



Help for BarCod

[Properties](#)

[Events](#)

Registration Information

Order Form

Getting Custom Controls Written

Description

BarCod makes bar code display really easy. Just pick the orientation, set the size, and pick the bar code style. Then, set the caption to whatever you want the bars to be.

BarCod uses the Caption property to determine what to display. ForeColor is used for the color of the Bars. BackColor is used for the color of the background. If BarCod cannot fit the entire bar code in the control window, it will fill the entire control with the background color.

Under VB 3.0, BarCod will bind to a data control. It uses the [DataField](#), [DataSource](#), and [DataChanged](#) properties.

File Name

BARCOD1.VBX

Object Type

BarCode

Remarks

Distribution Note When you develop and distribute an application that uses BarCod, you should install the file BARCOD2.VBX into the users Windows SYSTEM directory. BarCod has version information built into it. So, during installation, you should ensure that you are not overwriting a newer version of BarCod.

Properties

All of the properties that apply to this control are in this table. Properties that have special meaning for this control or that only apply to this control are marked with an asterisk (*).

| | | |
|--------------------|-----------------------|--------------------------|
| <u>BackColor</u> | <u>HelpContextID</u> | <u>*PrinterScaleMode</u> |
| <u>*BarWidth</u> | <u>Height</u> | <u>*PrinterTop</u> |
| <u>Caption</u> | <u>hWnd</u> | <u>*PrinterWidth</u> |
| <u>DataChanged</u> | <u>Index</u> | <u>*Style</u> |
| <u>DataField</u> | <u>Left</u> | <u>Tag</u> |
| <u>DataSource</u> | <u>Name</u> | <u>Top</u> |
| <u>*Direction</u> | <u>Parent</u> | <u>Visible</u> |
| <u>*Displayed</u> | <u>*PrinterHDC</u> | <u>Width</u> |
| <u>Enabled</u> | <u>*PrinterHeight</u> | |
| <u>ForeColor</u> | <u>*PrinterLeft</u> | |

Caption is the default value for the control.

BarWidth Property

[See Also](#)

[Example](#)

Description

Determines the width of the bars in the bar code.

Usage

[form.][control.]BarWidth[= integer]

Remarks

This represents the width of the "narrow" bar in the code selected (see [Style](#)). If this is set to zero (0), BarCod expands the bar code to fit in the control.

Data Type

Integer

See Also

Properties:

[Direction](#)

[Style](#)



BarWidth Property Example

In this example, the program will vary the width of the bars. To try this example, paste the code into the Declarations section of a form that contains a horizontal scroll bar, a label (which shows the current BarWidth property), and a BarCod control (make this control wide). Press F5. Play with the scroll bar to see things change.

```
Sub Form_Load ()
    HScroll1.Value = 0
    HScroll1.Min = 0
    HScroll1.Max = 10

    Label1.Caption = "0"

    BarCod1.Style = 3
    BarCod1.BarWidth = 0
    BarCod1.Caption = "SAMPLE"
End Sub

Sub HScroll1_Change ()
    BarCod1.BarWidth = HScroll1.Value
    Label1.Caption = Format$( HScroll1.Value )
End Sub

Sub HScroll1_Scroll ()
    Call HScroll_Change
End Sub
```

Direction Property

[See Also](#)

[Example](#)

Description

Determines the direction/orientation of the bar code.

Usage

[*form.*][*control.*]**Direction**[= *integer*]

Remarks

The value of this property determines what direction the bar code goes. This property can be one of following values:

| Value | Description |
|-------|---------------|
| 0 | Left-to-right |
| 1 | Right-to-left |
| 2 | Top-to-bottom |
| 3 | Bottom-to-top |

Data Type

Integer (enumerated)

See Also

Properties:

[BarWidth](#)

[Style](#)



Direction Property Example

In this example, the program will vary the direction the bar code is painted/printed. To try this example, paste the code into the Declarations section of a form that contains a horizontal scroll bar, a label (which shows the current Direction property), and a BarCod control (make this control tall and wide). Press F5. Play with the scroll bar to see things change.

```
Sub Form_Load ()
    HScroll1.Value = 0
    HScroll1.Min = 0
    HScroll1.Max = 3

    Label1.Caption = "0"

    BarCod1.Style = 3
    BarCod1.BarWidth = 2
    BarCod1.Direction = 0
    BarCod1.Caption = "SAMPLE"
End Sub

Sub HScroll1_Change ()
    BarCod1.Direction = HScroll1.Value
    Label1.Caption = Format$( HScroll1.Value )
End Sub

Sub HScroll1_Scroll ()
    Call HScroll_Change
End Sub
```


Displayed Property

[See Also](#)

[Example](#)

Description

Returns the actual string displayed in the bar code.

