

# Global Majic Software, Inc. Custom Controls

---

## Instrument Custom Control Library

### How To Use:



Toggle Switch Control



LED Control



Aircraft Instrument Controls

What Is Shareware?

Why Register?

Registration.

Product Support

Copyright

Usage



# Aircraft Instrument Control

Properties

Events

---

## Description:

Aircraft Instrument Controls display a variety of small airplane flight instruments including compass, heading indicator, altimeter, vertical speed indicator, airspeed indicator, and horizontal situation indicator. The controls can be used as input controls when MouseControl is set to True. The controls include bevels for a 3D appearance.

## Remarks:

When sizing the controls, keep in mind that smaller instruments update faster.

## Why Register?

---

Register because you find the software useful and you feel the efforts put into writing the code are worth compensating.

Register because you would like to be informed of updates to this software.

Register because you want to encourage the author to spend time to develop other useful shareware software.

Register because you would like to have technical support.

Register because you want to get rid of those annoying sign-on banners.

## Usage

---

The **Instrumentation Custom Control Library** is a collection of *Microsoft Visual Basic* custom controls that can be used by placing them on a *Visual Basic* form. In order for the controls to appear in the *Visual Basic* toolbox, use the **Add File** command to add each control to your project. Then you can position and size the control as desired on the form.

Once the control is placed on the form, use the properties box to set up controls display characteristics and default values. Some properties are only available at design-time and some only at run-time.

A test program using **TOGGLE.VBX**, TOGTEST.EXE, is included for illustration purposes. Double click on this from the File Manager to start the test program which allows the user to see examples of the toggle control. On computers equipped with sound boards, the sound capabilities of the control will be demonstrated,

A test program using **AIR.VBX**, AIRTEST.EXE, is included for illustration purposes. Double click on this from the File Manager to start the test program which allows the user to see each gauge type, and to use either the mouse (click Mouse Control ON and then use the mouse to move the needles on the gauges) or slider controls to adjust the values of the gauges.

A test program using **LED.VBX**, LEDTEST.EXE, is included for illustration purposes. Double click on this from the File Manager to start the test program which allows the user to see each LED type, and to use either the mouse (click Mouse Control ON and then use the mouse to turn the LEDs "on" or "off") or slider controls to adjust the values of the LEDs.

## Registration.

---

Mail completed form with payment to:

Global Majic Software, Inc.  
P.O. Box 322  
Madison, Alabama 35758

Make checks or money orders out to: Global Majic Software, Inc.  
Allow two weeks for personal check processing.

Name: \_\_\_\_\_

Address: \_\_\_\_\_

City, State, Zip: \_\_\_\_\_

Phone: \_\_\_\_\_

Fax: \_\_\_\_\_

CompuServe: \_\_\_\_\_

Internet: \_\_\_\_\_

### ***Instrument Custom Control Library***

	QTY	Amount
REGISTRATION: \$25 each,	_____	_____

SOURCE CODE AND REGISTRATION: \$55 each	_____	_____
--	-------	-------

Foreign addresses add \$5 shipping		_____
------------------------------------	--	-------

Total:	_____
--------	-------

Out of curiosity, where did you  
come across this software: \_\_\_\_\_

# Product Support

---

Product support for *Instrument Custom Control Library* is available to registered users by contacting **Global Majic Software, Inc.** at the following locations:

Global Majic Software, Inc.  
P.O. Box 322  
Madison, Alabama 35758

CompuServe: 73261,3642  
Internet: 73261.3642@compuserve.com

Product Support is free for a period of three (3) months from the date of registration.

## Events:

Change  
Click  
GotFocus  
LostFocus  
Turn

## Properties:

AltBarometer  
AltBarometricPressure  
BackColor  
BevelInner  
BevelOuter  
BevelWidth  
BorderWidth  
CompassStyle  
DisplayMode  
Enabled  
FontSize  
Height  
HelpContextID

HSIBearing  
HSICompass  
HSICourseDelta  
HSICourseDeviation  
HSICourseDisplay  
HSICourseHeading  
HSIDistance  
HSIDistanceDisplay  
HSIGlideSlopeDelta  
HSIGlideSlopeDeviation  
HSINavigationSource  
HSISpeed  
HSITime

HSITimeSpeedDisplay  
Index  
Left  
MouseControl  
MousePointer  
TabIndex  
TabStop  
Tag  
Top  
Value  
Visible  
  
Width



# AltBarometer Property

---

## Description

Enables and disables the barometric pressure indicator displayed on the altimeter.

