



Portable Data Collector Programming Guide for PC Application

Version: 3.0.0.3

2007/01/30

Support PT-10/12/18

Copyright c 2003-2007 by ARGOX Information Co., Ltd.

<http://www.argox.com>

Preface

When the customer uses PDC products, its final purpose is to gather the materials well to upload to PC end to do the materials to deal with. Or the materials that must be consulted are downloaded to the materials while doing and gathering the materials of PDC end than to the procedure. Like come say, customer will is it use procedure making one's own systematic procedure to use, develop course they can meet how link , communicate , exchange demand of materials in it, in order to solve above-mentioned problems here , offer Windows relevant component and help the customer to develop the system.

We offer DLL, DOS Command , Samples for Windows OS.

Table of Contents

Preface.....	0
Table of Contents	0
1 Development Environment:	2
1.1 The Catalogue Explaining:	2
1.2 Developing Instrument:	2
1.3 Component List:.....	2
2 Material of Library :	3
2.1 File System Property:	3
2.2 Channel for Connecting:	3
2.3 The Single File Transmitting:	3
2.4 The Multi Files Transmitting:	3
2.5 The Transmission Dialog:	3
2.6 The PDC Detail Information:.....	4
3 Function List:	5
Table 3-1 the Function List	5
4 Function Description:	8
4.1 Common Function:	8
4.2 Common Transmission:	11
4.3 Single Channel Function:	14
4.4 Multi-Channels Function:.....	29
5 Structure Description:	30
5.1 _DIR_INFO.....	30
6 Error Code List	31
Table 6-1 Error Code List	31
7 Edition History:.....	33
COMM_SDK 3.0.0.0	33
COMM_SDK 3.0.0.1	33
COMM_SDK 3.0.0.2	33
COMM_SDK 3.0.0.3	33
8 Example PortTest 2.3:	34
8.1 Main Function:	34
8.2 Development Notice:.....	34
8.3 Main Picture:	35
8.4 PortType picture is explained :	36
8.5 Inquire Picture:	36

8.6	File Access Picture:	37
8.7	Transmission Dialog:	37
8.8	Commands Picture:	37
8.9	Edition History:	38

1 Development Environment:

1.1 The Catalogue Explaining:

Directory	Sub-Directory	Function Description
BIN		Execution shelf site produced finally
	SDKDebug	Offer the place that the customer makes Debug
	SDKRelease	Offer the place that the customer makes Release
	TestFile	What need files temporarily while testing
	Library	When DLL component links, necessary head file.
DOC		Manual
OBJ		Provisional Object Files
PortTest		Example
	PortTest_600	Support VC6.0 in workspace
	PortTest_700	Support VC.NET 2003 in workspace
	Source	Test Program File

1.2 Developing Instrument:

- All components are support Microsoft Windows 98/ME/2000/XP.
- At present example offer Visual C++ for 6/ 7 (.NET 2003)
- Visual C++ .NET 2005 workspace is converted from .NET 2003 workspace.

1.3 Component List:

File Name	Function Description
COMM _ SDK.dll	DLL component master file
COMM _ SDK.lib	DLL component links address file
Library.h	DLL component links and head file
PortTest_600.exe	The example tests file for VC 6.0
PortTest_700.exe	The example tests file for VC .NET 2003

2 Material of Library :

2.1 File System Property:

- PT-10/12/18 provides 2 virtual disks for “Disk C”and “Disk D”.
- PDC Configuration:

	Model	PT-10	PT12	PT-18
Physical Component	RAM	1 MB	2 MB	4 MB
	ROM	1 MB	1 MB	2 MB
File System	Disk C (RAM)	706K	1.7 MB	3.7 MB
	Disk D (ROM)	576KB~704KB (Default: 704 KB)	576KB~704KB (Default: 704 KB)	1.5MB~1.7MB (Default: 1.7 MB)

- Material of Disk:

Path	Attribute	Physical Device	Description
D:\Program	R/W	ROM	Material of Project
D:\Lookup	R/W	ROM	Lookup Files
D:\Fonts	R/W	ROM	Font Image File
C:\Data	R/W	RAM	Collected Data

2.2 Channel for Connecting:

We offer two channels for connecting.

- One is RS-232 for Cradle's RS-232 and RS-232 Cable.
- Another is USB for Cradle's USB.

2.3 The Single File Transmitting:

We offer the single file transmission function, user's can use

[Req_DownloadFile](#), [Req_UploadFile](#) to complete mission.

2.4 The Multi Files Transmitting:

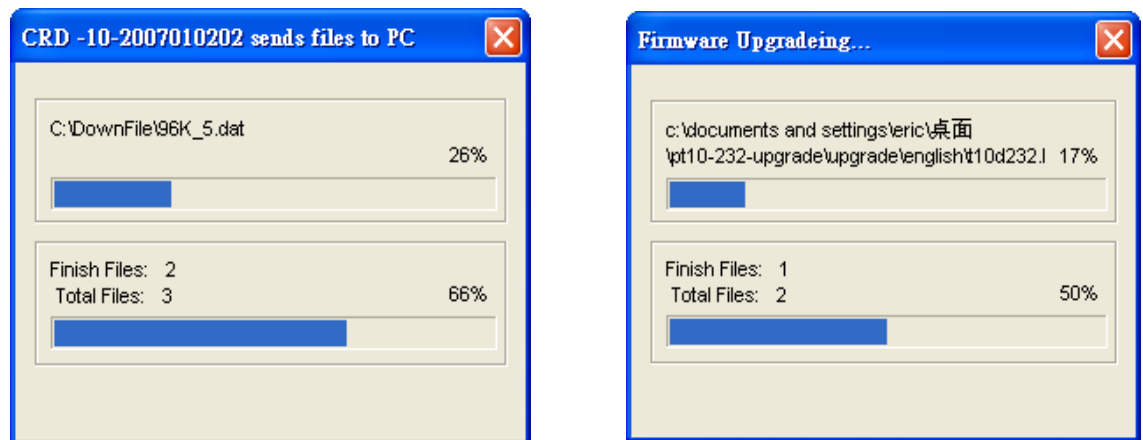
We offer the multi files transmission function, user's can use

[Req_DownloadMultiFile](#), [Req_UploadMultiFile](#) to complete mission.