Usage

*[form.]***Displayed**

Remarks

This property holds what was actually displayed. Bar codes dont have all the characters available. Some are purely numeric, some have numbers and letters, etc. This property lets your program know what was actually displayed.

This property is read-only.

Data Type

String

See Also

Properties:

[Caption](#)

[Style](#)



Displayed Property Example

In this example, the program will show what is displayed when various strings are attempted. To try this example, paste the code into the Declarations section of a form that contains a label (which shows the current Displayed property), a text box, and a BarCod control. Press F5. Play with the edit box and see what appears in the label.

```
Sub Form_Load ()
    Label1.Caption = "0"
    Text1.Text = "SAMPLE"

    BarCod1.Style = 3
    BarCod1.BarWidth = 2
    BarCod1.Direction = 0
    BarCod1.Caption = Text1.Text
End Sub

Sub Text1_Change ()
    BarCod1.Caption = Text1.Text
    Label1.Caption = BarCod1.Displayed
End Sub
```

PrinterHDC Property

[See Also](#)

[Example](#)

Description

Causes the bar code to be painted in the device handle specified (usually the printer object).

Usage

[form.][control.]PrinterHDC[= integer]

Remarks

When this property is set, BarCod paints itself into the Windows device context specified. Normally, this is used to print the barcode. To do this, use the hDC property of VBs Printer Object.

To print a bar code, make sure the Caption !JumpKeyword("VB.HLP","caption") property is set correctly. Then, set the PrinterHeight , PrinterLeft , PrinterTop , and PrinterWidth properties. Finally, set BarCod.PrinterHDC to Printer.hDC.

Data Type

Integer

See Also

Objects:

[Printer Object](#)

Properties:

[hDC](#)

[PrinterLeft, PrinterTop, PrinterRight, PrinterBottom](#)

[PrinterScaleMode](#)

[Style](#)

PrinterLeft, PrinterTop, PrinterWidth, and PrinterHeight Properties

[See Also](#)

[Example](#)

Description

Describes the area in which to paint the barcode when printing.

Usage

```
[form.][control.]PrinterLeft[ = x ]  
[form.][control.]PrinterTop[ = y ]  
[form.][control.]PrinterWidth[ = xx ]  
[form.][control.]PrinterHeight[ = ny ]
```

Remarks

These properties describe where to place the barcode when printing. The coordinates used should be in the units specified by the [PrinterScaleMode](#) property.

To print a bar code, make sure the Caption !JumpKeyword("VB.HLP","caption") property is set correctly. Then, set the [PrinterHeight](#) , [PrinterLeft](#) , [PrinterTop](#) , and [PrinterWidth](#) properties. Finally, set BarCod.PrinterHDC to [Printer.hDC](#).

Data Type

Single

See Also

Properties:

[Caption](#)

[PrinterHDC](#)

[PrinterScaleMode](#)

[Style](#)

PrinterScaleMode Property

[See Also](#)

[Example](#)

Description

Determines the units of measurement used by the [PrinterLeft](#), [PrinterTop](#), [PrinterWidth](#), and [PrinterHeight](#) properties.

Usage

[*form.*][*control.*]**PrinterScaleMode**[= *integer*]

Remarks

This property determines what units to use for the properties describing the area to print in. This property can be one of:

| Value | Description |
|-------|---|
| 0 | Pixel (default, smallest unit of printer resolution) |
| 1 | Twip (1440 twips per inch) |
| 2 | Point (72 points per inch) |
| 3 | Pixel (smallest unit of printer resolution) |
| 4 | Character (12 per inch horizontally, 6 per inch vertically) |
| 5 | Inch |
| 6 | Millimeter |
| 7 | Centimeter |

To print a bar code, make sure the [Caption](#) property is set correctly. Then, set the [PrinterHeight](#), [PrinterLeft](#), [PrinterTop](#), and [PrinterWidth](#) properties. Finally, set BarCod.[PrinterHDC](#) to [Printer.hDC](#).