## Usage

[*form.*]control.**AltBarometer**[ = *boolean* ]

## Setting

The AltBarometer property settings are:

Setting	Description
True	Barometric pressure setting is displayed
False	Barometric pressure setting is hidden

## Remarks

This property only applies when DisplayMode = 3 (Altimeter).

## Data Type

Integer (Boolean)

# DisplayMode Property

---

## Description

Selects the type of instrument being displayed.

## Usage

`[form.]control.DisplayMode[ = integer ]`

## Setting

The DisplayMode property settings are:

Setting	Description
0	Analog Compass
1	Analog/Digital Compass
2	Heading Indicator
3	Altimeter
4	Vertical Speed Indicator
5	Airspeed Indicator
6	HSI, Horizontal Situation Indicator

## Data Type

Integer (Enumerated)

# AltBarometricPressure Property

---

## Description

This property contains the value of the barometric pressure displayed on the altimeter.

## Usage

*[form.]control.AltBarometricPressure[ = single ]*

## Remarks

This property is only applicable when DisplayMode = 3 (Altimeter) and AltBarometer = TRUE. The barometric pressure is bound between 20 and 40 inches of mercury. Changes in the barometric pressure do not affect the altimeter Value.

## Data Type

Single

# BevelOuter Property

---

## Description

Sets or returns the outer shadow style of the control.

## Usage

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

## Setting

The BevelOuter property settings are:

Setting	Description
0	None
1	Raised
2	Inset

## Remarks

This property has no affect when BevelWidth=0. This property is used in conjunction with BevelInner, BevelWidth, and BorderWidth.

## Data Type

Integer (Enumerated)

# BevelInner Property

---

## Description

Sets or returns the inner shadow style of the control.

## Usage

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

## Setting

The BevelInner property settings are:

Setting	Description
0	None
1	Raised
2	Inset

## Remarks

This property has no affect when BevelWidth=0. This property is used in conjunction with BevelOuter, BevelWidth, and BorderWidth.

## Data Type

Integer (Enumerated)

# BevelWidth Property

---

## Description

Sets or returns the shadow sizes of the inner and outer bevels of the control.

## Usage

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

## Remarks

This property works in conjunction with BevelInner, BevelOuter and BorderWidth.

## Data Type

Integer

# BorderWidth Property

---

## Description

Sets or returns the border size between the inner and outer bevels of the control.

## Usage

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

## Remarks

This property works in conjunction with BevelInner, BevelOuter and BevelWidth.

## Data Type

Integer

# CompassStyle Property

---

## Description

Determines the display mode of the compass face.

## Usage

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

## Setting

The CompassStyle property settings are:

Setting	Description
0	Stationary.
1	Floating

## Remarks

When the CompassStyle = 0 (stationary), the compass face remains stationary while the compass needle points to the control's direction. When the CompassStyle = 1 (floating), the compass needle always points to the top of the control while the compass face rotates to the control's direction. This property only applies when DisplayMode = 0 or 1 (compass).

## Data Type

Integer (Enumerated)



# MouseControl Property

---

## Description

Enables and disables mouse input to the control.

## Usage

[*form.*]control.**MouseControl**[ = {TRUE|FALSE} ]

## Setting

The MouseControl property settings are:

Setting	Description
<b>True</b>	Allows the control's <u>Value</u> to be modified with mouse input.
<b>False</b>	Disables mouse input to the control.

## Data Type

Integer (Boolean)

# Value Property

---

## Description

Sets or returns the control's displayed **value**.

## TOGGLE.VBX

### Usage

[*form.*]control.**Value**[ = {TRUE|FALSE} ]

### Remarks

The OnPicture is displayed when the toggle control's **value** = TRUE and the OffPicture is displayed when the toggle control's **value** = FALSE. When the toggle control's **value** is changed to FALSE, the OffWave wave file is played and when the toggle control's value is changed to TRUE, the OnWave wave file is played.

### Data Type

Integer (Boolean)

## AIR.VBX

### Usage

[*form.*]control.**Value**[ = *single* ]

### Remarks

The compass and heading indicator are bound between 0 and 360 degrees. The vertical airspeed indicator is bound between -2000 and 2000 ft per minute. The airspeed indicator is bound between 0 and 170 knots, although the display does not register values below 35 knots.

