MicroHelp Custom Controls

Contents

> **Note:** Standard properties & events are described in the document accompanying the control pack. MicroHelp custom properties are noted with an asterisk, whereas standard properties do not have an asterisk.

Other MicroHelp Products for Visual Programmers

MicroHelp was the first company to release add-on products for Microsoft Visual programming languages. We have continued our commitment to Visual programming by offering you a complete line of products for all of your programming needs. The following is a brief description of our current Windows programming products. Note that all products are compatible with Microsoft Visual Basic. Other languages supported are noted in each product's description.

VBTools™ version 3- Contains **over 40** custom controls, including nine that are data-aware! This is the first and foremost Visual Basic custom control package. **BasicPro 1992 Readers' Choice Award. Window Tech Journal "Star Tech"Honorable Mention. Borland C++ compatible! MS Visual C++ compatible!**

MicroHelp 3-D Gizmos™ version 2- Windows Magazine May 1993 Recommended Buy! Not only do you get 18 different controls that have an optional 3-D appearance, we include bound and virtualized list and combo boxes as well as a control that allows you to make ANY VB custom control a bound control. **Borland C++ compatible! MS Visual C++ compatible!**

MicroHelp Muscle version 2 - The only Visual Basic add-on to win **Computer Language Magazine's 1991 Productivity Award.** Muscle contains over 600 assembly language routines that provide you with the speed and functionality not available anywhere else. DOS and Windows low level services, string and array manipulation, date and time functions, bit manipulation and routines that will allow you to create arrays that can use all available Windows memory (including temporary and permanent swap files). **BasicPro 1992 Readers' Choice Award.**

MicroHelp Communications Library - Provides you with everything you need for serial communications, including source code for a working Windows terminal program. Seven different file transfer protocols that work even when your application is minimized, ANSI terminal emulation, and the ability to have upto eight simultaneous communications sessions at ONE TIME! **BasicPro 1992 Readers' Choice Award.**

MicroHelp Network Library - 100% assembly language routines that support Novell, Lantastic and NetBIOS API functions. Now you can access and manipulate all objects in the bindery, add, change and delete items in the print queues as well as redirect all shared resources.

MicroHelp VBXRef™ version 2 - The "ultimate" Project Management tool for Visual Basic. VBXRef is the **ONLY** utility that can actually read the binary VB form file (.FRM). Open up to 10 projects at one time. Copy, move and delete forms, modules, DLL's and even individual procedures! VBXRef also creates a cross reference listing of your entire project, listing all functions, procedures and variables. It even tells you how many times each of these items was referenced in your project as well as where in the project any variable was assigned a value. Supports ALL versions of Visual Basic.

MicroHelp SpellPro™ - A Windows spell checking custom control and DLL. The Standard edition is for Visual Basic only, while the Professional edition also supports C/C++, TPW and SQL WIndows and includes a utility that allows you to create or modify dictionary files. Also available, at a nominal charge, are a Medical dictionary and Legal dictionary. **Borland C++ compatible! MS Visual C++ compatible!**

MicroHelp HighEdit - A true WYSIWYG word processing custom control for Microsoft Windows. The Standard edition is for Visual Basic, while the Professional edition also supports C/C++, TPW and SQL WIndows. **BasicPro 1992 Editors' Choice Award. Borland C++ compatible! MS Visual C++**

compatible!

MicroHelp Report Generator - The easiest and fastest way to design database reports, mailing labels and lists of data. Because it is database independent, you're not locked into a particular database format. It even supports 10 different barcode formats, including UPC.

You can receive product brochures 24 hours a day through our new FastFax™ service. Simply call (404) 591-6454 from you fax machine and follow the voice prompts. For more information and free DEMOS call:

4359 Shallowford Ind Pkwy
Marietta, GA 30066
FAX: (404) 516-1099
FastFax: (404) 591-6454

***In Georgia or Outside the U.S. (404) 516-0899**

MhAlarm

Properties Events Methods

Description

The MhAlarm is an audio-visual "attention getter" that comes in three default styles: telephone, alarm clock or wrist alarm. You can also define your own images. You can display MhAlarm controls on your forms and use the associated bitmaps in place of your form's icon when the form is minimized. The MhAlarm bitmaps are shown below:

File Name

MHAL200.VBX

Class Name

MhIAlarm

Remarks

The MhAlarm control is useful when you want to let the user know that your program has completed a task or needs attention. The ability to display an MhAlarm icon in place of the form's icon is especially helpful when the user minimizes your program to work with other applications.

The control remains in an idle state until you turn it on, using the .RingOn property.

You can use the .WindowState property to instruct the control to display its bitmaps where the icon is normally displayed when a form is minimized. When you replace the form's icon, the MhAlarm control isn't displayed until the alarm triggers for the first time.

If none of the built-in bitmaps suit your needs, you can use the .Picture*x* and .PictureMask*x* properties to specify your own bitmaps.

Note: The default values for most of the custom properties depend on the style of MhAlarm control you choose. Please consult the table shown with the .Style property for a complete list of default values.

Caution: When you set the .Style property, default values are placed into the .RingTime, .PauseTime, .RingTone, .RingMode, .RingLength and .Interval properties. If you wish to use different values for any of those properties, you must explicitly set each one after setting the .Style property.

Note: The .Interval property applies only to the toggling of bitmap displays when the .RingOn property is non-zero. The sound effects are controlled via the other properties having names beginning with "Ring".

Caution: If you want to take advantage of the control's ability to display its bitmaps in place of your form's icon, be sure to set the .AutoSize property to True.

Note: The ZOrder method and .HelpContextID property are valid only for VB versions 2.0 and later.

Properties

AutoSize	BackColor	Enabled
Height	HelpContextID	hWnd
Index	*Interval	Left
MousePointer	Name	Parent
*PauseTime	*Picture1	*Picture2
*Picture3	*PictureMask1	*PictureMask2
*PictureMask3	*RingLength	*RingMode
*RingOn	*RingTime	*RingTone
*Style	TabIndex	Tag
Top	Visible	Width
*WindowState		

Events

Change	Click	DblClick
GotFocus	KeyDown	KeyPress
KeyUp	LostFocus	MouseDown
MouseMove	MouseUp	*Ring

Methods

Drag	Move	Refresh
SetFocus	ZOrder	

See Also

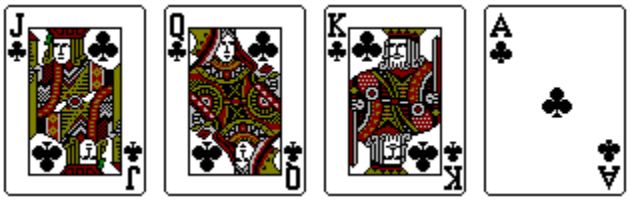
MhClock

MhCard

Properties Events Methods

Description

The MhCard custom control lets you display any card from a standard 52-card deck of playing cards. Typical MhCard controls are shown below:



File Name

MHCD200.VBX

Class Name

MhCardDeck

Remarks

The bitmaps used by the MhCard control are stored in the MHCARDS.DLL file, which must be in your path.

To display a card, simply set the card value (.Value property), the suit (.Suit property), choose the card "back" (.CardBack property), and we do the rest!

Note: The ZOrder method and .HelpContextID property are valid only for VB versions 2.0 and later.

Properties

AutoSize	BackColor	*CardBack
DragIcon	DragMode	Enabled
Height	HelpContextID	hWnd
Index	Left	MousePointer
Name	Parent	*Suit
TabIndex	TabStop	Tag
Top	*Value	Visible
Width		

Events

Change	Click	DblClick
DragDrop	DragOver	GotFocus
KeyDown	KeyPress	KeyUp
LostFocus	MouseDown	MouseMove
MouseUp		

Methods

Drag	Move	Refresh
SetFocus	ZOrder	

MhClock

Properties Events Methods

Description

The MhClock control displays a clock in digital or analog form. It also has an alarm feature. You can use the default clock images or your own bitmaps.