2.5 The Transmission Dialog:

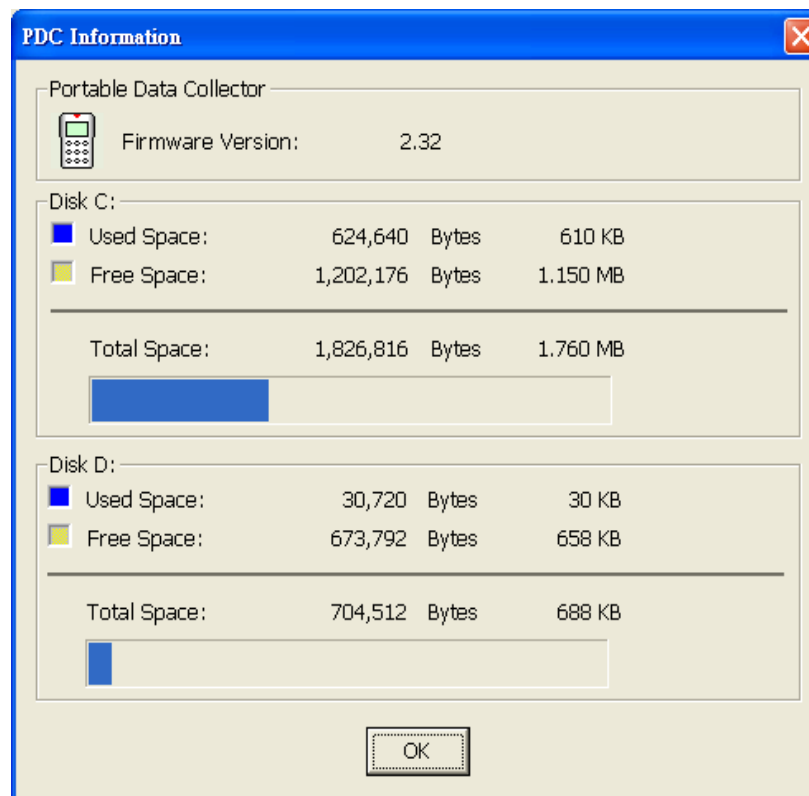
- We offer a dialog to indicate the percentage of transmission.
- It Supports the Up/Download “[Req_DownloadFile](#), [Req_UploadFile](#), [Req_DownloadMultiFile](#), [Req_UploadMultiFile](#)” and Firmware Upgrade” [Req_FirmWare](#)”.

- If you don't need the feature, you can call this ["SetTransmissionDialog"](#) to disable the dialog during transmission.



2.6 The PDC Detail Information:

- We offer two ways to get information by [Req_PDC_Info](#).
- First is from return value by import argument.
- Second is display internal dialog to indicate the PDC information.



3 Function List:

Table 3-1 the Function List

Common Function	
Below all functions can be called at anytime, anywhere.	
GetDllVersion	Obtain DLL component edition
SetMessageDialog	Enable or Disable all inside message dialogs.
SetTransmissionDialog	Enable or Disable all inside transmission dialog.
SetRWPath	Set the file path of Burning FROM file.
Common Transmission	
Below all functions must use the OpenPort and ClosePort o be start and end position.	
OpenPort	Open the channel
ClosePort	Close the channel
WriteData	Write into the materials to the channel
ReadData	Read the materials from the channel
Single Channel Function	
Below all functions must use the AL_OpenPort and AL_ClosePort to be start and end position.	
USB_GetDeviceName	Obtain USB device name after linking the channel
USB_GetDeviceVer	Obtain USB device F/W edition after linking the channel
AL_OpenPort	Open the demand channel
AL_ClosePort	Close the demand channel
Req_Dir	Ask for catalogue inquiry
Req_DirDataLen	The ones that asked for catalogues and needed while inquiring about stored the size of space.
Req_DirectoryExist	Ask to inquire whether the catalogue exists
Req_FileExist	Ask to inquire whether the file exists

Req_DownloadFile	Ask to download a file that single file of PC carries to PDC
Req_DownloadMultiFile	Ask to download many files that many files of PC carry to PDC
Req_UploadFile	Ask to upload a file that single file of PDC carries to PC
Req_UploadMultiFile	Ask to upload many files that many files of PDC carry to PC
Req_DeleteFile	Ask to delete the files in PDC
Req_FormatDisk	Ask to format disc catalogues of PDC.
Req_DefragDisk	Ask to defrag reform D magnetic disc catalogue of PDC
Req_FirmWare	Ask to upgrade the firmware editions of PDC
Req_SetDate	Ask to set PDC date and time.
Req_SerialNum	Ask to obtain a PDC product serial number.
Req_ConnectionTest	Ask to link with PDC.
Req_PDC_Info	Inquire PDC relevant information
Multi-Channels Function	
MultiPort _ Create	Have not opened
MultiPort _ Destroy	Have not opened
MultiPort _ Open	Have not opened
MultiPort _ Close	Have not opened
MultiPort _ Write	Have not opened
MultiPort _ Read	Have not opened
MultiPort _ USB _ GetDeviceName	Have not opened
MultiPort _ EnumAllUSBDeviceName	Have not opened
MultiPort _ Req _ Dir	Have not opened
MultiPort _ Req _ DirDataLen	Have not opened
MultiPort _ Req _ DirectoryExist	Have not opened
MultiPort _ Req _ FileExist	Have not opened
MultiPort _ Req _ DownloadFile	Have not opened
MultiPort _ Req _ DownloadMultiFile	Have not opened
MultiPort _ Req _ UploadFile	Have not opened

MultiPort _ Req _ DeleteFile	Have not opened
MultiPort _ Req _ FormatDisk	Have not opened
MultiPort _ Req _ DefragDisk	Have not opened
MultiPort _ Req _ FirmWare	Have not opened
MultiPort _ Req _ SetDate	Have not opened
MultiPort _ Req _ SerialNum	Have not opened
MultiPort _ Req _ ConnectionTest	Have not opened

4 Function Description:

4.1 Common Function:

GetDllVersion

Purpose

Obtain DLL component edition

Syntax

```
void GetDllVersion(char *pVersion);
```

Return Value

None

Parameters

pVersion

Make storing the space of information of the edition

Remarks

DLL version format is x.x.x.x where x is digit. (3.0.0.0)

Example

```
char acVersion[20];  
GetDllVersion(acVersion);
```

See Also

[\[Go To Table\]](#)

SetMessageDialog

Purpose

Enable or Disable all inside message dialogs.

Syntax

```
void SetMessageDialog(BOOL bEnable);
```

Return Value

None

Parameters

bEnable

“TRUE” allows to display all inside message dialogs.

“FALSE” forbid DLL to display any message dialog.

Remarks

All message dialogs are default for enabling display, if you don't call **SetMessageDialog**.

The COMM_SDK Message Dialog:



Example

```
SetMessageDialog(0);
```

See Also

[GetDllVersion](#)

[\[Go To Table\]](#)

SetTransmissionDialog

Purpose

Enable or Disable all inside transmission dialog.

