

استفاده از تابع *BitBlt* در ویژوال بیسیک

نام تابع **BitBlt** مخفف **Bit Block Transfer** بوده و یکی از توابع **API** ویندوز واقع در کتابخانه **GDI32.dll** می باشد. با استفاده از آن می توان یک ناحیه مستطیلی از تصویری را به ناحیه دیگری با همان ابعاد انتقال داد. آن شامل کپی کردن، ترکیب و ممزوج کردن تصاویر می باشد.

```
Public Enum RasterOps
```

```
    SRCCOPY = &HCC0020
```

```
    SRCAND = &H8800C6
```

```
    SRCINVERT = &H660046
```

```
    SRCPAINT = &HEE0086
```

```
    SRCERASE = &H4400328
```

```
    WHITENESS = &HFF0062
```

```
    BLACKNESS = &H42
```

```
End Enum
```

```
Declare Function BitBlt Lib "gdi32" ( _
```

```
    ByVal hDestDC As Long, _
```

```
    ByVal x As Long, _
```

```
    ByVal y As Long, _
```

```
    ByVal nWidth As Long, _
```

```
    ByVal nHeight As Long, _
```

```
    ByVal hSrcDC As Long, _
```

```
    ByVal xSrc As Long, _
```

```
    ByVal ySrc As Long, _
```

```
    ByVal dwRop As RasterOps _
```

```
) As Long
```

توضیح آرگومان ها :

ByVal hDestDC As Long, _

..

ByVal hSrcDC As Long, _

مقادیری را که اینجا مورد استفاده قرار می دهیم hDC شیء مقصد (hDestDC) و hDC شیء منبع (hSrcDC) می باشد. برای اینکه ویندوز بداند که تصویر را باید از کجا (منبع) به محلی دیگر (مقصد) کپی کند نیاز به یک handle دارد که در اینجا hDC نام گرفته است. خوشبختانه جعبه تصویر (PictureBox) ویژوال بیسیک این خاصیت را پشتیبانی می کند و به سادگی می توان بجای استفاده از چندین تابع API از PictureBox.hDC استفاده نمود.
(HDC= Hande of Device Context)

نکته : هنگامیکه خاصیت AutoRedraw جعبه تصویر ویژوال بیسیک به True تنظیم شده باشد هر گونه تغییری در تصویر آن فقط در حافظه نگهداری خواهد شد و روی صفحه نمایش اعمال نمی گردد مگر اینکه از خاصیت Refresh برای به روز در آوردن آن استفاده شود.

پارامترهای X,Y :

X : مختصات سمت چپ (بالا-چپ) مستطیل مقصد است.

Y : مختصات بالا (بالا-چپ) مستطیل مقصد است.

ایندو به مقادیر Object.Top و Object.Left بسیار شبیه هستند.

نکته : در این تابع و تمام توابع API ویندوز مقادیر اینگونه فقط بر حسب Pixel هستند. راهی مطمئن که می توان بوسیله آن از Pixel بودن واحد اندازه گیری طول در تصاویر اطمینان حاصل نمود این است که قرار داده شود :

PictureBox.ScaleMode=3

پارامترهای xSrc,ySrc

xSrc : مختصات سمت چپ (بالا-چپ) مستطیل منبع است.

ySrc : مختصات بالا (بالا-چپ) مستطیل منبع است.

پارامترهای nHeight,nWidth

nHeight : ارتفاع مستطیل تصویر مقصد است که باید ترکیب شود.

nWidth: عرض مستطیل تصویر مقصد است که باید ترکیب شود.

آخرین پارامتر **dwRop**

کار تابع **BitBlt** فقط به کپی ساده یک تصویر روی تصویر یا محلی مستطیلی خلاصه نمی شود بلکه به ما اجازه اعمال نمودن عملگرهای منطقی و ریاضی را روی دو تصویر منبع و مقصد، نیز می دهد. برای مثال ایجاد شفافیت در تصاویر و یا عکس نمودن رنگ ها.

پارامتر **dwRop** نحوه انتقال تصاویر را از مبدا به مقصد معین می نماید:

SRCCOPY: تصویر مبدا را روی تصویر مقصد کپی می کند.

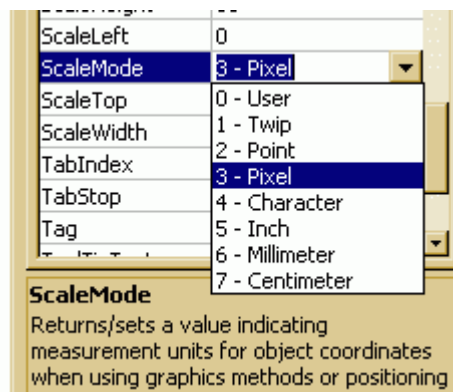
SRCAND: تصویر منبع را با تصویر مقصد بوسیله اپراتور AND ترکیب می کند.