File Name

MHCL200.VBX

Class Name

MhClock

Remarks

Using VB code, you can easily display the time using only numbers. The MhClock control enhances your project by displaying the image of a clock when you want to display the time.

The functionality of the .Hour, .Minute and .Second properties are slightly different in run mode than in design mode. At design time, they return the actual setting (zero or an "offset"). You set the property value to -1 to indicate the property should be set to the current time when the control is loaded. At run time, they return the current time, offset by the clock's setting, but setting all three properties to -1 causes them to return the current time. If one or more of these properties are "set" to a given time (or "offset"), the clock starts counting at that time and increments the properties accordingly. For example, to make the clock run 30 minutes "fast", use the following code:

```
MhClock1.Hour = -1
MhClock1.Minute = -1
MhClock1.Second = -1
Temp% = MhClock1.Minute + 30
If Temp% > 59 Then
  MhClock1.Hour = MhClock1.Hour + 1
  MhClock1.Minute = MhClock1.Minute - 30
Else
  MhClock1.Minute = MhClock1.Minute + 30
End If
```

If all the alarm elements are set to -1, the alarm is off. Setting the alarm hour to zero effectively turns it off as well, because hour zero is never reached. If any Alarm property is set to a value other than -1, the alarm is activated. The Alarm event procedure executes when:

```
(MhClock1.Alarmhour = MhClock1.Hour Or
⇒ MhClock1.AlarmHour = -1) And
⇒ (MhClock1.AlarmMinute = MhClock1.Minute Or
⇒ MhClock1.AlarmMinute = -1) And
⇒ (MhClock1.AlarmSecond = MhClock1.Second Or
⇒ MhClock1.AlarmSecond = -1)
```

To ignore a particular element of the time for alarm purposes, leave that element set to -1. For example, to ring the alarm any time within the minute of 10:22 am, execute the following code:

```
MhClock1.AlarmHour = 10
MhClock1.AlarmMinute = 22
MhClock1.AlarmSecond = -1
```

In the above scenario, the Alarm event procedure will execute at 10:22. You should code any action for the alarm in the Alarm event procedure.

The .FontStyle property determines the 3-D effect of the text displayed in the control, and includes a variety of "lowered" and "raised" styles.

Note: The ZOrder method and .HelpContextID property are valid only for VB versions 2.0 and later.

Properties

*AlarmHour	*AlarmMinute	*AlarmSecond
BackColor	*BestFit	*BorderColor
DragIcon	DragMode	Enabled
*FillColor	FontBold	FontItalic
FontName	FontSize	FontStrikeThru
*FontStyle	FontUnderline	ForeColor
Height	HelpContextID	hWnd
*Hour	*HourBottom	*HourHandLen
*HourHandWidth	*HourHeight	*HourLeft
*HourRight	*HourTop	*HourWidth
Index	*Interval	Left
*LightColor	*Minute	*MinuteBottom
*MinuteHandLen	*MinuteHandWidth	*MinuteHeight
*MinuteLeft	*MinuteRight	*MinuteTop
*MinuteWidth	MousePointer	Name
Parent	*PictureAClock	*PictureDClock
*ScaleMode	*Second	*SecondBottom
*SecondHandLen	*SecondHandWidth	*SecondHeight
*SecondLeft	*SecondRight	*SecondTop
*SecondWidth	*ShadowColor	*ShowMinimized
*Style	TabIndex	TabStop
Tag	*TextColor	Top
Visible	Width	

Events

*Alarm	*ChangeHour	*ChangeMinute
*ChangeSecond	DragDrop	DragOver
GotFocus	LostFocus	

Methods

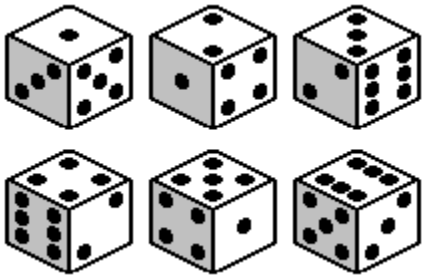
Drag	Move	Refresh
SetFocus	ZOrder	

MhDice

Properties Events Methods

Description

The MhDice control is a specialized picture box control that displays dice. These are the default MhDice bitmaps:



File Name

MHDC200.VBX

Class Name

MhDice

Remarks

The bitmaps are designed to appear as if you're looking down at a die and you see three of its faces – left, right, and top – simultaneously. To display a particular die, just set the .Value property to the value you want to see on the top of the die.

You can use custom bitmap files in the place of the bitmaps shown above. After you've created six bitmap files, add one MhDice control to your application and set the .Picture*x* properties to your custom bitmap files. Note that the MHDICE project contains dice drawn in red and black.

When you use bitmaps in the MhDice control, and you specify those bitmaps in the .Picture*x* properties at design time, the bitmaps become part of your program. This means you don't have to distribute the bitmaps with your programs. On the other hand, if you use the LoadPicture function to load pictures at run time, then the bitmap files must be present at run time.

The MhDice control contains four custom color properties – .FillColor, .LeftSideColor, .RightSideColor, and .TopSideColor – to give you control over the colors used to display the dice.

Note: The ZOrder method and .HelpContextID property are valid only for VB versions 2.0 and later.

See Also

MhCard

Properties

AutoSize	BackColor	DragIcon
DragMode	Enabled	*FillColor
Height	HelpContextID	hWnd
Index	Left	*LeftSideColor
MousePointer	Name	Parent
*Picture1	*Picture2	*Picture3
*Picture4	*Picture5	*Picture6
*RightSideColor	TabIndex	TabStop
Tag	Top	*TopSideColor
*Value	Visible	Width

Events

Click	DragDrop	DragOver
GotFocus	KeyDown	KeyPress
KeyUp	LostFocus	MouseDown
MouseMove	MouseUp	

Methods

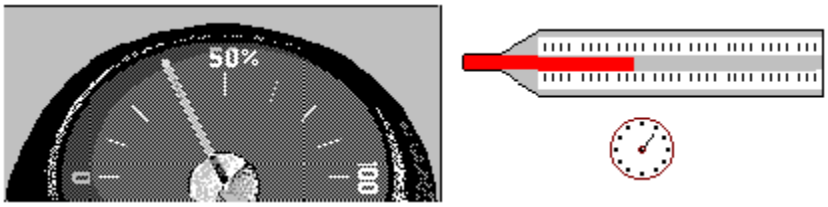
Drag	Move	Refresh
SetFocus	ZOrder	

MhGauge

Properties Events Methods

Description

The MhGauge control displays your choice of linear (filled), needle or bitmap style gauges. MhGauge can also use custom bitmaps of gauges that you create, instead of the bitmaps we supply. Typical MhGauge controls are shown below.



File Name

MHGA200.VBX

Class Name

MhGauge

Remarks

MhGauge controls are useful for percent-complete indicators, fuel gauges, thermometers or any other type of analog gauge.

In the directory containing the MHGAUGE project, you'll find the bitmaps we used to create the gauges shown above. When you use bitmaps (or icons) in the MhGauge control, and you specify them in the .Picture and .PictureFill properties at design time, the bitmaps become a part of your program. This means you don't have to distribute the bitmaps/icons separately with your programs. On the other hand, if you use the LoadPicture function to add bitmaps/icons at run time, then the files must be present at run time.

You can easily control the range in which the needles are displayed by setting a wide difference between .Min and .Max and a narrower range for the actual numbers placed into .Value. For example, to have a semi-circular gauge where the needle moves only in the 90-degree quadrant surrounding "straight up", you would set:

```
.Min = 0
.Max = 100
LowestValueYouPass = 25
HighestValueYouPass = 75
```

Note: The ZOrder method and .HelpContextID property are valid only for VB versions 2.0 and later.

