);`

说明：读取SAM_V的编号（字符串格式）

参数：iPort [in] 整数，表示端口号。参见7.1。
pcSAMID [out] 字符串指针，SAM_V编号。
iIfOpen [in] 整数，参见7.2。

返回值：成功返回0，错误返回见6。

3. 身份证卡类API

`int Syn_StartFindIDCard(int iPortID, unsigned char *pucManaInfo, int iIfOpen);`

说明：开始找卡

参数：iPort [in] 整数，表示端口号。参见7.1。
pucManaInfo [out] 无符号字符串指针，证/卡芯片管理号，4个字节。
iIfOpen [in] 整数，参见7.2。

返回值：成功返回0，错误返回见6。

```
int Syn_SelectIDCard(int iPortID, unsigned char *pucManaMsg, int iIfOpen);
```

说明：选卡

参数：iPort [in] 整数，表示端口号。参见7.1。

pucManaMsg [out] 无符号字符指针，证/卡芯片序列号，8个字节。

iIfOpen [in] 整数，参见7.2。

返回值：成功返回0，错误返回见6。

```
int Syn_ReadMsg(int iPortID, int iIfOpen, IDCardData *pIDCardData);
```

说明：读取证/卡信息

参数：iPort [in] 整数，表示端口号。参见7.1。

iIfOpen [in] 整数，参见7.2。

pIDCardData [out] IDCardData类型读出的数据，参见5。

返回值：成功返回0，错误返回见6。

```
int Syn_ReadCard(IDCardData *pIDCardData, int Rmode);
```

说明：读卡

参数：pIDCardData [out] IDCardData类型读出的数据，参见5。

Rmode 整数，1：为连续读卡；2：为一次一次读卡。

返回值：成功返回0，错误返回见6。

4. 附加类API

```
int Syn_SendSound(int iCmdNo);
```

说明：发送语音

参数：iCmdNo [in] 声音编号

返回值：成功返回0，错误返回见6。

```
void Syn_DelPhotoFile();
```

说明：删除临时照片文件

参数：无

返回值：成功返回0，错误返回见6。

5. 结构体声明

```
typedef struct tagIDCardData{  
    char Name[32]; //姓名  
    char Sex[4];   //性别  
    char Nation[6]; //民族  
    char Born[18]; //出生日期  
    char Address[72]; //住址  
    char IDCardNo[38]; //身份证号  
    char GrantDept[32]; //发证机关  
    char UserLifeBegin[18]; //有效开始日期  
    char UserLifeEnd[18]; //有效截止日期  
    char reserved[38]; //保留  
    char PhotoFileName[255]; //照片路径  
} IDCardData;
```

6. 函数返回值

- 0 操作成功或相片解码解码正确
- 1 端口打开失败/端口尚未打开/端口号不合法
- 2 证/卡中此项无内容
- 3 PC接收超时，在规定的时间内未接收到规定长度的数据
- 4 数据传输错误
- 5 该SAM_V串口不可用，只在SDT_GetCOMBaud时才有可能返回
- 6 接收业务终端数据的校验和错
- 7 接收业务终端数据的长度错
- 8 接收业务终端的命令错误，包括命令中的各种数值或逻辑搭配错误
- 9 越权操作
- 10 无法识别的错误
- 11 寻找证/卡失败
- 12 选取证/卡失败
- 13 调用sdtapi.dll错误
- 14 相片解码错误
- 15 授权文件不存在
- 16 设备连接错误

7. 注意

7.1 系统支持的串行端口号：1-16,USB端口号：1001-1016

7.2 iIfOpen [in] 整数，0表示不在该函数内部打开和关闭串口，此时确保之前调用了Syn_OpenPort来打开端口，并且在不需要与端口通信时，调用Syn_ClosePort关闭端口；非0表示在API函数内部包含了打开端口和关闭端口函数，之前不需要调用Syn_OpenPort，也不用再调用Syn_ClosePort。

7.3 普通开发中只使用“端口类API”与“身份证卡类API”与“附加类API”就能满足开发需要，使用方法见第10部分实例。

7.4 在使用Syn_SendSound函数时请先用Syn_ClosePort函数关闭打开的端口。

7.5 SDK包包含文件

Syn_IDCardRead.dll、Syn_IDCardRead.lib、Syn_IDCardRead.h、sdtapi.dll、WltRS.dll
Syn_ReadCard.DLL license.dat

8. 声音命令暂时未定

0. 读身份证成功

1.