SRCINVERT: تصویر منبع را با تصویر مقصد بوسیله اپراتور XOR ترکیب می کند.

SRCPAINT: تصویر منبع را با تصویر مقصد بوسیله اپراتور OR ترکیب می کند.

چند مثال:

تذکر: همانطور که ذکر شد، هر جعبه تصویری را که در مثالهای زیر بر روی فرم قرار می دهید، **ScaleMode** آنرا به **Pixel** تغییر دهید.



مثال ۱- کاشی کاری!

دو جعبه تصویر را روی فرم قرار دهید و سپس تصویری دلخواه را در یکی از آنها نمایش دهید.

```
Public Sub TilePictures(picTile As PictureBox, _  
    PicDestination As PictureBox)  
' TilePicture - tiles the given bitmap bmp, jpg, ... into the given  
' window  
picTile.AutoRedraw = True
```

```

PicDestination.AutoRedraw = True

Do
    Call BitBlt(PicDestination.hDC, PicDestination.CurrentX, _
        PicDestination.CurrentY, _
        picTile.ScaleHeight, picTile.ScaleWidth, _
        picTile.hDC, 0, 0, vbSrcCopy)
    DoEvents
    PicDestination.Refresh
    PicDestination.CurrentX = PicDestination.CurrentX + _
        picTile.ScaleWidth
    If PicDestination.CurrentX >= PicDestination.ScaleWidth Then
        PicDestination.CurrentX = 0
        PicDestination.CurrentY = PicDestination.CurrentY + _
            picTile.ScaleHeight
    End If
    If PicDestination.CurrentY >= PicDestination.ScaleHeight Then _
        Exit Do
Loop

End Sub

Private Sub Form_Load()
    TilePictures Picture1, Picture2
End Sub

```



مثال ۲- تلفیق متن و تصویر

در تمام مثالها شماره جعبه های متن از بالا به پایین زیاد می شود.

```

Public Sub Show_Textured_Text(picText As PictureBox, _
    picTexture As PictureBox, PicDestination As PictureBox, _
    X As Long, Y As Long, strText As String, strFontSize As String, _
    strFont As String, strBold As Boolean, strItalic As Boolean)

```

```

Dim w, h, t, hDcMem, hDcMem2, hdcScreen
picText.AutoRedraw = True
picTexture.AutoRedraw = True
PicDestination.AutoRedraw = True

picText.Font = strFont
picText.FontSize = strFontSize
picText.FontBold = strBold
picText.FontItalic = strItalic
picText.BackColor = RGB(255, 255, 255)
picText.ForeColor = RGB(0, 0, 0)

picText.Print strText

hDcMem = picTexture.hDC
hDcMem2 = picText.hDC
hdcScreen = PicDestination.hDC

w = PicDestination.Width
h = PicDestination.Height

t = BitBlt(hdcScreen, X, Y, w, h, hDcMem, 0, 0, vbSrcInvert)
t = BitBlt(hdcScreen, X, Y, w, h, hDcMem2, 0, 0, vbSrcAnd)
t = BitBlt(hdcScreen, X, Y, w, h, hDcMem, 0, 0, vbSrcInvert)

picText.Refresh
picTexture.Refresh
PicDestination.Refresh

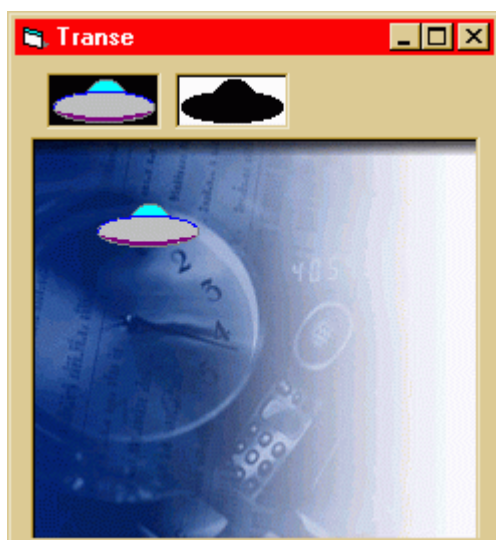
End Sub

Private Sub Form_Load()
TilePictures Picture1, Picture2
Show_Textured_Text Picture3, Picture2, Picture4, _
    10, 10, "Textured Text", "25", "Verdana", True, False
End Sub