Note: The .DataChanged, .DataField and .DataSource properties are valid only for VB version 3.0.

Properties

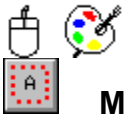
AutoSize	*BackColor	Caption
*ControlBox	DataChanged	DataField
DataSource	DragIcon	DragMode
Enabled	*ForeColor	Height
HelpContextID	hWnd	Icon
Index	*InnerBottom	*InnerLeft
*InnerRight	*InnerTop	Left
*Max	MaxButton	*MDI
*Min	MinButton	MousePointer
Name	*NeedleWidth	Parent
*Picture	*PictureFill	*ReverseFill
*ScaleMode	*Sizeable	*Style
TabIndex	TabStop	Tag
Top	*Value	Visible
Width	WindowState	

Events

*Change	Click	*Close
DblClick	DragDrop	DragOver
GotFocus	KeyDown	KeyPress
KeyUp	LostFocus	MouseDown
MouseMove	MouseUp	*Move
*Resize		

Methods

Drag	Move	Refresh
SetFocus	ZOrder	



MhMarque

Properties Events Methods

Description

The MhMarque control is an enhanced label control that can show a moving banner of text, with or without a set of "moving" bitmaps.

File Name

MHMQ200.VBX

Class Name

MhMarque

Remarks

MhMarque controls are useful when you want to create a control that attracts attention. The control combines some of the animation qualities of the MhAnimate custom control and the Mh3d enhanced label control.

This control contains two sets of "inner" properties (.Caption*xxxx* and .Picture*xxxx*). These properties define where the caption and/or the picture is displayed within the control. Both sets of properties are expressed in twips and are relative to the borders of the control. That is, setting *xxxx*Left = 5 sets that left border 5 twips inside the control's left border, setting *xxxx*Bottom = 5 sets that bottom border 5 twips inside the control's bottom border.

When you use bitmaps (or icons) in MhMarque controls, and you specify those bitmaps in the .Picture1, .Picture2 and .Picture3 properties at design time, the bitmaps become a part of your program. This means you do not have to distribute the bitmaps/icons separately with your programs. On the other hand, if you use the LoadPicture function to add bitmaps/icons at run time, then the files must be present at run time.

The .FontStyle property determines the 3-D effect of the text displayed in the control, and includes a variety of "lowered" and "raised" styles.

Note: The ZOrder method and .HelpContextID property are valid only for VB versions 2.0 and later.

Properties

*AutoSize	BackColor	*BorderColor
Caption	*CaptionBottom	*CaptionLeft
*CaptionRight	*CaptionTop	*Direction
DragIcon	DragMode	Enabled
FontBold	FontItalic	FontName
FontSize	FontStrikethru	*FontStyle
FontTransparent	FontUnderline	ForeColor
Height	HelpContextID	hWnd
Index	*Interval	Left
*LightColor	MousePointer	Name
*OuterFillColor	Parent	*Picture1
*Picture2	*Picture3	*PictureBottom
*PictureLeft	*PictureRight	*PictureTop
*ShadowColor	*Style	TabIndex
TabStop	Tag	*TextColor
*TextFillColor	Top	Visible
Width		

Events

Click	DragDrop	DragOver
GotFocus	KeyDown	KeyPress
KeyUp	LostFocus	MouseDown
MouseMove	MouseUp	

Methods

Drag	Move	Refresh
SetFocus	ZOrder	

MhSlide

Properties Events Methods

Description

MhSlide is a scrollbar-like control used to display a custom scale and button. You move the slide's button to set the .Value of the control. MhSlide provides four different slides you can use in your program, or you can create your own slides. The default bitmap is shown below:

File Name

MHSL200.VBX

Class Name

MhSlide

Remarks

MhSlide controls are useful for representing volume controls, slides on industrial equipment, rulers for positioning text or any situation where you want to display a scale and position a pointer within a range of values.

When you create an MhSlide control, the .Style property defaults to zero (Vertical). As you draw the slide, the default bitmaps are used to completely fill the control. If you change the orientation of a slide, you must adjust the height and width you've reserved on your form to match the new direction.

You can use either the .BackColor or .ForeColor property to define the color that fills the scale on MhSlide controls.

The MHSL200.VBX contains bitmaps for the default slide, so if you use the defaults, you don't need to include any other files with your program. If you specify custom bitmaps during design mode, VB includes those bitmap files in your .EXE file when you compile your program. However, if you want to specify the bitmaps at run time, be sure to make the bitmap files accessible when you run your program. If you don't, you'll receive a *File not found* error when you try to load the files.

While the operation of MhSlide is similar to Visual Basic's scroll bar controls, there are some differences:

- You can specify separate values for large and small movements of the button. VB provides a single property for these movements.

- You can specify custom bitmaps. If you do, be sure to set the .Style property to match the orientation (vertical or horizontal) of your bitmap. If you don't, MhSlide won't be able to display your slide the way you intend. If you're interested in creating your own slides, please see Creating your own slides.

Note: The ZOrder method and .HelpContextID property are valid only for VB versions 2.0 and later.

Note: The .DataChanged, .DataField and .DataSource properties are valid only for VB version 3.0.

Properties

BackColor	DataChanged	DataField
DataSource	DragIcon	DragMode
Enabled	ForeColor	Height
HelpContextID	hWnd	Index
*LargeDown	*LargeUp	Left
*Max	*Min	MousePointer
Name	Parent	*PictureButton
*PictureMask	*PictureSlide	*SmallDown
*SmallUp	*Style	TabIndex
TabStop	Tag	Top
*Value	Visible	Width

Events

Change	DragDrop	DragOver
GotFocus	KeyDown	KeyPress
KeyUp	LostFocus	MouseDown
MouseMove	MouseUp	

Methods

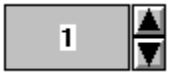
Drag	Move	Refresh
SetFocus	ZOrder	

MhSpin

Properties Events Methods

Description

The MhSpin control is a spin control with optional label and 3-D appearance. The term "spin" means that the control is similar to the arrow portions of a scrollbar, combined with a label, used to increase or decrease values. The default control is shown below:



File Name

MHSN200.VBX

Class Name

MhSpin

Remarks

The MhSpin control is useful when you want to make it easy for a user to change a value quickly, using the mouse.

This control contains two sets of "inner" properties (Picture*xxxx* and Value*xxxx*). These properties define where the picture (arrows or bitmaps) and (optionally) the value (which you can think of as a "Caption") are displayed within the control. Both sets of properties are expressed in twips and are relative to the borders of the control. That is, setting *xxxx*Left = 5 sets that left border 5 twips inside the control's left border, setting *xxxx*Bottom = 5 sets that bottom border 5 twips inside the control's bottom border.

When you use bitmaps in the MhSpin control, and you specify those bitmaps in the .Picture1Up, .Picture1Down, .Picture2Up and .Picture2Down properties at design time, the bitmaps become a part of your program. That means you don't have to distribute the bitmaps with your programs. On the other hand, if you use the LoadPicture function to add bitmaps at run time, then the files must be present at run time.

The .FontStyle property determines the 3-D effect of the text displayed in the control, and includes a variety of "lowered" and "raised" styles.

Note: The ZOrder method and .HelpContextID property are valid only for VB versions 2.0 and later.

Note: The .DataChanged, .DataField and .DataSource properties are valid only for VB version 3.0.