9. Syn_IDCardRead.h头文件：

```
#ifndef _WIN32
#define STDCALL __stdcall
#else
#define STDCALL
#endif
#ifndef SDTAPI_
#define SDTAPI_
#endif
#ifdef __cplusplus
extern "C"{
#endif
```

```
#pragma pack(1)
typedef struct tagIDCardData{
    char Name[32];
    char Sex[4];
    char Nation[6];
    char Born[18];
    char Address[72];
    char IDCardNo[38];
    char GrantDept[32];
    char UserLifeBegin[18];
    char UserLifeEnd[18];
    char reserved[38];
    char PhotoFileName[255];
}IDCardData;
```

```
#pragma pack()
```

```
/**端口类API ***/
```

```
int STDCALL Syn_GetCOMBaud(int iComID,unsigned int *puiBaud);
```

```
int STDCALL Syn_SetCOMBaud(int iComID,unsigned int uiCurrBaud,unsigned int uiSetBaud);
```

```

int STDCALL Syn_OpenPort(int iPortID);
int STDCALL Syn_ClosePort(int iPortID);

/***** SAM类API *****/
int STDCALL Syn_GetSAMStatus(int iPortID, int ilfOpen);
int STDCALL Syn_ResetSAM(int iPortID, int ilfOpen);
int STDCALL Syn_GetSAMID(int iPortID, unsigned char *pucSAMID, int ilfOpen);
int STDCALL Syn_GetSAMIDToStr(int iPortID, char *pcSAMID, int ilfOpen);

/***** 身份证卡类 API *****/
int STDCALL Syn_StartFindIDCard(int iPortID, unsigned char *pucManaInfo, int ilfOpen);
int STDCALL Syn_SelectIDCard(int iPortID, unsigned char *pucManaMsg, int ilfOpen);
int STDCALL Syn_ReadMsg(int iPortID, int ilfOpen, IDCardData *pIDCardData);

/***** 附加类API *****/
int STDCALL Syn_SendSound(int iCmdNo);
void STDCALL Syn_DelPhotoFile();
#ifdef __cplusplus
}
#endif
#endif
}

```

10. 调用实例:

10.1 vc6.0调试通过

```

void CSampleDlg::OnButton1()
{
    // TODO: Add your control notification handler code here
    int ifOpen;
    int iPort;
    int iRet;
    unsigned char pucIIN[4];
    unsigned char pucSN[8];
    IDCardData CardMsg;

    ifOpen=0; //是否需要打开串口
    iPort=1;
    iRet=Syn_OpenPort(iPort);
    if (iRet==0)
    {
        iRet=Syn_StartFindIDCard(iPort,pucIIN,ifOpen);
        if (iRet==0)
        {
            iRet=Syn_SelectIDCard(iPort,pucSN,ifOpen);
            if (iRet==0)
            {
                iRet=Syn_ReadMsg(iPort,0,&CardMsg);
                if (iRet==0)
                {
                    //显示读出的信息
                    UpdateData(true);
                    m_name=CardMsg.Name;
                    UpdateData(false);
                    Syn_SendSound(0);
                }
            }
        }
    }
}