Syntax

```
void SetTransmissionDialog (BOOL bEnable);
```

Return Value

None

Parameters

bEnable

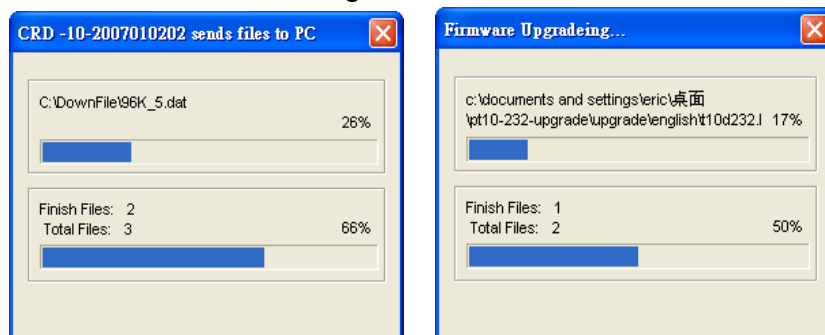
“TRUE” allows to display all inside transmission dialog.

“FALSE” forbid DLL to display the transmission dialog.

Remarks

The transmission dialog is default for enabling display, if you don't call **SetTransmissionDialog**.

The Transmission Dialog:



Example

```
SetTransmissionDialog (0);
```

See Also

[GetDllVersion](#), [SetMessageDialog](#)

[\[Go To Table\]](#)

SetRWPath

Purpose

Set the file path of Burning FROM file.

Syntax

void SetRWPath(char *path);

Return Value

None

Parameters

path

Please point out the “SST_RW.abp” file folder path in the PC.

Remarks

If the PDC can't find any internal program flash rom code, you have to set the path for installing program flash rom code from outside.

The “SST_RW.abp” file usually puts in “ArgoBuilder” application installing path as “C:\\Program File\\Argox\\ArgoBuilder\\DP”.

Example

SetRWPath("C:\\Program File\\Argox\\ArgoBuilder\\DP");

See Also

[\[Go To Table\]](#)

4.2 Common Transmission:

OpenPort

Purpose

Open the channel

Syntax

```
int OpenPort(int nPortType, int nSelPort, int nBaud, int  
nByteSize, int nParity, int nStopBits, int nFlowCtrl, int  
nShowDlg);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error
information, please refer below error code table([Table 6-1](#)).

Parameters

nPortType

0: Serial, 1:USB

nSelPort

0~9 means COM0~COM9, or USB Device Number 0~9

nBaud

only for serial port, 6 -> 4800,
0,7 -> 9600,
9 -> 19200,
10 -> 38400
11 -> 56000
13 -> 115200

nByteSize

Data bit number, 0 or 7 -> 7-bit data
8 -> 8-bit data

nParity

Parity, 0 -> none parity,
1 -> even parity,
2 -> odd parity

nStopBits

Stop bit number, 0 or 1 -> 1 stop bit,
2 -> 2 stop bits

nFlowCtrl

Flow Control, 0 -> Disable,
1 -> Enable

nShowDlg

Reserve

Remarks

Please use it with common transmission as [ClosePort](#), [ReadData](#), [WriteData](#).

Example

```
// PORT_SERIAL for COM1: 115200 bps, 8 data bits, none
parity , one stop bit, Enable Flow Control .
if(OpenPort(0, 1, 13, 8, 0, 1, 0) != REQ_SUCCESS)
    return;
// PORT_USB for device number 0, Don't need any setting.
if(!OpenPort(1, 0, 0, 0, 0, 0, 0) != REQ_SUCCESS)
    return;
```

See Also

[ClosePort](#), [ReadData](#), [WriteData](#)

[\[Go To Table\]](#)

ClosePort

Purpose

Close the channel

Syntax

void ClosePort(void);

Return Value

None

Parameters

None

Remarks

Please use it with common transmission as "[OpenPort](#)".

Example

ClosePort();

See Also

[OpenPort](#), [WriteData](#), [ReadData](#)

[\[Go To Table\]](#)

WriteData

Purpose

The **WriteData** function writes data to a channel.

Syntax

int WriteData(char *pDataBuf, int nBytesToWrite);

Return Value

The **WriteData** function returns variable that receives the number of bytes written.

Parameters

pDataBuf

[in] Pointer to the buffer containing the data to be written to the channel.

nBytesToWrite

[in] Specifies the number of bytes to write to the channel.

Remarks

Please use it with common transmission as [OpenPort](#).

Example

```
int nBytesToRead;
char acBuf[10] = "Test Write";
nBytesToWrite= sizeof(acBuf);
if(WriteData(acBuf, nBytesToWrite) != nBytesToWrite)
    FAIL;
```

See Also

[OpenPort](#), [ClosePort](#), [ReadData](#)

[\[Go To Table\]](#)

ReadData

Purpose

The **ReadData** function reads data from a channel.

Syntax

```
int ReadData(char* pDataBuf, int nBytesToRead);
```

Return Value

The **ReadData** function returns variable that receives the number of bytes read.

Parameters

pDataBuf

[out] Pointer to the buffer that receives the data read from channel.

nBytesToRead

[in] Specifies the number of bytes to be read from channel.

Remarks

Please use it with common transmission as [OpenPort](#).

Example

```
int nBytesToRead;
char acBuf[10];
```



```
nBytesToRead = 10;
if(ReadData(acBuf, nBytesToRead) != 10)
    FAIL;
```

See Also

[OpenPort](#), [ClosePort](#), [WriteData](#)

[\[Go To Table\]](#)

4.3 Single Channel Function:

USB_GetDeviceName

Purpose

Obtain USB device name after linking the channel

Syntax

```
char* USB_GetDeviceName(void);
```

Return Value

If the function succeeds, the return value is a USB device name.
If the function fails, the return NULL.

Parameters

None

Remarks

Only for linking with CRD-XX.
The device name format is "CRD -XX-XXXXXXXX".
For examples: CRD-10 is "CRD -10-60924710"
CRD-18 is "CRD -18-60924710"

Example

```
strTemp = USB_GetDeviceName();
GetDlgItem(IDC_USB_DEVICENAME)->SetWindowText(strTemp);
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#), [USB_GetDeviceVer](#)

[\[Go To Table\]](#)

USB_GetDeviceVer

Purpose

Obtain USB device F/W edition after linking the channel

Syntax

```
int USB_GetDeviceVer(void);
```

Return Value

If the function succeeds, the return decimal integer's value is a

USB device F/W edition.
If the function fails, the return zero.