Properties

Alignment	*Arrows	*AutoSize
BackColor	*BorderColor	*BorderStyle
DataChanged	DataField	DataSource
DragIcon	DragMode	Enabled
*FillColor	FontBold	FontItalic
FontName	FontSize	FontStrikeThru
*FontStyle	FontTransparent	FontUnderline
Height	HelpContextID	hWnd
Index	*Interval	Left
*LightColor	Max	Min
MousePointer	Multiline	Name
Parent	*Picture1Down	*Picture1Up
*Picture2Down	*Picture2Up	*PictureBottom
*PictureLeft	*PictureRight	*PictureTop
*ShadowColor	*SmallDown	*SmallUp
*Style	TabIndex	TabStop
Tag	*TextColor	Top
Value	*ValueBorderStyle	*ValueBottom
*ValueDisplay	*ValueLeft	*ValueRight
*ValueTop	Visible	Width

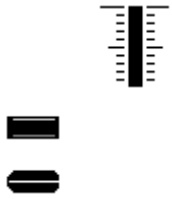
Events

*Change1	DragDrop	DragOver
GotFocus	KeyDown	KeyPress
KeyUp	LostFocus	

Methods

Drag	Move	Refresh
SetFocus	ZOrder	

Creating your own slides
The MhSlide control contains all the instructions to move the button and to set and return the value of the button. Consequently, if you want to create a custom slide, all you have to do is design three bitmaps like the ones shown below one for the slide, one for the button and one for the "mask."



Slide Button Mask

When you set the .Picture*x* properties of the control to your bitmap filenames, MhSlide will do the rest.

To create bitmap files, you can use PaintBrush or a program like the resource compiler Microsoft ships with its Windows SDK. Let's look at guidelines for creating bitmaps we'll use the bitmaps for the default vertical slide.

The Slide bitmap
When you design a slide bitmap, you need only create a bitmap that contains the unique portion of your slide's scale. MhSlide automatically fills the height (for .Style zero) or the width (for .Style one) of your slide with this bitmap. We suggest you use scales of some type.

The Button bitmap
MhSlide uses the button bitmap in combination with the mask bitmap to draw the indicator on the slide. The button bitmap should represent the button, or pointing object, of the slide. As you can see from the figure above, the button bitmap appears slightly different from the button of the default MhSlide control. This difference is due to the action of the mask bitmap.

The Mask bitmap
To draw the indicator on the slide, MhSlide combines the bitmaps of the slide, button and mask in such a way that the scale shows through certain portions of the button. The mask bitmap determines which portions of the button bitmap will be visible and which portions will allow the scale to show through.

You should draw the mask with black and white only. Use black where you want to see the pointing device and white where you want the slide to show through. Be sure that these two bitmaps are the same size!

When you design these three bitmaps, picture them displayed one on top of the other. Each white pixel on the mask bitmap lets the slide bitmap show, while the black pixels let the button bitmap show through.

Custom Property & Event Reference

This section contains a description of every custom property and event used in all of the **VBTools** controls.

All of the properties and events are listed alphabetically and are described using the format shown below. For general information about the custom controls, please refer to the details for each control (via the Contents screen).

The following describes the headings used in each entry:

Description

A brief description of the property or event.

Applies To

A list of all custom controls that have this property or event.

Remarks

Additional information about the property or event, such as the default setting, availability at design time and run time, and special considerations for its use.

Caution: (if present) Warns you about special considerations when using the property or event.

Important: (if present) Just what it says! Please take the time to read these special notes.

Alarm Event

Description

Occurs when the alarm time is reached. You should code any action for the alarm in this procedure. See the MhClock remarks for more information.

Applies To

MhClock

AlarmHour, AlarmMinute, AlarmSecond Properties

Description

Set or return the alarm hour, minute or second setting.

Apply To

MhClock

Data Type

Integer

Remarks

.AlarmHour – The default value is -1. The range is -1 to 24. Please note that the alarm hour setting should be made using a "24 hour" base, even if the clock is a 12 hour clock.

.AlarmMinute, .AlarmSecond – The default value is -1. The range is -1 to 59.

Arrows Property

Description

Sets or returns whether or not arrows are drawn automatically by the control.

Applies To

MhSpin

Data Type

Integer (Boolean)

Remarks

The .Arrows settings are:

Setting	Description
True (-1)	(Default) Use the GDI to draw standard arrows.
False (0)	Don't draw arrows, i.e., use programmer-defined bitmaps.

If you specify that the control should display bitmaps instead of arrows (using one or more of the Picture*nxxxx* properties), this property is ignored.

AutoSize Property

Description

- MhSpin – Sets or returns whether the control should automatically size the picture areas according to the size of the .Picture1Up bitmap.
- MhMarque – Sets or returns whether the control should automatically size the picture area when bitmaps are assigned to the .Picture*x* properties.

Applies To

MhMarque, MhSpin

Data Type

Integer (Boolean)

Remarks

MhSpin – The .AutoSize settings are:

Setting	Description
True (-1)	Size the picture areas to the size of the .Picture1Up bitmap.
False (0)	(Default) Stretch or shrink the bitmap to the size of the picture area.

MhMarque – The .AutoSize settings are:

Setting	Description
True (-1)	(Default) Size the picture area to the size of the .Picture*x* bitmaps.
False (0)	Stretch or shrink the bitmaps to the size of the picture area.

BackColor Property

Description

- MhGauge – Sets or returns the color used to fill an "empty" linear gauge.

Applies To

MhGauge

Data Type

Long Integer

Remarks

MhGauge – Think of this as the background color of the fill area. The default value is &H80000005& (White). The .BackColor property doesn't effect the color of needles when the .Style property is two or three.

BestFit Property

Description

Sets or returns whether the control automatically sizes the hour, minute and second areas of the clock when the .Style property or size of the control is changed.

Applies To

MhClock

Data Type

Integer (Boolean)

Remarks

The .BestFit settings are:

Setting	Description
True (-1)	(Default) The control automatically sizes itself upon change in .Style or size.
False (0)	The control does not automatically size itself.

BorderColor Property

Description

- Sets or returns the color of the outer border.

Applies To

MhClock, MhMarqueMhSpin

Data Type

Long Integer

Remarks

MhMarque, MhSpin – The default value is &H00000000& (Black).

MhClock – The default value is the same as the color defined in Control Panel for the Window Frame.

BorderStyle Property

Description

Sets or returns the control's border style.

Applies To

MhSpin

Data Type

Integer

Remarks

MhSpin – In addition to the "No Border" and "Single Line" options, we provide a "Single Line with Rounded Corners" option. The .BorderStyle settings are:

Setting	Description
0	No border
1	Single Line
2	Single Line with Rounded Corners

The default is 1 (Single Line).

CaptionTop, CaptionLeft, CaptionBottom, CaptionRight Properties

Description

Set or return the portion of the control where the caption is displayed, relative to the borders of the control.

Apply To

MhMarque

Data Type

Integer

Remarks

These values must be positive and expressed in twips. The values for the .CaptionTop and .CaptionLeft properties are relative to the borders of the control. In other words, the top left corner of the caption area is .CaptionTop units below the top of the control and .CaptionLeft units to the right of the left border. Similarly, the values for the .CaptionBottom and .CaptionRight properties are relative to the bottom right borders of the control.

CardBack Property

Description

Sets or returns the card-back image to display when the .Value property is zero.

Applies To

MhCard

Data Type

Integer

Remarks

The .CardBack settings are:

Setting	Description	Setting	Description
0	(Default) Red Checks	7	Leaves 2
1	Blue Checks	8	Fish
2	Red Hatch	9	Conch
3	Blue Hatch	10	Castle
4	Robot	11	Beach
5	Roses	12	Hand
6	Leaves 1		

Change Event

Description

- MhGauge – Occurs when the .Value property changes. The only way to change the .Value property is from code.

Applies To

MhGauge

Change1 Event

Description

Occurs when the .Value property changes, including when you set the Value property from code.