### Data Type

Single

## LED.VBX

### Usage

[*form.*]control.**Value**[ = *long* ]

### Remarks

This property has two possible uses, depending on the Mode property setting. If Mode = 0, then the Value corresponds to the number of LEDs turned "on". However, if Mode = 1, then the Value may be used to set the LEDs LEDStatus property. The Value may also be returned by setting the LEDStatus property of individual LEDs by using the mouse.

### Data Type

Long

# What Is Shareware?

---

Shareware distribution gives users a chance to try software before buying it. If you try a Shareware program and continue using it, you are expected to register. Individual programs differ in detail -- some request registration while others require it, some specify a maximum trial period. With registration, you get anything from the simple right to continue using the software to an updated program with printed manual.

Copyright laws apply to both Shareware and commercial software, and the copyright holder retains all rights, with a few specific exceptions as stated below. Shareware authors are accomplished programmers, just like commercial programmers, and the programs are of comparable quality. (in both cases there are good programs and bad ones!) The main difference is in the method of distribution. The author specifically grants the right to copy and distribute the software, either to all and sundry or to a specific group. For example, some authors require written permission before a commercial disk vendor may copy their Shareware.

Shareware is a distribution method, not a type of software. You should find software that suits your needs and pocketbook, whether it's commercial or Shareware. The Shareware system makes fitting your needs easier, because you can try before you buy. And because the overhead is low, prices are low also. Shareware has the ultimate money-back guarantee -- if you don't use the product, you don't pay for it.

## DISCLAIMER - AGREEMENT

Users of ***Instrument Custom Control Library*** must accept this disclaimer of warranty: "***Instrument Custom Control Library*** is supplied as is. The author disclaims all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The author assumes no liability for damages, direct or consequential, which may result from the use of ***Instrument Custom Control Library***."

***Instrument Custom Control Library*** is a "shareware library" and is provided at no charge to the user for evaluation. Feel free to share it with your friends, but please do not give it away altered or as part of another system. The essence of "user-supported" software is to provide personal computer users with quality software without high prices, and yet to provide incentive for programmers to continue to develop new products. If you find this library useful and find that you are using ***Instrument Custom Control Library*** and continue to use ***Instrument Custom Control Library*** after a reasonable trial period, you must make a registration payment of \$25 to ***Global Majic Software, Inc.*** The \$25 registration fee will license one copy for use on any one computer at any one time. You must treat this software just like a book. An example is that this software may be used by any number of people and may be freely moved from one computer location to another, so long as there is no possibility of it being used at one location while it's being used by another. Just as a book cannot be read by two different persons at the same time.

Commercial users of ***Instrument Custom Control Library*** must register and pay for their copies of ***Instrument Custom Control Library*** within 30 days of first use or their license is withdrawn. Site-License arrangements may be made by contacting ***Global Majic Software, Inc.***

Anyone distributing ***Instrument Custom Control Library*** for any kind of remuneration must first contact ***Global Majic Software, Inc.*** at the address below for authorization.

You are encouraged to pass a copy of ***Instrument Custom Control Library*** along to your friends for evaluation. Please encourage them to register their copy if they find that they can use it. All registered users will receive a copy of the latest version of ***Instrument Custom Control Library***.

**CONTACTING GLOBAL MAJIC SOFTWARE, INC.**

Global Majic Software, Inc.  
P.O. Box 322  
Madison, Alabama 35758

CompuServe: 73261,3642  
Internet: 73261.3642@compuserve.com

## COPYRIGHT INFORMATION

---

All **Global Majic Software, Inc.** software programs, shareware, and freeware are protected under the copyright laws of the United States and foreign countries. All rights are reserved to **Global Majic Software, Inc.** Violations of copyright laws are investigated by the FBI. Distribution of **Global Majic Software, Inc.** products implies that you have read and agreed to the distribution terms described below:

### INTENT

**Global Majic Software, Inc.** seeks to distribute its shareware as widely as possible. However, we want the end-users of our software to be properly informed that it is shareware.

### DISTRIBUTOR INFORMATION AND LICENSE INFORMATION

The license information and distribution requirements in this document supersede all previous license statements. To continue to distribute **Global Majic Software, Inc.** products, you must adhere to the licensing and distribution requirements below.

If you are a mail order or BBS-type distributor of shareware software, you may distribute these programs as they are, without any changes other than expanding files contained in the ZIP archives. However, you have the responsibility to check from time to time, at a minimum interval of 6 months, for new versions of these programs, and to update your copies in a timely manner. **Global Majic Software, Inc.** will gladly send you a diskette containing the current versions on request.