```



مثال ۳- نشان دادن تصاویر غیر چهار گوش بر روی تصویری دیگر
 گاهی اوقات لازم می شود که تصاویر را از حالت مستطیلی خارج کنیم . برای اینکار باید در
 ابتدا یک ماسک از تصویر ایجاد نمود . ماسک تصویری است دو رنگ (سیاه و سفید) از
 تصویر اول ، قسمت های زائد به رنگ سفید و شکل اصلی به رنگ سیاه در آن ظاهر می گردد.



```

Public Sub Show_Transparent_Picture(picSource As PictureBox, _
    picMask As PictureBox, PicDestination As PictureBox, _
    X As Long, Y As Long)

    picSource.AutoRedraw = True
    picMask.AutoRedraw = True
    PicDestination.AutoRedraw = True

    picSource.AutoSize = True
    picMask.AutoSize = True
    PicDestination.AutoSize = True

    'Colour , Colour, Mask
    'Colour , Colour, Colour

    ' Paint the Mask onto the Destination using AND operator.

    Call BitBlt(PicDestination.hDC, X, Y, _
        picSource.ScaleWidth , _
        picSource.ScaleHeight , _
        picMask.hDC, 0, 0, &H8800C6)

    ' Paint the Source onto the Destination using XOR operator.
    Call BitBlt(PicDestination.hDC, X, Y, _
        picSource.ScaleWidth , _
        picSource.ScaleHeight, _
        picSource.hDC, 0, 0, &H660046)

    picSource.Refresh
    picMask.Refresh
    PicDestination.Refresh

End Sub

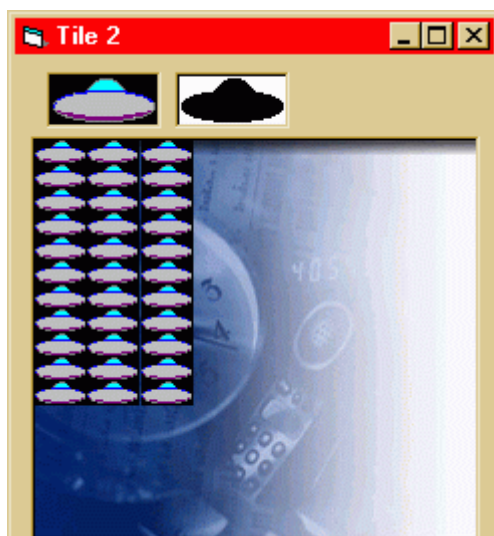
Private Sub Form_Paint()
    Show_Transparent_Picture _
        Picture1, Picture2, Picture3, 30, 30

End Sub

```

مثال ۴- کاشی کاری ۲

خود ویژگی‌ها هم تابعی اینچنینی دارد که در مثال زیر توضیح داده شده است.



```

Public Sub TileBitmapsProcessBar( _
ByRef ctl As Object, ByRef pic As PictureBox)
    Dim lRet As Long
    Dim nOuterCount As Integer
    Dim nInnerCount As Integer

    ctl.AutoRedraw = True
    pic.AutoRedraw = True

    For nOuterCount = _
        0 To (ctl.Width \ pic.Width \ 2) + 1
        For nInnerCount = _
            0 To (ctl.Height \ pic.Height \ 2) + 1
            ctl.PaintPicture pic.Picture, _
                0 + nOuterCount * pic.ScaleWidth \ 2, _
                0 + nInnerCount * _
                    pic.ScaleHeight \ 2, _
                pic.ScaleWidth / 2, pic.ScaleHeight / 2
            Next
        Next

        ctl.Refresh
        pic.Refresh
    End Sub

Private Sub Form_Paint()
    TileBitmapsProcessBar Picture3, Picture1
End Sub

```

مثال ۵- بزرگ و کوچک کردن تصاویر

با استفاده از تابع زیر می توان اندازه تصاویر را تغییر داد :

```

Declare Function StretchBlt Lib "gdi32" ( _
    ByVal hdc As Long, ByVal x As Long, _
    ByVal y As Long, ByVal nWidth As Long, _
    ByVal nHeight As Long, ByVal hSrcDC As Long, _
    ByVal xSrc As Long, ByVal ySrc As Long, _
    ByVal nSrcWidth As Long, ByVal nSrcHeight As Long, _

```

```
ByVal dwRop As Long) As Long
```

```
Public Sub StretchPicture( _  
picSource As PictureBox, _  
PicDestination As PictureBox, stretchX As Long, _  
stretchY As Long _  
)  
'this sub stretches the picSource into the picDestination  
'by : Vahid Nasiri  
  
PicDestination.AutoRedraw = True  
picSource.AutoRedraw = True  
  
PicDestination.Cls  
StretchBlt PicDestination.hdc, 0, 0, _  
stretchY, _  
stretchX, _  
picSource.hdc, _  
0, 0, _  
picSource.ScaleWidth, picSource.ScaleHeight, _  
vbSrcCopy  
  
PicDestination.Refresh  
  
End Sub
```

Ref:

www.vbexplorer.com