

Please install the software correctly, and copy the file spoem.dll to bin directory. (for example : \sp3000u\bin )

### 1. The class is exported from the spoem.dll (defined in spoem.h)

```
class SPOEM_API CSpoem {
public:
    CSpoem(void);
    // TODO: add your methods here.
    void InitParameter(char* chSPDir);
    void ExitSpDll();

    BOOL USBCommunication();
    BOOL SelectDevice(char* chMFG, char* chDEV);

    BOOL LoadDataFile(LPDATAFILEPARA lpDataFilePara);
    BOOL LoadDataDirectly(LPLOADDATAPARA LoadDataPara);

    BOOL BeginProgram(char* chAuto);
    void GetProgramStatus(LPPROGRAMSTATUS ProgramStatus);
    unsigned int GetCheckSum(BOOL bRecalc);
    BOOL CheckPin();
    void GetError(LPOPERATIONERROR OperationError);
    void OperationOption(LPOPERATIONOPTION OperOption);
    BOOL CancelProgram();

    BOOL SelectTTLDevice(char* chDEV);
    void GetTTLTestError(LPTTLTESTERROR TtlTestError);
};
```

### 2. Demo(Test.exe)

1. Introduce how to use class : Compiling the demo needs "spoem.h" and "spoem.lib".  
Running the demo needs "spoem.dll".
2. VC++ project of "Test" supposes the software of USB installed in the directory "c:\sp3000"u and "spoem.dll" in the directory "c:\sp3000u\bin".
3. Running the demo: Communication first,  
Select Device -> Load Data File -> Program

### 3. Discuss details

1. Initialize ----- InitParameter(char\* chSPDir)  
Parameter: chSPDir = the directory which the file "spoem.h" is in  
Return : no  
Note : Call one time and only one time  
Before exiting, call ExitSpDll().  
Demo: BOOL CTestDlg::OnInitDialog()
2. Communication ----- USBCommunication()  
Parameter: no  
Return : BOOL  
  
Error: See GetError(LPOPERATIONERROR OperationError)  
Step: 1  
Type: 0 No error or unknown error

1 directory is error(chSPDir) or lack of nessary files

Demo: void CTestDlg::OnButtonTest()



3.Select Device ----- SelectDevice(char\* chMFG,char\* chDEV)

Parameter : Manufactuer , Device

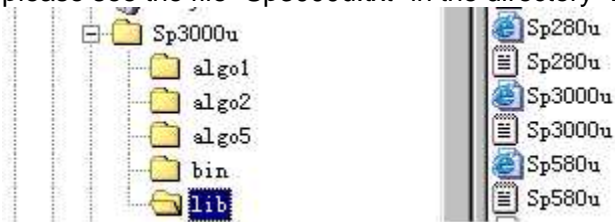
Return : BOOL

Error :

Step: 2

- Type: 0 No error or unknown error
- 1 Not Support this chip
- 2 Communication first or no programmer
- 3 chMFG error
- 4 chDEV error

Note : chMFG,chDEV must be correct, including space character . If not sure, please see the file "Sp3000u.txt" in the directory "Lib"



and copy chMFG,chDEV.

ALTERA ----- chMFG

EPROM & EEPROM

EPC1 (LC)@PLCC20 [SA001A]	EPC1 (LI)@PLCC20 [SA001A]
EPC1 (PC)	EPC1 (PI)
EPC1441 (LC)@PLCC20 [SA001A]	EPC1441 (LI)@PLCC20 [SA001A]
EPC1441 (PC)	EPC1441 (PI)
EPC2 (LC)@PLCC20 [SA001A]	EPC2 (LI)@PLCC20 [SA001A]
EPC2 (TC)@TQFP32 [SA663]	EPC2 (TI)@TQFP32 [SA663]
EPCS1SI8@S0IC8 [SA602A]	EPCS4SI8@S0IC8 [SA602A]

PLD

EP600	EP600@PLCC28 [SA008A]
EP610	EP610@PLCC28 [SA008A]
EP610A	EP610A@PLCC28 [SA008A]
EP610T	EP610T@PLCC28 [SA008A]
EP630	EP630@PLCC28 [SA008A]
EPM7032@PLCC44 [SA244]	
EPM7032S@PLCC44 (7032 mode) [SA244]	

----- chDEV

Demo: void CTestDlg::OnButtonSelect()



4. Load Data File ----- LoadDataFile(LPDATAFILEPARA lpDataFilePara)

Parameter :

```
typedef struct _DATAFILEPARA{
```

```
    char chFileName[256];
    char chType;          //0  binary
                        //1  Intel hex
                        //2  Motorola
                        //3  Tek
                        //4  Extend Tek
                        //5  pof
                        //6  jed
                        //remain error
```

```
    int nBufferAddress;
```

```
    int nFileAddress;
```

```
    char chFileMode;     //0 normal
                        //1 Even[1st of 2]
                        //2 Odd[2nd of 2]
                        //3 1st byte of 4
                        //4 2nd byte of 4
                        //5 3rd byte of 4
                        //6 4th byte of 4
                        //7 1st 2_byte of 4
                        //8 2nd 2_byte of 4
                        //remain error
```

```
    BOOL bInitBuffer;    // 1, fill buffer with initvalue before load file
```

```
}DATAFILEPARA, *LPDATAFILEPARA;
```

Return : BOOL

Checksum : GetChecksum(BOOL bReCalc);

if bReCalc=TRUE, recalculate and return CheckSum  
FALSE, return CheckSum

Error :

