1 Files

PkDecode.h –head file for procedure declarations, constant definitions and macros
PkDecode .dll – dynamic library for decoding PDF417
PkDecode .lib – import library
Sample for Visual C++ 6.0
SDK manual

2 Data Structures

PKPDF417BAR

The **PKPDF417BAR** structure is used for decoding PDF417 barcode from an image.

```
typedef struct tagPKPDF417BAR
{
    BYTE*   pImg;
    int     width;
    int     height;
    char    data[3000];
    int     dataLen;
} PKPDF417BAR;
```

Members:

pImg

Pointer to 8 bits gray image buffer.

Width

Image's width in pixels.

Height

Image's height in pixels.

data

The Buffer containing the barcode’s data.

dataLen

The length of the barcode’s data in bytes.

3 Functions

The PkDecode.dll includes five functions:

DllImport void PkPDF417Init(PKPDF417BAR* pBar)
DllImport int PkLoadBmpfile(const char* fileName, PKPDF417BAR* pBar)
DllImport int PkPDF417Decode(PKPDF417BAR* pBar)
DllImport int PkShowBarImage(PKPDF417BAR* pBar, CDC* pDC, int startX, int startY,

DllImport void PkPDF417Init(PKPDF417BAR* pBar)

Parameters

pBar

Pointer to a PKPDF417BAR structure.

Return Values

none

Remarks

Initializes a PKPDF417BAR structure.

DllImport int PkLoadBmpfile(const char* fileName, PKPDF417BAR* pBar)

Parameters

fileName

Name of the file.

pBar

Pointer to a PKPDF417BAR structure.

Return Values

PK_FILE_OPEN_ERROR

Failed to open the file.

PK_FILE_READ_ERROR

Error while reading the file.

PK_FILE_NOT_BMP

The file is not a Bmp file.

PK_FILE_NOT_148BIT

Isn't a 1 bit ,4 bits or 8 bits per pixel image

PK_FILE_COMPRESS_INVALID

Compression mode of the file is invalid.

PK_MEMORY_ALLOC_ERROR

Cannot allocate memory to store image.

PK_SUCCESS

Loading the Bmp file is successful.

Remarks

The function loads 1 bit,4 bits or 8 bits Bmp file and converts them to 8 bits per pixel gray image.If the image contains color information, the function will discards color information and the gray value of the pixel is (red+green+blue)/3.

If successful, the width member of PKPDF417BAR is the image's width. The height member of PKPDF417BAR is the image's height. And the pImg member of PKPDF417BAR points to the image buffer. If failing to load the file, the pImg member of PKPDF417BAR is NULL.

The memory containing the image buffer is allocated by the function automatically. The caller needn’t allocate it. If the pImg member of PKPDF417BAR isn’t NULL before calling the function, the function will free the memory and reallocate it. It’s necessary to call the PkPDF417MemoryFree function to free the image buffer before exiting the application.

DllImport int PkPDF417Decode(PKPDF417BAR* pBar)

Parameters

pBar

Pointer to a PKPDF417BAR structure.

Return Values

PK_MEMORY_INVALID

Image buffer in PKPDF417BAR structure is NULL.

PK_PARAMETERS_INVALID

Image’s width or height in PKPDF417BAR structure is invalid.

PK_MEMORY_ALLOC_ERROR

Cannot allocate memory for internal operations(memory allocated inside function for decoding).

PK_SUCCESS

Decoding the barcode is successful.

PK_FAIL

Failed to decode the barcode .

Remarks

The PkPDF417Decode function decodes PDF417 barcode from an 8 bits per pixel gray image. After decoding, the dataLen member of the PKPDF417BAR structure is the length of the data in bytes and the data member of the PKPDF417BAR structure contains the barcode’ data.

DllImport int PkShowBarImage(PKPDF417BAR* pBar, CDC* pDC, int startX, int startY, float scale)

Parameters

pBar

Pointer to a PKPDF417BAR structure.

PDC

Pointer to the device context used for displaying the image.

startX

Specifies the x-coordinate in pixels, of the upper-left corner of the destination rectangle.

startY

Specifies the y-coordinate in pixels, of the upper-left corner of the destination rectangle.

scale

Scale used to stretches or compresses the image. The destination width=(pBar->width)*scale, the destination height=(pBar->height) *scale.

Return Values

PK_SUCCESS

Displaying the image is successful.

PK_MEMORY_INVALID

Image buffer in PKPDF417BAR structure is NULL.

PK_DC_INVALID

Device context is NULL.

PK_PARAMETERS_INVALID

Image's width or height in PKPDF417BAR structure is invalid.

Remarks

The PkShowBarImage function displays an 8 bits gray image. If the value of scale isn't equal to 1.0, the function will stretch or compress the image according to the value of scale.

DllImport void PkPDF417MemoryFree(PKPDF417BAR* pBar)

Parameters

pBar

Pointer to a PKPDF417BAR structure.

Return Values

none

Remarks

Free the memory allocated by the engine (memory allocated for storing image buffer in PkLoadBmpfile function). The PkPDF417MemoryFree function must be called before exiting application.

4 Integration Examples

Example 1

Decode barcode from a Bmp file:

```
PKPDF417BAR bar;
PkPDF417Init (&bar);
if( PkLoadBmpfile(“1.bmp”, &bar)!=PK_SUCCESS )
{
    AfxMessageBox ("Failed to load Bmp file");
    PkPDF417MemoryFree (&bar);
    return;
}
if( PkPDF417Decode(&bar)!=PK_SUCCESS )
```

```
        AfxMessageBox("Failed to decode the barcode");
    else
        AfxMessageBox(bar.data);
    PkPDF417MemoryFree(&bar);
```

Example 2

Decode barcode from an 8 bits gray image in memory:

```
    PKPDF417BAR bar;
    PkPDF417Init(&bar);
    //Here width is the image’s width;height is the image’s height;
    //imgBuf points to the image buffer.
    bar.width=width;
    bar.height=height;
    bar.pImg=new BYTE[width*height];
    if (bar.pImg==NULL)
        return;
    memcpy ( bar.pImg, imgBuf, width*height*sizeof(BYTE) );
    if ( PkPDF417Decode(&bar)!=PK_SUCCESS )
        AfxMessageBox ("Failed to decode the barcode ");
    else
        AfxMessageBox (bar.data);
    PkPDF417MemoryFree (&bar);
```

Example 3

Decode barcode from several Bmp files:

```
    PKPDF417BAR bar;
    PkPDF417Init(&bar);
    if( PkLoadBmpfile(“1.bmp”, &bar)!=PK_SUCCESS )
    {
        AfxMessageBox("Failed to load Bmp file");
        PkPDF417MemoryFree(&bar);
        return;
    }
    if ( PkPDF417Decode(&bar)!=PK_SUCCESS )
        AfxMessageBox ("Failed to decode the barcode ");
    else
        AfxMessageBox (bar.data);
    //Here you needn’t call PkPDF417MemoryFree to free the memory allocated for 1.bmp.
    //PkLoadBmpfile will free the memory for 1.bmp automatically and reallocate memory for
    //2.bmp.
```

```
if ( PkLoadBmpfile(“2.bmp”, &bar)!=PK_SUCCESS )
{
        AfxMessageBox ("Failed to load Bmp file");
        PkPDF417MemoryFree (&bar);
        return;
}
if ( PkPDF417Decode(&bar)!=PK_SUCCESS )
        AfxMessageBox("Failed to decode the barcode ");
else
        AfxMessageBox(bar.data);
//decode from other Bmp files
.
.
.
PkPDF417MemoryFree(&bar);
```