Applies To

MhSpin

Remarks

MhSpin – *PartBar* indicates what action was taken by the user, and contains one of the following values:

Value	Description
1	The up/right arrow was clicked.
2	The down/left arrow was clicked.
3	Your program changed the Value property

ChangeHour, ChangeMinute, ChangeSecond Events

Description

Occurs when the .Hour, .Minute or .Second property changes.

Apply To

MhClock

Remarks

ChangeHour – You can put code for special effects, such as ringing chimes, in this procedure to ring at the change of the hour.

ChangeMinute – You can place code in this procedure for any action you want to occur once per minute.

ChangeSecond – If you want a clock that ticks, you can put code here to invoke the sound effect.

Close Event

Description

Occurs when the Close option is selected from the control box menu.

Applies To

MhGauge

Remarks

MhGauge – This event occurs only when .MDI is True and .ControlBox is True.

ControlBox Property

Description

Sets or returns whether a control box is available on the control.

Applies To

MhGauge

Data Type

Integer (Boolean)

Remarks

The .ControlBox settings are:

Setting	Description
True (-1)	The control box is displayed.
False (0)	(Default) The control box is not displayed.

MhGauge – You must set both the .MDI and .ControlBox properties to True for the control to display a control box.

Direction Property

Description

Sets or returns the direction of text movement in the control.

Applies To

MhMarque

Data Type

Integer

Remarks

The .Direction settings are:

Setting	Description
0	(Default) The text moves forward.
1	The text moves backward.

FillColor Property

Description

- MhClock – Sets or returns the background color of the control.
- MhDice – Sets or returns the color displayed in the body of the MhDice control. The body is the portion of the bitmap that doesn't contain the image of the die.
- MhSpin – Sets or returns the background color of the caption (referred to as the "Value").

Applies To

MhClock, MhDice, MhSpin

Data Type

Long Integer

Remarks

MhDice, – The default value is &H00FFFFFF& (White).

MhClock – The default value is the same as the color defined in Control Panel for the Button Face. While the .FillColor property can be set to any valid color value, the best effect is normally achieved by setting it to light gray (&H00C0C0C0&).

MhSpin – The default value is &H00C0C0C0& (Light Gray).

FontStyle Property

Description

Sets or returns the 3-D effect of the text or caption that is printed on the control.

Applies To

MhClock, MhMarque, MhSpin,

Data Type

Integer

Remarks

The .FontStyle settings are:

Setting	Description
0	(Default) None.
1	Raised (the text appears raised off the control).
2	Raised with more shading.
3	Lowered (the text appears inset into the control).
4	Lowered with more shading.

The 3-D effect is accomplished by outlining the text with the colors specified in the .LightColor and .ShadowColor properties. When .FontStyle is set to one or two (raised), the outline around the top and left sides of the text are drawn with the .LightColor value and the right and lower portions with the .ShadowColor value. When .FontStyle is set to three or four (lowered), the outline colors are reversed.

MhClock – Note that the .FontStyle setting also affects the needle and "tick" marks in the analog clock.

ForeColor Property

Description

- MhGauge – Sets the color used to paint the "percent complete" portion of the gauge.

Applies To

MhGauge

Data Type

Long Integer

Remarks

MhGauge – Think of this as the foreground color of the fill area. The .ForeColor property does not affect the color of needles when .Style is two or three.

Hour, Minute, Second Properties

Description

Set or return the hour, minute or second setting of the clock.

Apply To

MhClock

Data Type

Integer

Remarks

.Hour – The default value is -1. The range is -1 to 24, excluding zero. The .Hour property is in 24 hour format, regardless of the style setting of the clock. Please see the MhClock remarks for more details.

.Minute, .Second – The default value is -1. The range is -1 to 59.

HourBottom, MinuteBottom, SecondBottom Properties

Description

Set or return the bottom position of the hour, minute or second display, in the control's .ScaleMode, relative to the bottom border of the control. These properties are valid for both digital and analog clocks.

Apply To

MhClock

Data Type

Integer

Remarks

The default value is zero. The range is zero to 32,767.

HourHandLen, MinuteHandLen, SecondHandLen Properties

Description

Set or return the length of the hour, minute or second hand on an analog clock.

Apply To

MhClock

Data Type

Integer

Remarks

.HourHandLen – The default value is 50. The range is zero to 100. Note that this property is expressed as a percentage of the total area available to the hour hand (defined by .HourWidth and .HourHeight).

.MinuteHandLen, .SecondHandLen – The default value is 90. The range is zero to 100. Note that this property is expressed as a percentage of the total area available to the minute/second hand (defined by .MinuteWidth/.SecondWidth and .MinuteHeight/.SecondHeight).

HourHandWidth, MinuteHandWidth, SecondHandWidth Properties

Description

Set or return the width, in the control's .ScaleMode, of the hour, minute or second hand.

Apply To

MhClock

Data Type

Integer

Remarks

.HourHandWidth – The default value is four. The range is zero to 32,767.

.MinuteHandWidth, .SecondHandWidth – The default value is 30. The range is zero to 32,767.

HourHeight, MinuteHeight, SecondHeight Properties

Description

Set or return the height, in the control's .ScaleMode, available to the hour, minute or second display. These properties are valid for both digital and analog clocks.

Apply To

MhClock

Data Type

Integer

Remarks

The default value is the control's default height. The range is zero to 32,767.

HourLeft, MinuteLeft, SecondLeft Properties

Description

Set or return the left position of the hour, minute or second display, in the control's .ScaleMode, relative to the left border of the control. These properties are valid for both digital and analog clocks.

Apply To

MhClock

Data Type

Integer

Remarks

.HourLeft – The default value is zero. The range is zero to 32,767.

.MinuteLeft – The default value is 1/3 of the default width of the control. The range is zero to 32,767.

.SecondLeft – The default value is 2/3 of the default width of the control. The range is zero to 32,767.

HourRight, MinuteRight, SecondRight Properties

Description

Set or return the right position of the hour, minute or second display, in the control's .ScaleMode, relative to the right border of the control. These properties are valid for both digital and analog clocks.

Apply To

MhClock

Data Type

Integer

Remarks

.HourRight – The default value is 2/3 of the default width of the control. The range is zero to 32,767.

.MinuteRight – The default value is 1/3 of the default width of the control. The range is zero to 32,767.

.SecondRight – The default value is zero. The range is zero to 32,767.

HourTop, MinuteTop, SecondTop Properties

Description

Set or return the top position of the hour, minute or second display, in the control's .ScaleMode, relative to the top border of the control. These properties are valid for both digital and analog clocks.

Apply To

MhClock

Data Type

Integer

Remarks

The default value is zero. The range is zero to 32,767.

HourWidth, MinuteWidth, SecondWidth Properties

Description

Set or return the width, in the control's .ScaleMode, available to the hour, minute or second display. These properties are valid for both digital and analog clocks.

Apply To

MhClock

Data Type

Integer

Remarks

The default value is 1/3 the default width of the control. The range is zero to 32,767.

InnerBottom, InnerLeft, InnerRight, InnerTop Properties

Description

- MhGauge – Set the inner dimensions of a gauge control, relative to the borders of the control.

Apply To

MhGauge

Data Type

Integer

Remarks

These values must be positive and expressed using values in the control's ScaleMode. The values for the .InnerTop and .InnerLeft properties are relative to the top left borders of the control. In other words, the top left corner of the inner area is .InnerTop units below the top border and .InnerLeft units to the right of the left border. Similarly, the values for the .InnerBottom and .InnerRight properties are relative to the bottom right edge of the control.

MhGauge – These properties specify the coordinates of the area of the control in which the "percent complete" value is displayed (see the .Style property). Needle gauges display a needle within these coordinates. See the remarks for the MhGauge control.