You must fully identify all **Global Majic Software, Inc.** programs in your advertising, by the program's full name and version, and indicate the registration fee in the program description. The words **Global Majic Software, Inc.** must appear in all program descriptions.

### SHAREWARE DISCLOSURE REQUIRED

All advertising and packaging information including references to **Global Majic Software, Inc.** products must contain a statement explaining the shareware concept (see Registration). Specifically, that statement must explain that shareware software MUST be registered by the user, after a trial period, by paying a registration fee, and that all monies paid for the shareware version are duplication and distribution charges only. All such statements must be clearly displayed in a position where they are likely to be read by potential customers.

### RETAIL RACK AND CD-ROM DISTRIBUTION

If you distribute shareware in a retail setting in racks, store displays, vending machines, at computer fairs, or in any way other than normal BBS or catalog-based sales, you must contact **Global Majic Software, Inc.** for permission to distribute any **Global Majic Software, Inc.** program. Rack or retail-like sales require a special distribution license, normally requiring royalties paid to **Global Majic Software, Inc.** If you distribute shareware on CD-ROM disks, you must also contact **Global Majic Software, Inc.** before including any **Global Majic Software, Inc.** shareware programs on a CD-ROM disk. Normally, permission is granted, but current versions must be included and all old versions of any **Global Majic Software, Inc.** program removed from any CD-ROM disk containing **Global Majic Software, Inc.** products.

# BackgroundPicture Property

---

## Description

Determines the background graphic to be displayed in the LED control.

## Usage

[*form.*]control.**BackgroundPicture**[ = *picture* ]

## Setting

The BackgroundPicture property settings are:

Setting	Description
(none)	No picture is displayed.
(bitmap)	At design-time, specify the bitmap file name to be displayed. At run-time, specify the bitmap using the <b>LoadPicture</b> function.

## Remarks

When setting the picture at design-time, the picture will be saved with the form, and will be compiled into the executable. When the BackgroundPicture property is set to (none), the BackColor property is used to fill the background of the control.

## Data Type

Integer



# Toggle Switch Control

Properties

Events

---

## Description:

Toggle Switch Control is a TRUE/FALSE, YES/NO or ON/OFF indicator. The control displays captions or pictures to represent the toggle state depending on the control's value. The control is capable of playing wave files to indicate the switch turning on or off. It includes bevels for a 3D appearance and blinking capability to draw attention to the control. It also has a built in timed shut-off capability.

## Remarks:

Wave files used with the control will not be stored in the control as the bitmap files are, therefore, all wave files need to accompany your application. Be careful with using many timers and blinking controls on a single form because windows has a limited number of timer handles.

# OnWave Property

---

## Description

Determines the wave file to be played when the toggle control's value is set to TRUE.

## Usage

*[form.]control.OnWave[ = filename ]*

## Remarks

When setting the wave file at design-time, the wave file will not be saved with the form or compiled into the executable. Wave files need to accompany executables. All sounds from the control can be disabled by setting Sound = FALSE.

## Data Type

String



# OnPicture Property

---

## Description

Determines the graphic to be displayed when the toggle control's value = TRUE.

## Usage

[*form.*]control.**OnPicture**[ = *picture* ]

## Setting

The OnPicture property settings are:

Setting	Description
(none)	No picture is displayed.
(bitmap)	At design-time, specify the bitmap file name to be displayed. At run-time, specify the bitmap using the <b>LoadPicture</b> function.

## Remarks

When setting the picture at design-time, the picture will be saved with the form, and will be compiled into the executable. When AutoSize = FALSE, the picture is stretched to fit the controls dimensions, set AutoSize = TRUE to display the picture in its full size.

## Data Type

Integer

# OffWave Property

---

## Description

Determines the wave file to be played when the toggle control's value is set to FALSE.

## Usage

`[form.]control.OffWave[ = filename ]`

## Remarks

When setting the wave file at design-time, the wave file will not be saved with the form or compiled into the executable. Wave files need to accompany executables. All sounds from the control can be disabled by setting Sound = FALSE.

## Data Type

String

# AutoSize Property

---

## Description

Enables the control to automatically size itself to fit its contents.

## Usage

[*form.*]control.**AutoSize**[ = {TRUE|FALSE} ]

## Setting

The AutoSize property settings are:

Setting	Description
TRUE	Automatically sizes itself to fit its contents.
FALSE	Allows the width and height of the control to be set manually.

## Data Type

Integer (Boolean)

## OffBackColor Property

---

### Description

Determines the background color of the toggle control when it is Off (Value = 0) .