Step: 3

Type: 0 No error or unknown error

1 Communication first or no programmer

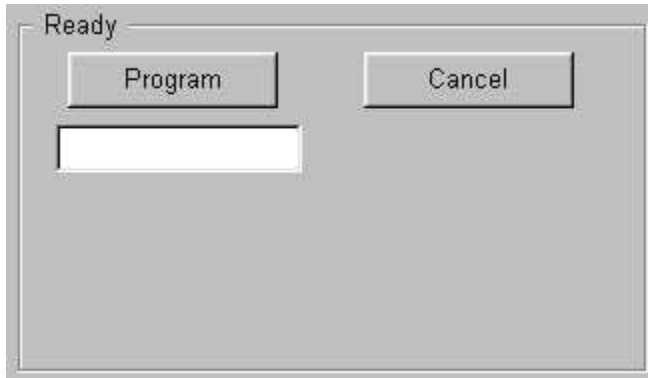
2 Select device first

3 Porgrammer is busy

4 File format error , only POF or JED file valid ( PLD ) (chType)

5 POF error or unmatched



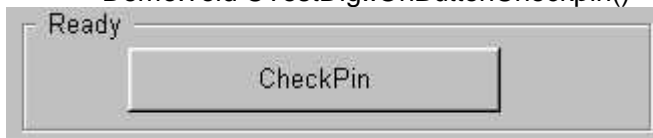


Note: Call BeginProgram(char\* chAuto) will return immediately, not waiting end of programming. Call GetProgramStatus() to check programming status. In demo, using SetTimer() to status .

5. Insertion Test ----- CheckPin();  
 Parameter : no  
 Return : BOOL

Error:  
 Step: 5  
 Type: 0 No error or unknown error  
 1 Communication first or no programmer  
 2 Select device first  
 3 Programmer is busy  
 4 No chip  
 5 Pins unmatched  
 6 Chip inserted upward  
 7 Chip inserted backward  
 8 Not support insertion test

Demo: void CTestDlg::OnButtonCheckpin()



6. Load Data into Buffer directly ----- LoadDataDirectly(LPLOADDATAPARA LoadDataPara)

Parameter :  

```
typedef struct _LOADDATAPARA{
  unsigned int nStartAddress;
  unsigned int nEndAddress;
  char chDataType; //0 Data
                  //1 EncrypTion
                  //2 Dev.Config (64 Bytes)
  unsigned char* chData;
}LOADDATAPARA,*LPLOADDATAPARA;
```

Return : BOOL ; FALSE = out of range

Note : Data Buffer(EPROM and FUSE) ( chDataType=0 )

Encryption Buffer ( chDataType=1 )

Dev.Config Buffer ( chDataType=2 )

When chDataType=2, chData is pointer to the array which size is 64 , nStartAddress and nEndAddress invalid. The data of array comes from the file generated by SP3000.exe. (See how to generate

Dev.Config file)

Error :

Step: 6

Type: 0 No error or unknown error  
1 Communication first or no programmer  
2 Select device first  
3 Porgrammer is busy

Demo: void CTestDlg::OnButtonTestLoaddata()  
void CTestDlg::OnButtonEncry()  
void CTestDlg::OnButtonDevconfig()

7.Set Operation Option ----- OperationOption(LPOPERATIONOPTION  
OperOption)

Parameter :

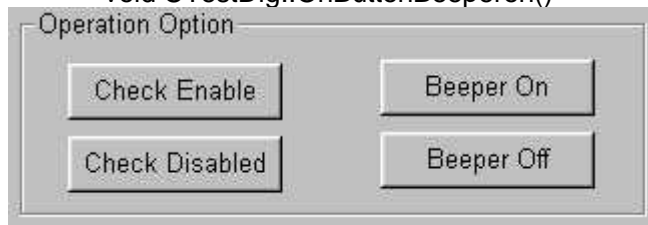
```
typedef struct _OPERATIONOPTION{  
    char chEnable; //bit0 = 1, bIDCheck Enable  
                                     //bit1 = 1, bBeeper Enable  
                                     //bit2 = 1, nStartAddress and nEndAddress Enable  
  
    BOOL bIDCheck;  
    BOOL bBeeper;  
    unsigned int nStartAddress;  
    unsigned int nEndAddress;  
}OPERATIONOPTION,*LPOPERATIONOPTION;
```

Return:无

Note: Set when valid

Demo: void CTestDlg::OnButtonIdcheckoff()

void CTestDlg::OnButtonIdcheckon()  
  
void CTestDlg::OnButtonBeeperoff()  
void CTestDlg::OnButtonBeeperon()



8.Cancel ----- CancelProgram()

Parameter: no

Return: BOOL

Error:

Step: 7

Type: 0 No error or unknown error  
1 Communication first or no programmer  
2 Select device first  
3 Porgrammer is busy  
4 Can't cancel

9.Select TTL Devcie

See "Select Device" , chMFG = "TTLTEST" , chDEV= name of TTL  
device("22100")

Step: 8

Type:       0 No error or unknown error  
          1 Communication first or no programmer  
          2 chDEV error

11. Check Error       -----    GetError(LPOPERATIONERROR OperationError)

Parameter:

typedef struct \_OPERATIONERROR{

char chStep;

char chType;

}OPERATIONERROR,\*LPOPERATIONERROR;

Return: no

Note : See Demo

Demo : void CTestDlg::ShowError()



12. How generate the Dev.Config file : when save the project file ,will also generate the Dev.Config file with extended name "dcf" which save the content of Dev.Config.