Data Type

Integer (enumerated)

See Also

Properties:

[Caption](#)

[PrinterHDC](#)

[PrinterLeft, PrinterTop, PrinterWidth, PrinterHeight](#)

[Style](#)



Printing Example

In this example, the program will print a bar code on your printer. The bar code and its style are determined by an edit box and a horizontal scroll bar. To try this example, paste the code into the Declarations section of a form that contains a text box, a command button, two labels, a horizontal scroll bar, and a BarCod control. Press F5. Then, set up the bar code using the horizontal scroll bar and the text box. Once you have a bar code you like, press the command button to print it.

```
Sub Form_Load ()
    HScroll1.Min = 0
    HScroll1.Max = 3
    HScroll1.Value = 3

    Text1.Text = "SAMPLE"

    BarCod1.Style = 3
    BarCod1.BarWidth = 2
    BarCod1.Direction = 0
    BarCod1.Caption = Text1.Text

    Label1.Caption = BarCod1.Displayed
    Label2.Caption = "3"
End Sub

Sub Command1_Click ()
    Screen.MousePointer = 11

    Printer.ScaleMode = 1
    Barcode1.PrinterScaleMode = Printer.ScaleMode

    Barcode1.PrinterLeft = 1440
    Barcode1.PrinterTop = 1440
    Barcode1.PrinterWidth = 3 * 1440
    Barcode1.PrinterHeight = 720
    Barcode1.PrinterHDC = Printer.hDC

    Printer.CurrentX = Barcode1.PrinterLeft
    Printer.CurrentY = Barcode1.PrinterTop + Barcode1.PrinterHeight + 180
    Printer.Print Barcode1.Displayed

    Printer.NewPage
    Printer.EndDoc

    Screen.MousePointer = 0
End Sub

Sub Text1_Change ()
    BarCod1.Caption = Text1.Text
    Label1.Caption = BarCod1.Displayed
End Sub

Sub HScroll1_Change ()
    BarCod1.Style = HScroll1.Value
    Label1.Caption = BarCod1.Displayed
    Label2.Caption = Format$( HScroll1.Value )
```

```
End Sub
```

```
Sub HScroll1_Scroll ()  
    Call HScroll1_Change  
End Sub
```

Style Property

[See Also](#)

[Example](#)

Description

Determines what bar code style is used.

Usage

[*form.*][*control.*]**Style**[= *integer*]

Remarks

This property determines the style of bar code used. Each style has a different set of characters that it supports. The styles are:

| Value | Description |
|-------|---|
| 0 | None |
| 1 | <u>2 of 5</u> - numbers only |
| 2 | <u>Interleaved 2 of 5</u> - numbers only |
| 3 | <u>3 of 9</u> - numbers, upper case, and - . * \$ / + % |
| 4 | <u>Codabar (rationalized)</u> - numbers and A B C D - \$: / . + |
| 5 | <u>Extended 3 of 9</u> - numbers, letters, and - . * \$ / + % |
| 6 | <u>Code 128A</u> - upper case, numbers, punctuation, and control characters |
| 7 | <u>Code 128B</u> - upper and lower case, numbers, and punctuation |
| 8 | <u>Code 128C</u> - numbers only |
| 9 | <u>UPC-A</u> - numbers only (11 numbers and a check digit) |

Data Type

Integer (enumerated)

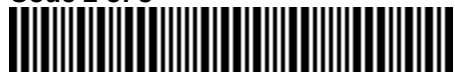
See Also

Properties:

[BarWidth](#)

[Direction](#)