### Usage

[*form.*]control.**OffBackColor**[ = *color* ]

### Remarks

This property can be set using the RGB or QBColor functions.

### Data Type

Long.

## Properties:

AutoSize  
BackColor  
BevelInner  
BevelOuter  
BevelWidth  
BlinkMode  
BlinkRate  
BorderWidth  
Enabled  
FontBold  
FontItalic  
FontName  
FontSize

FontStrikethru  
FontUnderline  
Height  
HelpContextID  
Index  
Left  
MousePointer  
OffBackColor  
OffCaption  
OffForeColor  
OffPicture  
OffWave  
OnBackColor

OnCaption  
OnForeColor  
OnPicture  
OnTimer  
OnWave  
Sound  
TabIndex  
TabStop  
Tag  
Top  
Value  
Visible  
Width

# BlinkMode Property

---

## Description

Determines the mode in which the toggle control blinks.

## Usage

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

## Setting

The BlinkMode property settings are:

Setting	Description
<b>0 (Never)</b>	The toggle switch never blinks.
<b>1 (Always)</b>	The toggle switch always blinks.
<b>2 (When On)</b>	The toggle switch blinks only when the switch is On.
<b>3 (When Off)</b>	The toggle switch blinks only when the switch is Off.

## Remarks

This property has no affect when BlinkRate = 0.

## Data Type

Integer (Enumerated)

# OffPicture Property

---

## Description

Determines the graphic to be displayed when the toggle control's value = FALSE

## Usage

[*form.*]control.**OffPicture**[ = *picture* ]

## Setting

The OffPicture property settings are:

Setting	Description
(none)	No picture is displayed.
(bitmap)	At design-time, specify the bitmap file name to be displayed. At run-time, specify the bitmap using the <b>LoadPicture</b> function.

## Remarks

When setting the picture at design-time, the picture will be saved with the form, and will be compiled into the executable. When AutoSize = FALSE, the picture is stretched to fit the controls dimensions. Set AutoSize = TRUE to display the picture in its full size.

## Data Type

Integer

# Sound Property

---

## Description

Enables the control to play wave files when its value is changed.

## Usage

[*form.*]control.Sound[ = {TRUE|FALSE} ]

## Setting

The Sound property settings are:

Setting	Description
TRUE	Enables the control to play wave files.
FALSE	Disables the control from playing wave files.

## Data Type

Integer (Boolean)





# LED Control

Properties

Events

## Description:

This control displays a variety of LED shapes such as rectangular and circular. User supplied bitmaps may also be used to define a LEDs appearance. The mouse can be used for input when the MouseControl property is set to True. The control include bevels for a 3D appearance.

## Remarks:

When sizing the controls, keep in mind that smaller instruments update faster. It should also be noted that when using bitmaps to define LEDs, the LEDColor property has no effect.

## Properties:

AutoSize  
BackColor  
BackgroundPicture  
BevelInner  
BevelOuter  
BevelWidth  
BorderHorizontal  
BorderVertical  
BorderWidth  
Direction  
Enabled  
Height  
HelpContextID

Index  
LEDColor  
LEDHeight  
LEDIndex  
LEDSep  
LEDStatus  
LEDStyle  
LEDWidth  
Left  
Mode  
MouseControl  
Name  
NumLEDs

OffPictureGreen  
OffPictureRed  
OffPictureYellow  
OnPictureGreen  
OnPictureRed  
OnPictureYellow  
TabIndex  
TabStop  
Tag  
Top  
Value  
Visible  
Width

## BorderVertical Property

---

### Description

Sets or returns the vertical border size of the control. This defines the distance between the top edge of the control and the position of the upper left portion of the first LED.

### Usage

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

### Remarks

This property works in conjunction with [BorderHorizontal](#).

### Data Type

Integer

# Mode Property

## Description

Selects the analysis method of the Value property.

## Usage

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

## Setting

The Mode property settings are:

Setting	Description
0	The Value property determines the number of LEDs "on"
1	Perform at bitwise analysis on the Value property

## Data Type

Integer (Enumerated)

## Events:

Click  
GotFocus  
LostFocus

# BorderHorizontal Property

---

## Description

Sets or returns the horizontal border size of the control. This defines the distance between the left edge of the control and the position of the upper left portion of the first LED.

## Usage

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

## Remarks

This property works in conjunction with BorderVertical.