Parameters

None

Remarks

Only for linking with CRD-XX.
The device F/W edition format is 2 digits for "XX".
For examples: "10" means version 1.0, "22" means version 2.2

Example

```
strTemp.Format("%d.%d", USB_GetDeviceVer()/10,
    USB_GetDeviceVer()%10);
GetDlgItem(IDC_USB_DEVICE_VER)->SetWindowText(strTemp);
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#), [USB_GetDeviceName](#)
[\[Go To Table\]](#)

AL_OpenPort

Purpose

Open the demand channel

Syntax

```
int AL_OpenPort(int nPortType, int nSelPort, int nBaud, int
nByteSize, int nParity, int nStopBits);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

nPortType

0: Serial, 1:USB

nSelPort

0~9 means COM0~COM9, or USB Device Number 0~9

nBaud

only for serial port, 6 -> 4800,
0,7 -> 9600,
9 -> 19200,
10 -> 38400
11 -> 56000
13 -> 115200

nByteSize

Data bit number, 0 or 7 -> 7-bit data
8 -> 8-bit data

nParity

Parity, 0 -> none parity,
1 -> even parity,
2 -> odd parity

nStopBits

Stop bit number, 0 or 1 -> 1 stop bit,
2 -> 2 stop bits

nFlowCtrl

Flow Control, 0 -> Disable,
1 -> Enable

Remarks

If you want to use any REQ function, you have to call this **AL_OpenPort** first.

Example

```
// PORT_SERIAL for COM1: 115200 bps, 8 data bits, none
// parity , one stop bit, Enable Flow Control .
if(AL_OpenPort(0, 1, 13, 8, 0, 1) != REQ_SUCCESS)
    return;
// PORT_USB for device number 0, Don't need any setting.
if(!AL_OpenPort(1, 0, 0, 0, 0, 0) != REQ_SUCCESS)
    return;
```

See Also

[AL_ClosePort](#), [OpenPort](#), [ClosePort](#)

[\[Go To Table\]](#)

AL_ClosePort

Purpose

Close the demand channel

Syntax

void AL_ClosePort(void);

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

None

Remarks

If you want to use any REQ function, you have to call this **AL_OpenPort** first and end in **AL_ClosePort**.

Example

```
AL_ClosePort();
```

See Also

[AL_OpenPort](#), [OpenPort](#), [ClosePort](#)

[\[Go To Table\]](#)

Req_Dir

Purpose

Ask for catalogue inquiry

Syntax

```
int Req_Dir(char *DirItem, int * nFileItem);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

DirItem

[out] Pointer to the buffer that receives the data get from PDC disk property.

The buffer data is for [DIR_INFO](#) structure. You can get file or directory information by [DIR_INFO](#).

nFileItem

[in] Specifies the number of bytes to be got from PDC disk property.

The **nFileItem** parameter is from **ILen** of [Req_DirDataLen](#).

Remarks

If you want to use this **Req_Dir** function, please call **Req_DirDataLen** first to get buffer length.

Example

Please see OnButReqDir or OnButReqDirFolder of PortTest application.

See Also

[AL_OpenPort](#), [AL_ClosePort](#), [DIR_INFO](#)

[\[Go To Table\]](#)

Req_DirDataLen

Purpose

The ones that asked for catalogues and needed while inquiring about stored the size of space.

Syntax

```
int Req_DirDataLen(char *strFile, long *ILen);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

strFile

[in] Specifies the path of request about disk directory to be got from PDC disk property. As "C:\", "C:\Data", "D:\", "D:\Lookup".

ILen

[out] Pointer to the buffer that receives the buffer length get from PDC disk property..

Remarks

This **Req_DirDataLen** must be with **Req_Dir** is shared together, by **Req_DirDataLen** inquires using the size of demand of Buffer, gives **Req_Dir** takes out the materials and puts in Buffer.
PDC only supports Disk C (RAM Type), D (ROM Type) at present.

Example

```
long lBufferSize;  
if(Req_DirDataLen("C:\\", &lBufferSize) == REQ_SUCCESS)
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#), [DIR_INFO](#)

[\[Go To Table\]](#)

Req_DirectoryExist

Purpose

Ask to inquire whether the catalogue exists.

Syntax

```
int Req_DirectoryExist(char *strDir);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

Remarks

Example

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_FileExist

Purpose

Ask to inquire whether the file exists.

Syntax

```
int Req_FileExist(char *strFile);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".

If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

Remarks

Example

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_DownloadFile

Purpose

Ask to download a file that single file of PC carries to PDC.

Syntax

```
int Req_DownloadFile(char *strFile, char *strDest, int  
nCreateFlag);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".

If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

Remarks

Example

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_DownloadMultiFile

Purpose

Ask to download many files that many files of PC carry to PDC.

Syntax

```
int Req_DownloadMultiFile(char *strFile, char *strDest, int  
nMaxFiles, int nFileIndex);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error
information, please refer below error code table([Table 6-1](#)).

Parameters

Remarks

Example

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_UploadFile

Purpose

Ask to upload a file that single file of PDC carries to PC

Syntax

```
int Req_UploadFile(char *strOrigin, char *strDestination);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error
information, please refer below error code table([Table 6-1](#)).

Parameters

Remarks

Example

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_UploadMultiFile

Purpose

Ask to upload many files that many files of PDC carrie to PC.

Syntax

```
int Req_UploadMultiFile(char *strOrigin, char
*strDestination, int nMaxFiles, int nFileIndex);
```

Return Value

If the function succeeds, the return value is “[REQ_SUCCESS](#)”.

If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

Remarks

Example

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_DeleteFile

Purpose

Ask to delete the files in PDC

Syntax

```
int Req_DeleteFile(char *strFile);
```

Return Value

If the function succeeds, the return value is “[REQ_SUCCESS](#)”.

If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

strFile

Remarks

Example


```
If(Req_DeleteFile("C:\\Data\\data.dat") == REQ_SUCCESS)
    AfxMessageBox("Delete file succeeds.");
else
    AfxMessageBox("Delete file failed.");
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_FormatDisk**Purpose**

Ask to format disc catalogues of PDC.

Syntax

```
int Req_FormatDisk(char *strDisk);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

strDisk

Remarks**Example**

```
if(Req_FormatDisk("C:\\") == REQ_SUCCESS)
    AfxMessageBox("Formatting the disk C drive succeeds.");
else
    AfxMessageBox("Formatting the disk C drive failed.");
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_DefragDisk**Purpose**

Ask to defrag reform D magnetic disc catalogue of PDC

Syntax

```
int Req_DefragDisk(char *strDisk);
```

Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

strDisk

Remarks

Example

```
if(Req_DefragDisk("D:\\") == REQ_SUCCESS)
    AfxMessageBox("Defrag the disk D drive succeeds.");
else
    AfxMessageBox("Defrag the disk D drive failed.");
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_FirmWare