Interval Property

Description

- MhAlarm – Sets or returns the interval, in milliseconds, between changes in the bitmap displays when the .RingOn property is non-zero.
- MhClock – Sets or returns the interval, in milliseconds, at which the clock is updated.
- MhMarque – Sets or returns the interval, in milliseconds, between the display of the pictures and/or the moving text.
- MhSpin – Sets or returns the interval, in milliseconds, between each Change1 event when the mouse or keyboard is held down.

Applies To

MhAlarm, MhClock, MhMarque, MhSpin

Data Type

MhAlarm, MhClock, MhMarque, MhSpin – Long Integer

Remarks

MhAlarm – The default value is determined by the .Style property. See the .Style property for possible defaults. The range is 1 to 2,147,483,647.

MhClock, MhMarque – The default value is 1,000. The range is zero to 2,147,483,647.

MhSpin – The default value is 10. The range is zero to 2,147,483,647. Use this property to control the speed at which the .Value property changes.

LargeDown, LargeUp Properties

Description

- MhSlide – Set or return the amount to increase the .Value property when the PgDn/PgUp key is pressed.

Apply To

MhSlide

Data Type

Integer

Remarks

MhSlide – The default value is 10. The range is zero to 32,767.

LeftSideColor, RightSideColor, TopSideColor Properties

Description

Set or return the color displayed on the left, right or top side of a die.

Apply To

MhDice

Data Type

Long Integer

Remarks

.LeftSideColor – The default value is &H00C0C0C0& (Light Gray).

.RightSideColor, .TopSideColor – The default value is &H00FFFFFF& (White).

LightColor Property

Description

Sets or returns the "light" color, as opposed to the "shadow" color of the control.

Applies To

MhClock, MhMarque, MhSpin

Data Type

Long Integer

Remarks

There are three colors used when creating the 3-D effect for the control border and text: .LightColor, .ShadowColor and .TextColor. While any colors can be used, the best effect is normally achieved by setting .LightColor to &H00FFFFFF& (White), .ShadowColor to &H00808080& (Medium Gray) and .TextColor to &H00000000& (Black).

MhMarque, MhSpin – The default value is &H00FFFFFF& (White).

MhClock – The default value is the same as the color defined in Control Panel for the Window Background.

Max Property

Description

- MhGauge, MhSlide – Sets or returns the maximum amount for the .Value property.

Applies To

MhGauge, MhSlide

Data Type

Integer

Remarks

MhGauge – The default value is 100. The range is zero to 32,767.

MhSlide – Vertical slides are at the .Max when the button is at the bottom of the slide. Horizontal slides are at this point when the button is at the right end of the slide. The default value is 100. The range is zero to 32,767.

MDI Property

Description

Sets or returns whether the control displays a caption, has MDI properties and can be resized by the user at run time.

Applies To

MhGauge

Data Type

Integer (Boolean)

Remarks

If the .MDI property is False, the control does not display the Control Box, the Minimize Button or the Maximize Button, even if those properties are True. The .MDI settings are:

Setting	Description
True (-1)	The control has MDI characteristics.
False (0)	(Default) The control does not have MDI characteristics.

Min Property

Description

- MhGauge, MhSlide – Sets or returns the minimum for the .Value property.

Applies To

MhGauge, MhSlide

Data Type

Integer

Remarks

MhGauge – The default value is zero. The range is zero to 32,767.

MhSlide – Vertical slides are at the .Min point when the button is at the top of the slide. Horizontal slides are at this point when the button is at the left end of the slide.

NeedleWidth Property

Description

Sets the width of the needle, in the control's .ScaleMode, when the .Style property is two or three.

Applies To

MhGauge

Data Type

Integer

Remarks

The default value is one pixel or 15 twips.

OuterFillColor Property

Description

Sets or returns the color that is displayed between the caption area and the outer border.

Applies To

MhMarque

Data Type

Long Integer

Remarks

The default value is &H00C0C0C0& (Light Gray).

PauseTime Property

Description

Sets or returns the length, in milliseconds, of the silence between ringing sounds.

Applies To

MhAlarm

Data Type

Integer

Remarks

The range is zero to 32,767.

Picture Property

Description

- MhGauge – Sets or returns the image to display on the gauge.

Applies To

MhGauge

Data Type

Picture

Remarks

MhGauge – Usage is the same as the standard .Picture property. You can use the LoadPicture function or set the .Picture property equal to the picture property of another control. Unless indicated otherwise below, you may select .BMP or .ICO files for this property. By default, there is no picture loaded. Write-only at design time. At run time, this property returns the picture's "handle."

MhGauge – See the .PictureFill property for information on how this property can be used to acheive a "percent complete" effect.

Picture1, Picture2, Picture3, Picture4, Picture5, Picture6 Properties

Description

- MhAlarm (.Picture1, .Picture2 and .Picture3 only) – Set or return the three pictures, used in conjunction with the .PictureMask*x* properties, to display in the control.
- MhDice – Set or return the six pictures used to display the programmer-supplied images of dice.
- MhMarque (.Picture1, .Picture2 and .Picture3 only) – Set or return the three pictures that can be loaded and used for an "animated" control.

Apply To

MhAlarm, MhDice, MhMarque

Data Type

Picture

Remarks

Usage is the same as the standard .Picture property. You can use the LoadPicture function or set the .Picture*x* property equal to the picture property of another control. By default, there are no pictures loaded. Write-only at design time. At run time, this property returns the specified picture's "handle."

MhAlarm – These properties are used only when the .Style property is set to 3 (User-supplied bitmaps). See the .PictureMask*x* properties for more information on how these properties are used together to display irregularly-shaped images.

MhDice – When no pictures are loaded, the control uses the default images.

Picture1Down, Picture2Down Properties

Description

- MhSpin – Set or return the picture to display in the up/right or down/left arrow position when the arrow is depressed.

Apply To

MhSpin

Data Type

Picture

Remarks

Usage is the same as the standard .Picture property. You can use the LoadPicture function or set the .Picture*x*Down property equal to the picture property of another control. By default, there are no pictures loaded. Write-only at design time. At run time, this property returns the specified picture's "handle."

Picture1Up, Picture2Up Properties

Description

- MhSpin – Set or return the picture to display in the up/right or down/left arrow position when the arrow is not depressed.

Apply To

MhSpin

Data Type

Picture

Remarks

Usage is the same as the standard .Picture property. You can use the LoadPicture function or set the .Picture*x*Up property equal to the picture property of another control. By default, there are no pictures loaded. Write-only at design time. At run time, this property returns the specified picture's "handle."

PictureAClock Property

Description

Sets or returns the image (bitmap or icon) displayed within the control when the style of the clock is set to one of the analog types.

Applies To

MhClock

Data Type

Picture

Remarks

Usage is the same as the standard .Picture property. You can use the LoadPicture function or set the .PictureAClock property equal to the picture property of another control. By default, there is no picture loaded. Write-only at design time. At run time, this property returns the picture's "handle."

Note that when no picture is loaded, the control displays a set of "tick" marks for the analog clock.

PictureButton Property

Description

Sets or returns the bitmap, used in combination with the bitmap assigned to the .PictureMask property, to draw the *button* or *indicator* on the slide.

Applies To

MhSlide

Data Type

Picture

Remarks

Usage is the same as the standard .Picture property. You can use the LoadPicture function or set the .PictureButton property equal to the picture property of another control. By default, there is no picture loaded. Write-only at design time. At run time, this property returns the picture's "handle."

For more information on creating the images that make up the slide, please refer to the .PictureMask property.

PictureDClock Property

Description

Sets or returns the image (bitmap or icon) displayed within the control when the style of the clock is set to one of the digital types.

Applies To

MhClock

Data Type

Picture

Remarks

Usage is the same as the standard .Picture property. You can use the LoadPicture function or set the .PictureDClock property equal to the picture property of another control. By default, there is no picture loaded. Write-only at design time. At run time, this property returns the picture's "handle."

Note that when no picture is loaded, the control displays a border around the digital clock.