## Data Type

Integer

# Direction Property

## Description

Selects either a forward or backward fill for the LEDs.

## Usage

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

## Setting

The Direction property settings are:

Setting	Description
0	Forward. This is left to right for horizontal Orientation and top to bottom for vertical Orientation.
1	Backward. This is right to left for horizontal Orientation bottom to top for vertical Orientation.

## Data Type

Integer (Enumerated)

# Orientation Property

## Description

Selects either a horizontal or vertical positioning of the LEDs.

## Usage

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

## Setting

The Orientation property settings are:

Setting	Description
0	Horizontal
1	Vertical

## Data Type

Integer (Enumerated)



# LEDStyle Property

## Description

Selects the display type for the LEDs.

## Usage

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

## Setting

The LEDStyle property settings are:

Setting	Description
0	Rectangular
1	Circular
2	Bitmap (uses User supplied bitmaps for the LEDs)

## Data Type

Integer (Enumerated)

# LEDColor Property

## Description

Selects the color for each LED.

## Usage

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

## Setting

The LEDColor property settings are:

Setting	Description
0	Red
1	Yellow
2	Green

## Remark

If LEDStyle=2 (bitmap), color is set using this property in conjunction with the bitmap properties: OnPictureRed, OffPictureRed, OnPictureYellow, OffPictureYellow, OnPictureGreen, and OffPictureGreen. For example, if a multicolored bitmap LED display is desired, the LEDColor property controls which bitmap property (listed above) is accessed.

## Data Type

Integer (Enumerated)

# OnPictureGreen Property

# OnPictureYellow Property

# OnPictureRed Property

## Description

Determines the graphic to be displayed when the LED control's LEDStyle = 2 (bitmap) and the LED is turned on. Although a color is referenced in the name of the property, it in no way limits the color of the bitmap. It merely allows the bitmaps to be selected by the LEDColor property.

## Usage

```
[form.]control.OnPictureGreen[ = picture ]  
[form.]control.OnPictureYellow[ = picture ]  
[form.]control.OnPictureRed[ = picture ]
```

## Setting

The property settings are:

Setting	Description
(none)	No picture is displayed.
(bitmap)	At design-time, specify the bitmap file name to be displayed. At run-time, specify the bitmap using the <b>LoadPicture</b> function.

## Remarks

When setting the picture at design-time, the picture will be saved with the form, and will be compiled into the executable.

## Data Type

Integer

# LEDIndex Properties

## Description

Used to assign properties to individual LEDs. Each LED has a unique LEDIndex.

## Usage

`[form.]control.LEDIndex[ = integer ]`

## Remarks

The LEDColor and LEDStatus may be set separately for each LED. These properties are referenced through the LEDIndex property. For example, for a Green LED, turned "on", in the 4th position:

```
LED1.LEDIndex=3 //numbering starts with 0
LED1.LEDColor=2
LED1.LEDStatus=1
```

## Data Type

Integer

# OffPictureGreen Property

# OffPictureYellow Property

# OffPictureRed Property

## Description

Determines the graphic to be displayed when the LED control's LEDStyle = 2 (bitmap) and the LED is turned off. Although a color is referenced in the name of the property, it in no way limits the color of the bitmap. It merely allows the bitmaps to be selected by the LEDColor property.

## Usage

```
[form.]control.OffPictureGreen[ = picture ]  
[form.]control.OffPictureYellow[ = picture ]  
[form.]control.OffPictureRed[ = picture ]
```

## Setting

The property settings are:

Setting	Description
(none)	No picture is displayed.
(bitmap)	At design-time, specify the bitmap file name to be displayed. At run-time, specify the bitmap using the <b>LoadPicture</b> function.

## Remarks

When setting the picture at design-time, the picture will be saved with the form, and will be compiled into the executable.

## Data Type

Integer

# LEDStatus Property

## Description

Selects the "on" or "off" status of the given LED.

## Usage

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

## Setting

The LEDStatus property settings are:

Setting	Description
0	Off
1	On

## Remarks

Individual LEDs can be turned "on" using the LEDStatus property only if the Mode property is set to "bitwise."

## Data Type

Integer (Enumerated)

# LEDHeight, LEDWidth, LEDSep Properties

## Description

Determines the dimensions of an LED and the spacing between multiple LEDs.

## Usage

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

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

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

## Remarks

The LEDWidth is the dimension between the left and right edges of the LED. The LEDHeight is the dimension between the top and bottom edges of the LED. If multiple LEDs reside in the control then each LED is separated by the dimension specified in the LEDSep property.