Purpose

Ask to upgrade the firmware ditions of PDC

Syntax

```
int Req_FirmWare(char *strFile,char *strAbpFile);
```

Return Value

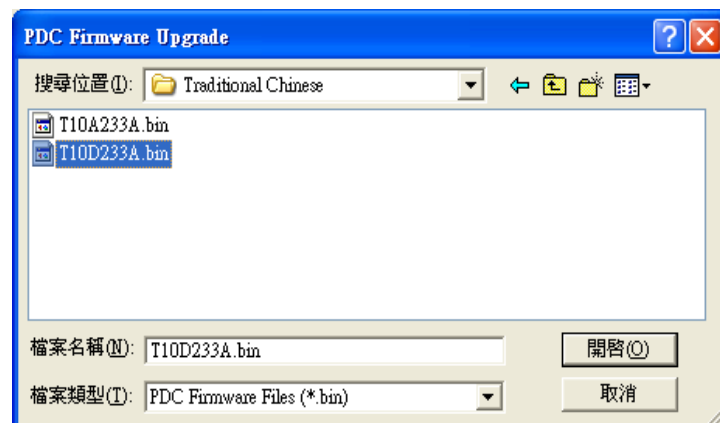
If the function succeeds, the return value is “[REQ_SUCCESS](#)”.
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

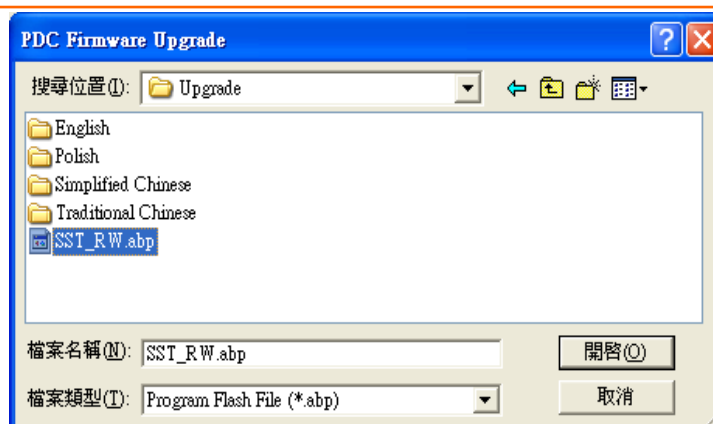
Parameters

strFile

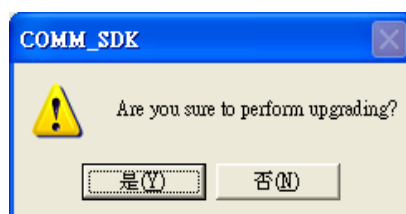
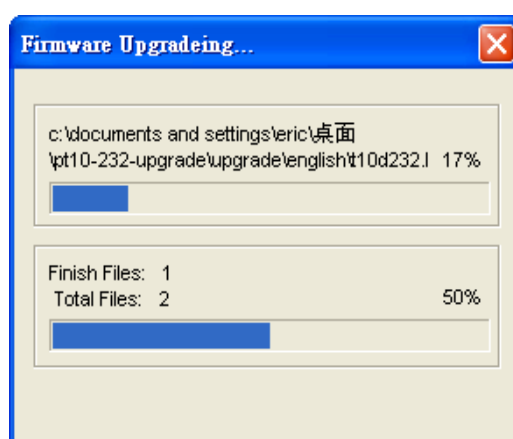
strAbpFile

Remarks





Transmission Dialog:



Example

```
if(Req_FirmWare("T10A230.bin","SST_RW.abp"))
or
if(Req_FirmWare(NULL, NULL) == REQ_SUCCESS)
    AfxMessageBox("The firmware is upgraded successfully.");
else
    AfxMessageBox("The firmware upgrading and failing.");
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_SetDate

Purpose

Ask to set PDC date and time.

Syntax

```
int Req_SetDate(char *strDate);
```

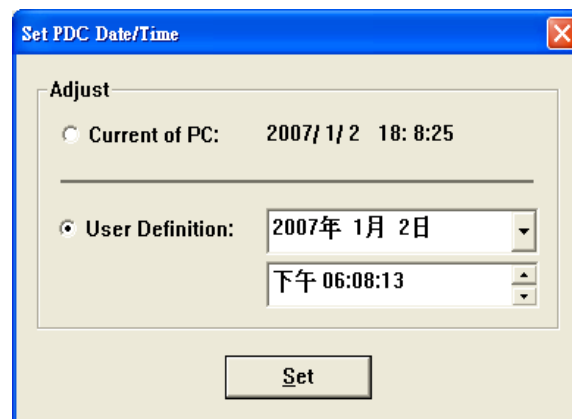
Return Value

If the function succeeds, the return value is "[REQ_SUCCESS](#)".
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

strDate

Remarks



Example

```
nResult = Req_SetDate("20070110093000");
    // Date and Time is 2007/01/10 AM 09:30:00
or
nResult = Req_SetDate(NULL);
    // To call internal date and time dialog to set date and time
if(nResult != REQ_SUCCESS)
    AfxMessageBox("It sets for PDC date and time to fail");
else
    AfxMessageBox("Set date and time successfully");
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_SerialNum

Purpose

Ask to obtain a PDC product serial number.

Syntax

```
int Req_SerialNum(char *pSerialNum);
```

Return Value

If the function succeeds, the return decimal integers value is a USB device F/W edition.

If the function fails, the return zero.

Parameters

pSerialNum

[out] Pointer to the buffer that receives the PDC product number.
The buffer length must be bigger than 8 characters.

Remarks

PDC product number format is 8 digits.

Example: "60924492"

The PDC firmware edition "2.32B" or later can support this request.

Example

```
char acSerialNumBuffer[20];
memset(acSerialNumBuffer, 0, 20);
if(Req_SerialNum(acSerialNumBuffer) == REQ_SUCCESS)
    GetDlgItem(IDC_STATIC_SERIALNUM)->SetWindowText(acSerialNumBuffer);
else
{
    GetDlgItem(IDC_STATIC_SERIALNUM)->SetWindowText("");
    AfxMessageBox("Can't get any serial number");
}
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_ConnectionTest

Purpose

Ask to link with PDC.

Syntax

```
int Req_ConnectionTest(void);
```

Return Value

If the function succeeds, the return value is “[REQ_SUCCESS](#)”.
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

None

Remarks

Every Req_XXXXX function will perform the **Req_ConnectionTest** before performing themselves function. Here we recommend user to add this function before your procedure.

Example

```
if(Req_ConnectionTest() != REQ_SUCCESS)
    return ERR_CONNECTING_FAIL;
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#)

[\[Go To Table\]](#)

Req_PDC_Info

Purpose

Inquire PDC relevant information

Syntax

```
int Req_PDC_Info(int* pnPDC_Ver, int* pnTotalSpace_C,
int* pnFreeSpace_C, int* pnTotalSpace_D, int*
pnFreeSpace_D);
```

Return Value

If the function succeeds, the return value is “[REQ_SUCCESS](#)”.
If the function fails, the return a value. To get the error information, please refer below error code table([Table 6-1](#)).