```

```

        Syn_DelPhotoFile();
    }
}

}

}
Syn_ClosePort(iPort);
}

```

10.2 delphi7.0 调试通过

首先将 Syn_IDCardRead.h 翻译为 Delphi 格式。

```

procedure TForm1.Button4Click(Sender: TObject);
var
    ifOpen:integer;
    iPort:integer;
    iRet:integer;
    pucIIN:array[0..3] of Byte;
    pucSN:array[0..7] of Byte;
    CardMsg:TCard;
begin
    IfOpen:=0; //是否需要打开串口
    iPort:=1;

    iRet:=Syn_OpenPort(iPort);
    if iRet<>0 then
    begin
        Syn_ClosePort(iPort);
        Exit;
    end;
    iRet:=Syn_StartFindIDCard(iPort,@pucIIN, IfOpen);
    if iRet=0 then
    begin
        iRet:=Syn_SelectIDCard(iPort,@pucSN, IfOpen);
        if iRet=0 then
        begin
            iRet:=Syn_ReadMsg(iPort,0,CardMsg);
            if iRet=0 then
            begin
                label1.Caption:=trim(CardMsg.Name);
                label2.Caption:=trim(CardMsg.Sex);
                label3.Caption:=trim(CardMsg.Nation);
                label4.Caption:=trim(CardMsg.Born);
                label5.Caption:=trim(CardMsg.Address);
                label6.Caption:=trim(CardMsg.IDCardNo);
                label7.Caption:=trim(CardMsg.GrantDept);
            end;
        end;
    end;
end;

```



```

        label8.Caption:=trim(CardMsg.UserLifeBegin);
        label9.Caption:=trim(CardMsg.UserLifeEnd);
        label10.Caption:=trim(CardMsg.reserved);
        image1.Picture.Bitmap.LoadFromFile(CardMsg.PhotoFileName);
        image1.Picture.Bitmap.Transparent := True;
        image1.Picture.Bitmap.TransparentColor := image1.Picture.Bitmap.canvas.pixels[5,3];
        Syn_SendSound(0);
        Syn_DelPhotoFile;
    end;
end;
end;
Syn_ClosePort(iPort);
end;

```

10.3 delphi7.0 调试通过

```

function Syn_ReadIDCard(i:integer):integer;stdcall;external 'Syn_ReadCard.Dll'; //读卡
参数 i:= 1 为连续读卡, i:= 2 为一次一次读卡

function Syn_GetState():string;stdcall;external 'Syn_ReadCard.Dll'; //获得读卡状态
function Syn_Name():String;stdcall; external 'Syn_ReadCard.Dll'; // 姓名
function Syn_Sex():string;stdcall; external 'Syn_ReadCard.Dll'; // 性别
function Syn_Nation():String;stdcall;external 'Syn_ReadCard.Dll'; //民族
function Syn_Born():String;stdcall;external 'Syn_ReadCard.Dll';//出生年月
function Syn_Address():String;stdcall;external 'Syn_ReadCard.Dll';//家庭住址
function Syn_IDCardNo():String;stdcall;external 'Syn_ReadCard.Dll';//身份证号
function Syn_GrantDept():String;stdcall;external 'Syn_ReadCard.Dll';//发证机关
function Syn_UserLifeBegin():String;stdcall;external 'Syn_ReadCard.Dll';//发证日期
function Syn_UserLifeEnd():String;stdcall;external 'Syn_ReadCard.Dll';//结束日期
function Syn_Reserved():String;stdcall;external 'Syn_ReadCard.Dll';//保留字
function Syn_PhotoFileName():String;stdcall;external 'Syn_ReadCard.Dll';//照片路径

function Syn_PhotoStr():String;stdcall;external 'Syn_ReadCard.Dll'; //照片转化为字符串
function Syn_StrToPhoto(PhotoFile:string):String;stdcall;external 'Syn_ReadCard.Dll'; //字符串转
化为照片

procedure TForm1.Button3Click(Sender: TObject);
var
i,j:integer;
IDCardInfor:TIDCardInfo;
Bmp:TBitMap;
MyJpeg:TJPEGImage;
begin
    i:= Syn_ReadIDCard(1); //1, 为连续读卡! 2, 为一次读卡
    Edit1.Text:= Syn_GetState();
    Label1.Caption:= Syn_Name();
    Label2.Caption:= Syn_Sex();
end;

```