PictureFill Property

Description

Sets or returns the fill picture used when the .Style property is five or six.

Applies To

MhGauge

Data Type

Picture

Remarks

The .PictureFill property lets you use bitmaps to display the "percent complete" value instead of a linear or needle indicator. You must set this property for the bitmap fill operation to work.

The bitmap (or icon) you load with the .PictureFill property should be an exact copy of the .Picture bitmap (or icon) except for the fill area. The .PictureFill bitmap should show a completely full gauge and the .Picture bitmap should show a completely empty gauge. When you specify an amount for the .Value property, the MhGauge control determines how much of the bitmap to display in the .Picture bitmap.

Even though most gauges will not use the full length (or height) of the picture you supply, it is still important to set the Inner*xxxx* properties so they accurately reflect the area that "appears" to be filled by the gauge. When you set these properties, the control scales the .Min, .Max and .Value properties to that area and the gauge appears to reflect the value setting properly. If the Inner*xxxx* properties are not set properly, the .Value property might appear to be "out of sync" with the gauge.

The easiest way to size the Inner*xxxx* properties is to first set the gauge style to one of the "normal" linear styles. Next, set the Inner*xxxx* properties (using the right mouse button is easiest), then switch the gauge back to the bitmap fill style.

Usage is the same as the standard .Picture property. You can use the LoadPicture function or set the .PictureFill property equal to the picture property of another control. By default, there is no picture loaded. Write-only at design time. At run time, this property returns the picture's "handle."

PictureMask Property

Description

- MhSlide – Sets or returns the bitmap used (in combination with the .PictureButton property) to draw the *button* or "*indicator"* on the slide.

Applies To

MhSlide

Data Type

Picture

Remarks

Usage is the same as the standard .Picture property. You can use the LoadPicture function or set the .PictureMask property equal to the picture property of another control. By default, there is no picture loaded. Write-only at design time. At run time, this property returns the picture's "handle."

MhSlide – MhSlide combines the .PictureButton and the .PictureMask bitmaps so that the background (the "scale") shows through certain portions of the button.

PictureMask1, PictureMask2, PictureMask3 Properties

Description

Sets or returns the "mask pictures" to use in conjunction with the .Picture*x* properties.. These pictures "mask" the main pictures, allowing the background to show through in certain areas of the pictures.

Apply To

MhAlarm

Data Type

Picture

Remarks

These properties are used only when the .Style property is set to 3 (User-supplied bitmaps). Usage is the same as the standard .Picture property. You can use the LoadPicture function or set a .PictureMask*x* property equal to the picture property of another control. By default, there are no pictures loaded. Write-only at design time. At run time, this property returns the specified picture's "handle."

When you design a bitmap for use with these properties, draw the image using black and white only. Use white to specify the areas of the .PictureMask*x* that should let the background show through. Fill the areas of .PictureMask*x* that should let the .Picture*x* image show with black. You may select .BMP or .ICO files for this property. For information on creating the bitmaps for MhAlarm controls, please refer to *MhSlide* in Section 5(K), which shows you how to create buttons for the MhSlide control. The same technique applies to MhAlarm bitmaps.

PictureSlide Property

Description

Sets or returns the bitmap used to draw the slide. If you don't specify a bitmap, the control uses the default bitmap.

Applies To

MhSlide

Data Type

Picture

Remarks

Usage is the same as the standard .Picture property. You can use the LoadPicture function or set the .PictureSlide property equal to the picture property of another control. By default, there is no picture loaded. Write-only at design time. At run time, this property returns the picture's "handle."

MhSlide places copies of the bitmap end to end until it fills the height (for .Style zero) or the width (for .Style one) of the control. The best slides display some type of scale.

PictureTop, PictureLeft, PictureBottom, PictureRight Properties

Description

Set or return the position where the pictures are displayed, relative to the borders of the control.

Apply To

MhMarque, MhSpin

Data Type

Integer

Remarks

These values must be positive and expressed in twips. The values are relative to the borders of the control. In other words, the top left corner of the picture area is .PictureTop twips below the .top of the control and .PictureLeft twips to the right of the left border. Similarly, the values for the .PictureBottom and .PictureRight properties are relative to the bottom right edge of the control.

MhSpin – When .Arrows is True, these properties affect the built-in arrow bitmaps. When .Arrow is False, these properties affect the programmer-defined bitmaps.

Resize Event

Description

- MhGauge – Occurs when the control is resized (with the mouse or via the control box menu). This event can occur only when .MDI is True and .Sizeable is True.

Applies To

MhGauge

ReverseFill Property

Description

- MhGauge – Sets or returns the direction in which the fill takes place.

Applies To

MhGauge

Data Type

Integer (Boolean)

Remarks

MhGauge – The .ReverseFill settings are:

Setting	Description
True (-1)	Linear horizontal gauges fill from right to left and vertical linear gauges fill from top to bottom.
False (0)	(Default) Linear horizontal gauges fill from left to right and linear vertical gauges fill from top to bottom.

Ring Event

Description

Occurs approximately every .RingTime milliseconds. You can use it to update your program's status, or to execute any other task you want to perform while waiting for the user to respond to the alarm.

Applies To

MhAlarm

RingLength Property

Description

Sets or returns the length of the note sent to the speaker.

Applies To

MhAlarm

Data Type

Integer

Remarks

The range is 1 to 64.

RingMode Property

Description

Sets or returns the type of ring sound.

Applies To

MhAlarm

Data Type

Integer

Remarks

The .RingMode settings are:

Setting	Description
0	(Default) Normal
1	Legato
2	Staccato

Note: The Windows 3.0 sound driver has a bug that causes the .RingMode property to have no effect.

RingOn Property

Description

Sets or returns whether the control is active and, if so, how frequently the alarm rings.

Applies To

MhAlarm

Data Type

Integer

Remarks

The .RingOn settings are:

Setting	Description
0	(Default) Ring off - do not sound tones or toggle bitmaps.
1	Flash ring - single ring only, using .RingTime property.
2	Ring on - continuous ring.
3	No Flash Ring - do not ring on mouse click.

RingTime Property

Description

Sets or returns the length of time, in milliseconds, during which tones are sent to the speaker.

Applies To

MhAlarm

Data Type

Integer

Remarks

The range is zero to 32,767. (Also see .RingLength.)

RingTone Property

Description

Sets or returns the note number that is sent to the Windows API sound driver.

Applies To

MhAlarm

Data Type

Integer

Remarks

The range is 1 to 84.

ScaleMode Property

Description

Sets or returns the scale of measurement for the control.

Applies To

MhClock, MhGauge

Data Type

Integer

Remarks

The .ScaleMode settings are:

Setting	Description
1	Twips
3	(Default) Pixels

ShadowColor Property

Description

Sets or returns the color used for "shadows."

Applies To

MhClock, MhMarque, MhSpin

Data Type

Long Integer

Remarks

There are three colors used when creating the 3-D effect for the control border and text: .LightColor, .ShadowColor and .TextColor. While any colors can be used, the best effect is normally achieved by setting .LightColor to &H00FFFFFF& (White), .ShadowColor to &H00808080& (Medium Gray) and .TextColor to &H00000000& (Black).

MhMarque, MhSpin – The default value is &H00808080& (Medium Gray).

MhClock – The default value is the same as the color defined in Control Panel for the Button Shadow.

ShowMinimized Property

Description

Sets or returns the action taken by the control when its form is minimized.

Applies To

MhClock

Data Type

Integer (Boolean)

Remarks

The .ShowMinimized settings are:

Setting	Description
True (-1)	The clock is displayed in the icon area of the form while the form is minimized. You must size the clock so that it appears properly in the icon area.
False (0)	(Default) The form's icon appears while the form is minimized.

Sizeable Property

Description

If the .MDI property is True, sets or returns whether the control can be sized by the user at run time.