Parameters

pnPDC_Ver

[out] Pointer to the variable that receives the PDC firmware edition number. The number is around 100~9999 as 1.00 version ~ 99.99 version.

pnTotalSpace_C

[out] Pointer to the variable that receives the number of byte for disk C capacity.

If the **pnFreeSpace_C** is “-1”, the **pnTotalSpace_C** will obtain a zero value.

pnFreeSpace_C

[out] Pointer to the variable that receives the number of byte for disk C free space.

If the **pnFreeSpace_C** is "-1", it means the disk C is unformat.

pnTotalSpace_D

[out] Pointer to the variable that receives the number of byte for disk D capacity.

If the **pnFreeSpace_D** is "-1", the **pnTotalSpace_D** will obtain a zero value.

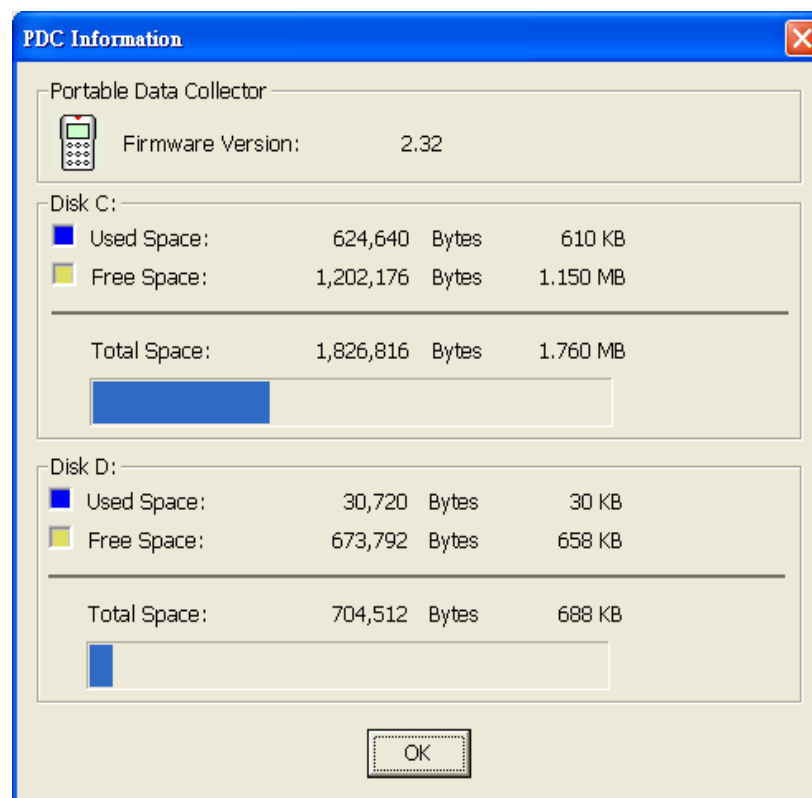
pnFreeSpace_D

[out] Pointer to the variable that receives the number of byte for disk D free space.

If the **pnFreeSpace_D** is "-1", it means the disk D is unformat.

Remarks

If all parameters are NULL, it will display the dialog about PDC information. The dialog likes as below chart.



Example

```
// Show the DLL inside dialog about PDC Information directly
Req_PDC_Info(NULL, NULL, NULL, NULL, NULL);
```

```
// or
// Fetch the PDC information form DLL
int nPDC_Ver, nTotalSpace_C, nFreeSpace_C,
nTotalSpace_D, nFreeSpace_D;
Req_PDC_Info(&nPDC_Ver, &nTotalSpace_C,
&nFreeSpace_C, &nTotalSpace_D, &nFreeSpace_D);
```

See Also

[AL_OpenPort](#), [AL_ClosePort](#), [Req_SerialNum](#), [Req_FormatDisk](#)
[\[Go To Table\]](#)

4.4 Multi-Channels Function:

Preparing...

5 Structure Description:

5.1 _DIR_INFO

Table 5-1 DIR_INFO

typedef struct DIRINFO																	
{																	
char		assFName[8];					// Filename										
char		assExtend[3];					// Extension										
char		usAttrib;					// Attribute										
char		usTime[2];					// Access Time										
char		usDate[2];					// Access Date										
unsigned short		umStart;					// Cluster Start Position										
unsigned long		ulSize;					// file size										
char		endChar[2];					// partition char.(\\x0d\\x0a)										
}_DIRINFO;																	
char assFName[8];		XXXXXXXX					Name for directory or file										
char assExtend[3];		XXX					Extended name for file										
char usAttrib;		0x00 or 0x10					The point is to appoint the assFName for directory.										
		0x20					The point is to appoint the assFName and assExtend for file.										
char		usTime[2];															
Offset		usTime[1]								usTime[0]							
Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		Hour					Minute					Second					
char		usDate[2];															
Offset		usDate[1]								usDate[0]							
Bit		15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		Year-1980					Month					Day					
unsigned short umStart;		Cluster Start Position															
unsigned long ulSize;		file size															
char endChar[2];		partition char.(\\x0d\\x0a)															

6 Error Code List

Table 6-1 Error Code List

Definition Label	Error Code
REQ_SUCCESS	1
ERR_AL_OPENPORT_INVALID_USB_DEVICE	1000
ERR_AL_OPENPORT_INVALID_PORT	1001
ERR_AL_OPENPORT_CREATE_FAIL	1002
ERR_REQ_UPLOADFILE_INVALID_PATH	1100
ERR_REQ_UPLOADMULTIFILE_PDCFILE_INEXISTENCE	1101
ERR_REQ_UPLOADMULTIFILE_PCFILE_INEXISTENCE	1102
ERR_FILEUPLOAD_FAIL	1110
ERR_REQ_DOWNLOADMULTIFILE_PDCFILE_INEXISTENCE	1200
ERR_REQ_DOWNLOADMULTIFILE_PCFILE_INEXISTENCE	1201
ERR_DIR_FAIL	1300
ERR_DIR_DISKC_UNFORMAT	1301
ERR_DIR_DISKD_UNFORMAT	1302
ERR_DIR_DISK_UNFORMAT	1303
ERR_SETDATETIME_FAIL	1310
ERR_GETSERIALNUM_FAIL	1320
ERR_IMPORT_BUF_ERROR	1321
ERR_FWUPGRADE_FAIL	1330
ERR_ASKPCDINFO_FAIL	1340
ERR_GETROMWRITER_ID_FAIL	1350
ERR_FILEDEL_FAIL	1360
ERR_DISKFORMAT_FAIL	1370
ERR_NO_SUPPORT_DISK	1371
ERR_DFU_BINFILE_ERROR	1380
ERR_DFU_BINFILE_EXTENSION_ERROR	1381
ERR_DFU_ROMFILE_EXTENSION_ERROR	1382
ERR_DFU_ROMID_ERROR	1383
ERR_DFU_ROMID_NOTSUPPORT	1384
ERR_DFU_ROMTYPE_NOTSUPPORT	1385
ERR_DFU_DOWNLOAD_FAIL_ROMFILE	1386
ERR_DFU_DOWNLOAD_FAIL_BINFILE	1387
ERR_DFU_BREAK_OPENFILE	1388
ERR_DFU_INEXISTENCE_BINFILE	1389
ERR_DFU_INEXISTENCE_ROMFILE	1390