## Data Type

Integer

# NumLEDs Property

## Description

Defines the number of LEDs in the given control.

## Usage

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

## Data Type

Integer



# HSIBearing Property

---

## Description

Selects the bearing direction displayed in the HSI instrument.

## Usage

`[form.]control.HSIBearing[ = single ]`

## Remarks

The HSIBearing property value is bound between 0 and 360 degrees. The bearing indicator is only displayed when the DisplayMode = 6 (HSI), HSICompass = TRUE, and HSINavigationSource is set to 0 (VOR1) ,1 (VOR2) ,4 (TCN) or 5 (LRN).

## Data Type

Single.

# HSICompass Property

---

## Description

Enables and disables the compass face displayed on the HSI.

## Usage

[*form.*]control.HSICompass[ = *boolean* ]

## Setting

The HSICompass property settings are:

Setting	Description
True	The aircraft heading and bearing are displayed on a circular compass face.
False	Compass face is not displayed, only course deviation and glide slope deviation are displayed.

## Remarks

This property only applies when DisplayMode = 6 (HSI).

## Data Type

Integer (Boolean)

# HSINavigationSource Property

---

## Description

Determines navigationsource displayed in the lower right hand corner of the HSI control.

## Usage

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

## Setting

The HSINavigationSource property settings are:

### Setting

---

0 - VOR1

1 - VOR2

2 - ILS1

3 - ILS2

4 - TCN

5 - LRN

6 - None

## Remarks

This property only applies when DisplayMode = 6 (HSI).

## Data Type

Integer (Enumerated)

## HSICourseDelta Property

---

### Description

Selects the magnitude of the difference between course deviation markers displayed in the HSI instrument.

### Usage

*[form.]control.HSICourseDelta[ = single ]*

### Remarks

The HSICourseDelta property only applies when DisplayMode = 6 (HSI). Typical values for this property are 2.5 degrees or 7.5 miles.

### Data Type

Single.

# HSICourseHeading Property

---

## Description

Selects the course heading direction displayed in the HSI instrument.

## Usage

`[form.]control.HSICourseHeading[ = single ]`

## Remarks

This property is bound between 0 and 360 degrees and only applies when DisplayMode = 6 (HSI) and HSICompass = TRUE.

## Data Type

Single.

# HSICourseDeviation Property

---

## Description

Determines the course deviation displayed in the HSI instrument.

## Usage

[*form.*]control.HSICourseDeviation[ = *single* ]

## Remarks

The HSICourseDeviation property only applies when DisplayMode = 6 (HSI). The displayable range depends on the value set for HSICourseDelta.

## Data Type

Single.

# HSICourseDisplay Property

---

## Description

Returns the type of course display being displayed in the upper right hand corner of the HSI control.

## Usage

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

## Setting

The HSICourseDisplay property settings are:

Setting	Description
---------	-------------

---

0 - CRS	Displays course direction.
---------	----------------------------

1 - DTK	Displays course direction.
---------	----------------------------

2 - None	Displays nothing.
----------	-------------------

## Remarks

This property is read only and only applies when DisplayMode = 6 (HSI).

## Data Type

Integer (Enumerated)

# HSIDistanceDisplay Property

---

## Description

Determines the distance mode displayed in the upper left hand corner of the HSI control.

## Usage

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

## Setting

The HSIDistanceDisplay property settings are:

Setting	Description
---------	-------------

---

0 - DME	Distance
---------	----------

1 - TCN	TACON
---------	-------

2 - WPT	Waypoint
---------	----------

3 - None	Distance is not displayed.
----------	----------------------------

## Remarks

This property only applies when DisplayMode = 6 (HSI).

## Data Type

Integer (Enumerated)



# HSIDistance Property

---

## Description

Determines the distance displayed in the upper left hand corner of the HSI instrument.

## Usage

`[form.]control.HSIDistance[ = single ]`

## Remarks

This property only applies when DisplayMode = 6 (HSI) and HSIDistanceDisplay = 0 (DME), 1 (TCN), or 2 (WPT). Only three place holders are allowed to display the distance so when the value is less than 100, it is displayed with one decimal point; when the value is between 100 and 999, no decimal point values are displayed; and when the value is 1000 or greater, a value of 999 is displayed. If the distance is set to a negative number, the control will display ---, indicating that the distance is not available.

## Data Type

Single.

# HSIGlideSlopeDelta Property

---

## Description

Selects the magnitude of the difference between glide slope deviation markers displayed in the HSI instrument.

