

BitBlt

The **BitBlt** function performs a bit-block transfer of the color data corresponding to a rectangle of pixels from the specified source device context into a destination device context.

```
BOOL BitBlt(  
    HDC hdcDest,           // handle of destination device context  
    int nXDest,             // x-coordinate of destination rectangle's upper-left corner  
    int nYDest,             // y-coordinate of destination rectangle's upper-left corner  
    int nWidth,             // width of destination rectangle  
    int nHeight,            // height of destination rectangle  
    HDC hdcSrc,            // handle of source device context  
    int nXSrc,              // x-coordinate of source rectangle's upper-left corner  
    int nYSrc,              // y-coordinate of source rectangle's upper-left corner  
    DWORD dwRop            // raster operation code  
);
```

Parameters

hdcDest

Identifies the destination device context.

nXDest

Specifies the logical x-coordinate of the upper-left corner of the destination rectangle.

nYDest

Specifies the logical y-coordinate of the upper-left corner of the destination rectangle.

nWidth

Specifies the logical width of the source and destination rectangles.

nHeight

Specifies the logical height of the source and the destination rectangles.

hdcSrc

Identifies the source device context.

nXSrc

Specifies the logical x-coordinate of the upper-left corner of the source rectangle.

nYSrc

Specifies the logical y-coordinate of the upper-left corner of the source rectangle.

dwRop

Specifies a raster-operation code. These codes define how the color data for the source rectangle is to be combined with the color data for the destination rectangle to achieve the final color.

The following list shows some common raster operation codes:

Value	Description
BLACKNESS	Fills the destination rectangle using the color associated with index 0 in the physical palette. (This color is black for the default physical palette.)
DSTINVERT	Inverts the destination rectangle.
MERGECOPY	Merges the colors of the source rectangle with the specified pattern by using the Boolean AND operator.
MERGEPAINT	Merges the colors of the inverted source rectangle with the colors of the destination rectangle by using the Boolean OR operator.
NOTSRCCOPY	Copies the inverted source rectangle to the destination.
NOTSRCERASE	Combines the colors of the source and destination rectangles by using the Boolean OR operator and then inverts the resultant color.

PATCOPY	Copies the specified pattern into the destination bitmap.
PATINVERT	Combines the colors of the specified pattern with the colors of the destination rectangle by using the Boolean XOR operator.
PATPAINT	Combines the colors of the pattern with the colors of the inverted source rectangle by using the Boolean OR operator. The result of this operation is combined with the colors of the destination rectangle by using the Boolean OR operator.
SRCAND	Combines the colors of the source and destination rectangles by using the Boolean AND operator.
SRCCOPY	Copies the source rectangle directly to the destination rectangle.
SRCERASE	Combines the inverted colors of the destination rectangle with the colors of the source rectangle by using the Boolean AND operator.
SRCINVERT	Combines the colors of the source and destination rectangles by using the Boolean XOR operator.
SRCPAINT	Combines the colors of the source and destination rectangles by using the Boolean OR operator.
WHITENESS	Fills the destination rectangle using the color associated with index 1 in the physical palette. (This color is white for the default physical palette.)

Return Value

If the function succeeds, the return value is TRUE.

If the function fails, the return value is FALSE. To get extended error information, call [GetLastError](#).

Remarks

If a rotation or shear transformation is in effect in the source device context, **BitBlt** returns an error. If other transformations exist in the source device context (and a matching transformation is *not* in effect in the destination device context), the rectangle in the destination device context is stretched, compressed, or rotated as necessary.

If the color formats of the source and destination device contexts do not match, the **BitBlt** function converts the source color format to match the destination format.

When an enhanced metafile is being recorded, an error occurs if the source device context identifies an enhanced-metafile device context.

Not all devices support the **BitBlt** function. For more information, see the RC_BITBLT raster capability entry in **GetDeviceCaps**.

BitBlt returns an error if the source and destination device contexts represent different devices.