Applies To

MhGauge

Data Type

Integer (Boolean)

Remarks

The .MDI property must be set to True for this property to have an affect. The .Sizeable settings are:

Setting	Description
True (-1)	The control can be resized at run time.
False (0)	(Default) The size of the control is determined at design time.

SmallDown, SmallUp Properties

Description

- MhSlide – Set or return the amount to increase/decrease the .Value property when the right/up or left/down arrow keys are pressed.
- MhSpin – Set or return the amount subtracted or added to .Value when the down/up arrow (vertical spin control) or the left/right arrow (horizontal spin control) is clicked.

Apply To

MhSlide, MhSpin

Data Type

Integer

Remarks

The default value is one. The valid range is zero to 32,767.

Style Property

Description

- MhAlarm – Sets or returns which icon(s) are displayed by the control.
- MhClock – Sets or returns the clock style.
- MhGauge – Sets or returns the control's style.
- MhMarque – Sets or returns the way the .Caption and pictures are displayed in the control.
- MhSlide – Sets or returns the slide orientation.
- MhSpin – Sets or returns the orientation and appearance of the control.

Applies To

MhAlarm, MhClock, MhGauge, MhMarque, MhSlide, MhSpin

Data Type

Integer

Remarks

MhAlarm – The .Style settings are:

Setting	Description
0	(Default) Telephone.
1	Alarm clock.
2	Wrist alarm.
3	User-supplied bitmaps.

Whenever you change the .Style property to a value other than 3, certain properties are automatically adjusted for optimum results (no properties are adjusted when .Style is set to 3). The following table shows the default values of the properties affected by the .Style property:

Property	Style		
	Phone	Alarm clock	Wrist Watch
.PauseTime	1000	500	1000
.RingMode	2-Staccato	2-Staccato	0-Normal
.RingLength	34	32	34
.RingTime	1000	2000	1000
.RingTone	41	45	72
.Interval	29	62	29

MhClock – The valid .Style settings are:

Setting	Description
0	(Default) Digital 12 hour.
1	Digital 12 hour with seconds.
2	Digital 24 hour.
3	Digital 24 hour with seconds.
4	Analog 12 hour.
5	Analog 12 hour with seconds.
6	Analog 24 hour.
7	Analog 24 hour with seconds.

MhGauge – The valid .Style settings are:

Setting	Description
0	(Default) Horizontal Bar, a horizontal linear gauge with "fill."
1	Vertical Bar, a vertical linear gauge with "fill."
2	'Semi' Needle, a semi-circular gauge with "needle." The needle base is centered at the

	bottom of the "inner" rectangle. When .Value = .Min, the needle points 90 degrees to left. When .Value = .Max, the needle points 90 degrees to the right. When .Value = (.Min + .Max) / 2, the needle points straight up.
3	'Full' Needle, a circular gauge with a "needle." The base of the needle is placed at the center of the inner rectangle. When .Value = .Min or .Value = .Max, the needle points 90 degrees to the left. Setting the .Value property in between .Min and .Max points the needle to a proportionate point on the circle, moving clockwise.
4	Horizontal fill with bitmap. This style works the same as style zero, but uses a programmer-supplied bitmap to fill the control. See the .PictureFill property for details.
5	Vertical fill with bitmap. This style works the same as style one, but uses a programmer-supplied bitmap to fill the control. See the .PictureFill property for details.

MhMarque – The valid .Style settings are:

Setting	Description
0	(Default) Moving .Caption, Cycle through .Picture1, .Picture2 and .Picture3.
1	Moving .Caption, Static .Picture1 or 3-D effect if there is no picture in .Picture1.
2	Static .Caption, Cycle through .Picture1, .Picture2 and .Picture3 .
3	Static .Caption, Static .Picture1 or 3-D effect if there is no picture in .Picture1.

MhSpin – The valid .Style settings are:

Setting	Description
0	(Default) 3d Vertical
1	3d Horizontal
2	Normal Vertical
3	Normal Horizontal

Suit Property

Description

Determines the suit to use when the .Value property is non-zero.

Applies To

MhCard

Data Type

Integer

Remarks

The .Suit settings are:

Setting	Description
0	(Default) Clubs
1	Diamonds
2	Hearts
3	Spades

TextColor Property

Description

- MhMarque, MhSpin – Sets or returns the foreground color of the caption.
- MhClock – Sets or returns the color used to draw all text, needles and "tick" marks displayed in the control.

Applies To

MhClock, MhMarque, MhSpin

Data Type

Long Integer

Remarks

There are three colors used when creating the 3-D effect for the control border and text: .LightColor, .ShadowColor and .TextColor. While any colors can be used, the best effect is normally achieved by setting .LightColor to &H00FFFFFF& (White), .ShadowColor to &H00808080& (Medium Gray) and .TextColor to &H00000000& (Black).

MhMarque, MhSpin – The default value is &H00000000& (Black).

MhClock – The default value is the same as the color defined in Control Panel for the Window Text.

TextFillColor Property

Description

Sets or returns the background color of the caption.

Applies To

MhMarque

Data Type

Long Integer

Remarks

The default value is &H00C0C0C0& (Light Gray).

Value Property

Description

- MhCard – Sets or returns the card image to display.
- MhDice – Sets or returns the number on the top of the die.
- MhGauge – Sets or returns the indicator position on the slide.
- MhSlide – Sets or returns the button position on the slide.

Applies To

MhCard, MhDice, MhGauge, MhSlide

Data Type

Integer

Remarks

MhCard – The .Value settings are:

Setting	Description
0	(Default) Card back (not the face).
1	Ace.
2-10	Deuce through ten.
11	Jack.
12	Queen.
13	King.

MhDice – The default value is one. The range is 1 to 6.

MhGauge, MhSlide – If you assign a number greater than .Max, the control sets the .Value property to .Max. If you assign a number less .Min, the control sets the .Value property to .Min. The range is zero to 32,767.

ValueBorderStyle Property

Description

If .ValueDisplay is True, sets or returns a whether to display a border around the value.

Applies To

MhSpin

Data Type

Integer

Remarks

The .ValueBorderStyle settings are:

Setting	Description
0	(Default) No border
1	Thin border

ValueBottom, ValueLeft, ValueRight, ValueTop Properties

Description

Set or return the area where the control should display the .Value property, when the .ValueDisplay property is True.

Apply To

MhSpin

Data Type

Integer

Remarks

These values must be positive and expressed in twips. The values are relative to the borders of the control. In other words, the top left corner of the picture area is .ValueTop units below the top border and .ValueLeft units to the right of the left border. Similarly, the values for the .ValueBottom and .ValueRight properties are relative to the bottom right edge of the control.

ValueDisplay Property

Description

Sets or returns whether the control displays the .Value property in the defined area.

Applies To

MhSpin

Data Type

Integer (Boolean)

Remarks

The .ValueDisplay settings are:

Setting	Description
True (-1)	Display the .Value in the defined area.
False (0)	(Default) Do not display the .Value.

WindowState Property

Description

- MhAlarm – Sets or returns whether you want the MhAlarm bitmaps to be displayed (instead of the form's icon) when the form is minimized.

Applies To

MhAlarm

Data Type

Integer

Remarks

MhAlarm – The .WindowState settings are:

Setting	Description
0	(Default) Normal display.
1	Alarm bitmaps are displayed in place of form icon.

If you set the .WindowState property to one when your form is minimized, the control displays its bitmaps in place of the Icon associated with the form, provided that the MhAlarm control is activated. At all other times, this property should be zero. The best time to set this property is in the Form_Resize event, as shown in the following example:

```
Sub Form1_Resize
  If Form1.WindowState = 1 Then
    MhAlarm1.WindowState = 1
  Else
    MhAlarm1.WindowState = 0
  End If
End Sub
```