ERR_DFU_BREAK_UPGRADE	1391
ERR_PROGESS_DLG_SHOWFAIL	1400
ERR_AL_MAKEPREFIXPACKET	1450
ERR_AL_READPREFIXPACKET_RESPONSE	1451
ERR_AL_CHKPREFIXPACKET_RESPONSE	1452
ERR_AL_SENDFILE_BREAK	1453
ERR_AL_MAKEDATAPACKET	1454
ERR_AL_READDATAPACKET_RESPONSE	1455
ERR_AL_RETRY_FAIL_1	1456
ERR_AL_RETRY_FAIL_2	1457
ERR_NEED_HEAD	1500
ERR_CONNECTING_FAIL	2000

7 Edition History:

COMM_SDK 3.0.0.0

- Remove: [GetUSBDataLen] function. Change to [\[USB_GetDeviceName\]](#) function.
- Remove: [GetUSBData] function. Change to [\[USB_GetDeviceName\]](#) function.
- Remove: [GetALDIIVersion] function. Change to [\[GetDIIVersion\]](#) function.
- Add: [\[Req_FileExist\]](#) new function.
- Add: [\[Req_DirectoryExist\]](#) new function.
- Add: [\[Req_FirmWare\]](#) new function.
- Add: [\[Req_DownloadMultiFile\]](#) new function.
- Add: [\[Req_UploadMultiFile\]](#) new function.
- Add: [\[Req_ConnectionTest\]](#) new function.
- Add: [\[Req_SerialNum\]](#) new function.

COMM_SDK 3.0.0.1

- Add: [\[SetMessageDialog\]](#) new function.
- Add: [\[USB_GetDeviceVer\]](#) new function.
- Add: [\[Req_PDC_Info\]](#) new function.
- Fixed: [\[AL_OpenPort\]](#) return value.
- Fixed: [\[OpenPort\]](#) return value.
- Fixed: [\[Req_DirDataLen\]](#) return value.
- Fixed: [\[Req_Dir\]](#) return value.
- Modify: [\[Req_FirmWare\]](#) add internal procedure dialogs.
- Modify: [\[Req_SetDate\]](#) add internal procedure dialogs.

COMM_SDK 3.0.0.2

- Fixed: [\[Req_Dir\]](#) return value.
- Add: [\[SetTransmissionDialog\]](#) new function.

COMM_SDK 3.0.0.3

- Fixed: [\[Transmission Dialog\]](#) Make a task to perform the dialog.
- Fixed: [\[Transmission Dialog\]](#) crash AP reason when break file transmission.
- Fixed: [\[Req_FirmWare\]](#) crash AP reason when break processing.

8 Example PortTest 2.3:

8.1 Main Function:

This example is mainly use and designing to the single channel , every service demand offer, with easily understood way appear , user consult every function button corresponding procedure yard that reach only, can transplant to one's own application program easily , it accords with one's own demand or derive the edition here.

8.2 Development Notice:

8.2.1 This Application links with COMM_SDK.lib to match corresponding import function.

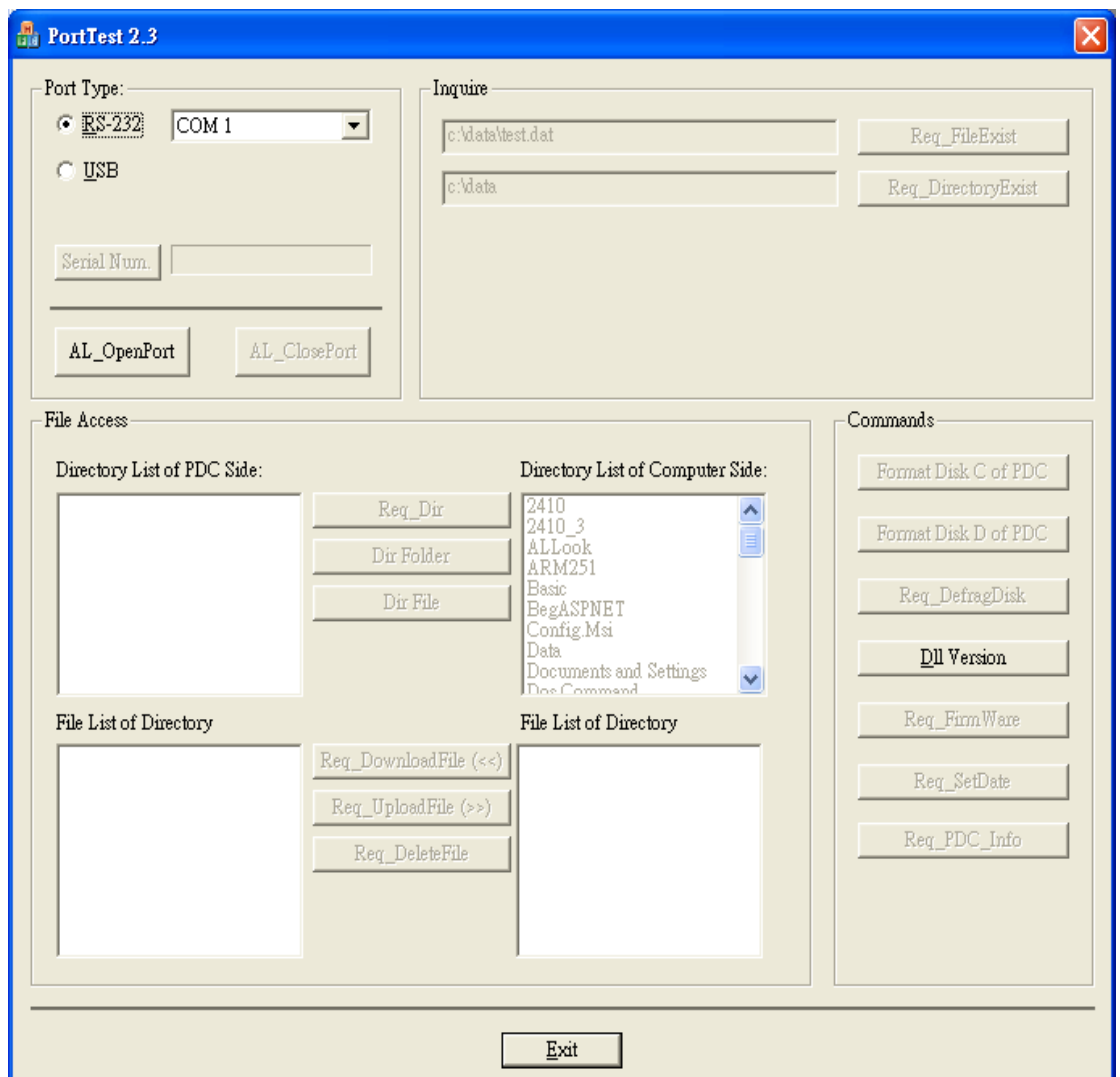
8.2.2 This Application provides two development environments. First workspace is VC 6.0; Secondly workspace is VC .NET 2003. If user needs VC .NET 2005 workspace, please take 200. Workspace file to convert to 2005 workspace file.