## Usage

[*form.*]control.HSIGlideSlopeDelta[ = *single* ]

## Remarks

This property only applies when DisplayMode = 6 (HSI).

## Data Type

Single.

# HSISpeed Property

---

## Description

Sets the ground speed displayed in the lower left hand corner of the HSI instrument.

## Usage

`[form.]control.HSISpeed[ = single ]`

## Remarks

This property only applies when DisplayMode = 6 (HSI) and HSITimeSpeedDisplay = 1 (SPD). Only three place holders are allowed to display the value so when the value is less than 100, it is displayed with one decimal point; when the value is between 100 and 999, no decimal point values are displayed; and when the value is 1000 or greater, a value of 999 is displayed. If the value is set to a negative number, the control will display ---, indicating that the ground speed is not available.

## Data Type

Single.

# HSIGlideSlopeDeviation Property

---

## Description

Determines the glide slope deviation displayed in the HSI instrument.

## Usage

*[form.]control.HSIGlideSlopeDeviation[ = *single* ]*

## Remarks

This property only applies when DisplayMode = 6 (HSI). The displayable range depends on the value set for HSIGlideSlopeDelta.

## Data Type

Single.

# HSITime Property

---

## Description

Sets the time-to-go displayed in the lower left hand corner of the HSI instrument.

## Usage

`[form.]control.HSITime[ = single ]`

## Remarks

This property only applies when DisplayMode = 6 (HSI) and HSITimeSpeedDisplay = 0 (TTG). Only three place holders are allowed to display the value so when the value is less than 100, it is displayed with one decimal point; when the value is between 100 and 999, no decimal point values are displayed; and when the value is 1000 or greater, a value of 999 is displayed. If the value is set to a negative number, the control will display ---, indicating that the time-to-go is not available.

## Data Type

Single.

# HSITimeSpeedDisplay Property

---

## Description

Determines whether time-to-go or ground speed is displayed in the lower left hand corner of the HSI control.

## Usage

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

## Setting

The HSITimeSpeedDisplay property settings are:

Setting	Description
<b>0 - TTG</b>	Time-To-Go, displays the <u>HSITime</u> value.
<b>1 - SPD</b>	Ground speed, displays the <u>HSISpeed</u> value.
<b>2 - None</b>	Nothing is displayed in the lower left corner of the control.

## Remarks

This property only applies when DisplayMode = 6 (HSI).

## Data Type

Integer (Enumerated)

# BlinkRate Property

---

## Description

Determines the rate (int milliseconds) at which the toggle control blinks.

## Usage

*[form.]control.BlinkRate[ = milliseconds ]*

## Remarks

This property works in conjunction with BlinkMode to determine the blinking characteristics of the toggle control. When the BlinkRate = 0, the toggle control will not blink.

## Data Type

Integer

## OffForeColor Property

---

### Description

Determines the foreground color used to display text for the toggle control when it is Off (Value = 0) .

### Usage

[*form.*]control.**OffForeColor**[ = *color* ]

### Remarks

This property can be set using the RGB or QBColor functions.

### Data Type

Long.



# OnBackColor Property

---

## Description

Determines the background color of the toggle control when it is On (Value = 1) .

## Usage

[*form.*]control.**OnBackColor**[ = *color* ]

## Remarks

This property can be set using the RGB or QBColor functions.

## Data Type

Long.

# OnForeColor Property

---

## Description

Determines the foreground color used to display text for the toggle control when it is On (Value = 1) .

## Usage

[*form.*]control.**OnForeColor**[ = *color* ]

## Remarks

This property can be set using the RGB or QBColor functions.

## Data Type

Long.

# OffCaption Property

---

## Description

Determines the text displayed on the toggle control when it is off (value = 0).

## Usage

`[form.]control.OffCaption[ = stringexpression ]`

## Remarks

The default settings for this property is "Off".

## Data Type

String

# OnCaption Property

---

## Description

Determines the text displayed on the toggle control when it is on (value = 1).

## Usage

[*form.*]control.**OnCaption**[ = *stringexpression* ]

## Remarks

The default settings for this property is "On".

## Data Type

String

## OnTimer Property

---

### Description

Determines the how long (in milliseconds) the toggle control will remain on (Value = 1).

### Usage

*[form.]control.OnTimer[ = milliseconds ]*

### Remarks

When OnTimer = 0, the toggle control will not automatically turn itself off (Value = 0).

### Data Type

Integer