Code 2 of 5



Interleaved 2 of 5



Extended Code 3 of 9



Code 3 of 9



Code 128A



Code 128B



Code 128C



UPC-A



Rationalized Codabar





Style Property Example

In this example, the program will vary style of the bar code used. To try this example, paste the code into the Declarations section of a form that contains a horizontal scroll bar, two labels (the first shows the Style property and the second shows the current Displayedn property), a text box, and a BarCod control. Press F5. Play with the text box and the scroll bar and see what appears in the second label.

```
Sub Form_Load ()
    HScroll1.Min = 0
    HScroll1.Max = 9
    HScroll1.Value = 3

    Text1.Text = "SAMPLE"

    BarCod1.Style = 3
    BarCod1.BarWidth = 2
    BarCod1.Direction = 0
    BarCod1.Caption = Text1.Text

    Label1.Caption = BarCod1.Displayed
    Label2.Caption = "3"
End Sub

Sub Text1_Change ()
    BarCod1.Caption = Text1.Text
    Label1.Caption = BarCod1.Displayed
End Sub

Sub HScroll1_Change ()
    BarCod1.Style = HScroll1.Value
    Label1.Caption = BarCod1.Displayed
    Label2.Caption = Format$( HScroll1.Value )
End Sub

Sub HScroll1_Scroll ()
    Call HScroll1_Change
End Sub
```

Events

All of the events that apply to this control are in this table. Events that have special meaning for this control or that only apply to this control are marked with an asterisk (*).

Click

DbClick

Registration Information

Credits

BarCod was written by James Shields. Inquiries can be sent to 71231,2066 on CompuServe, or mabry@halcyon.com on Internet. If you must send something via U.S. Mail, the address is:

Mabry Software
Post Office Box 31926
Seattle, WA 98103-1926

Registration

You can register this program by sending \$15 (\$17 for international orders) and your address. CompuServe members may register by sending \$10 and their account number (the registered version will be E-mailed to you). CompuServe members may also register this package by going to the SWREG forum. BarCod is registered there. BarCods registration ID number is 1314.

For your convenience, an order form has been provided that you can print out directly from help.

Source Code and Registration

Source code (which includes a registered copy) to this control is available for \$45 (\$50 for international orders). With source code you get a registered version of the control. If you are a CompuServe member, you may get the source code in the Software Registration forum (GO SWREG) for \$40. Its registration number is 1315.

Credit Card Orders

You can order this program with Mastercard, Visa, American Express, or Discover from Public (software) Library by calling 800-242-4775 or 713-524-6394 or by FAX to 713-524-6398 or by CompuServe E-mail to 71355,470. You can also mail credit card orders to PsL at Post Office Box 35705; Houston, TX 77235-5705. THESE NUMBERS ARE FOR CREDIT CARDS ONLY.

BarCod's ID number for this service is 11092. This is good for both the normal registered version, and the source code version.

Any questions about the status of the shipment of the order, refunds, registration options, product details, technical support, volume discounts, dealer pricing, site licenses, etc., must be directed to Mabry Software at 206-634-1443 or FAX at 206-632-0272.

To ensure that you get the latest version, PsL will notify us the day of your order and we will ship the product directly to you.

© Copyright 1993-1994 by James Shields





BarCod Order Form

Use the Print Topic.. command from the File menu to print this order form.

Mail this form to: Mabry Software
Post Office Box 31926
Seattle, WA 98103-1926
Phone: 206-634-1443
Fax: 206-632-0272
BBS: WinDev BBS 206-634-0783
CompuServe: 71231,2066
Internet: mabry@halcyon.com

Where did you get this copy of BarCod?

Ship to:

Phone:

Fax:

E-Mail:

Disk Size: (circle one) $3\frac{1}{2}$ $5\frac{1}{4}$

qty ordered _____ REGISTRATION
\$15 each, postpaid (check or money order in hard currency). Foreign addresses add \$2.00 shipping. No additional shipping charges to Canada or Mexico.

qty ordered _____ SOURCE CODE AND REGISTRATION
\$45 each, postpaid (check or money order in hard currency). Foreign addresses add \$5.00 shipping. No additional shipping charges to Canada or Mexico.

Getting Custom Controls Written

If you or your organization would like to have custom controls written, you can contact me at the following:

James Shields
Mabry Software
Post Office Box 31926
Seattle, WA 98103-1926
Phone: 206-634-1443
Fax: 206-632-0272
BBS: WinDev BBS 206-634-0783
CompuServe: 71231,2066
Internet: mabry@halcyon.com

You can also contact Zane Thomas. He can be reached at:

Zane Thomas
Post Office Box 300
Indianola, WA 98342
CompuServe: 72060,3327