8.2.3 In the every workspace, we provide four configurations as “Debug”, “Release”, “SDK Debug”, “SDK Release”. Only two configurations can be used in developing that are “SDK Debug” and “SDK Release”, because we only provide address file as *.lib for the two configurations. So please set active configuration in those two configurations as “SDK Debug”, “SDK Release”.

8.2.4 All EXE files are built in BIN directory.

8.2.5 Inside of DLL provides a “_DIRINFO” structure that is for file system function. Please Refer [DIR_INFO](#)

8.3 Main Picture:



By “AL_OpenPort” button first, please it comes to open all functions.

8.4 PortType picture is explained :

- RS-232: COM1~COM9
- USB: CRD -10-XXXXXXXX
- SerialNum. : PDC products serial number
- AL_OpenPort/AL_ClosePort: Open or close the channel.

Port Type:

☐ RS-232

☒ USB

Cradle F/W Version:

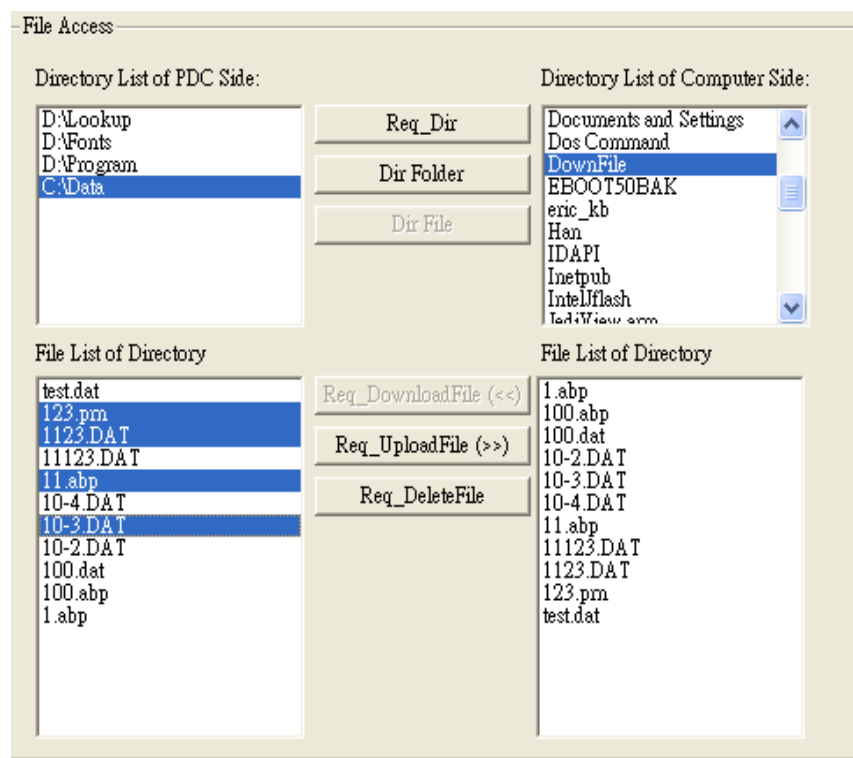
8.5 Inquire Picture:

Inquire

- Req _ FileExist: Inquire about PDC file; please import the route of PDC and shelf name to inquire about in the left editor box.
- Req _ DirectoryExist: Inquire about PDC catalogue, please input the route of PDC and catalogue name to inquire about in the left editor box.

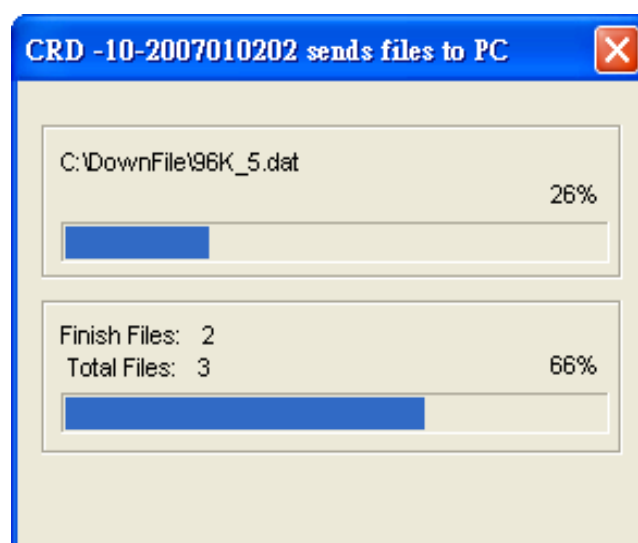
8.6 File Access Picture:

Can inquire about the catalogues of PDC or PC, can upload or download to the appointed catalogue directly after choosing, and support multi-files to be transmitted.



8.7 Transmission Dialog:

The state communication dialog while offering one transmission file, tell the present state.



8.8 Commands Picture:

Send out the single request function to PDC.



8.9 Edition History:

8.9.1 PortTest 2.1

- Add: [Cradle F/W Version] Item.
- Add: [Req_PDC_Info] Button Function.

8.9.2 PortTest 2.2

- Fixed: [OnButReqDir] return value of Req_Dir().

8.9.3 PortTest 2.3

- Remake workspace file for VC 6.0 and VC .NET 2003