

# Help for Advanced Clock Control

Version 1.2.00



[Properties](#)

[Events](#)

[Methods](#)

[Tools](#)

---

[Registration](#)

[Order form](#)

[Licensing information](#)

## Description

A control which displays a photorealistic analog clock in different styles.

Nine different hand styles to choose from makes the AdvClock custom control very versatile. User defined textures to all elements, different time formats and many more greatly enhances the AdvClock's usefulness for developing commercial applications.

There is no need for writing your own clock - get your clock control custom written !

Easily control AdvClock color with twelve different color properties. About **120 properties, methods and events** are at your finger tips :

- Normal, raised, lowered bevel and font properties.
- Rotated fonts in Roman or Arabic style.
- User definable number of ticks with numerals and/or pictures.
- Full control of ticks, caption, fonts.
- User definable caption with hyperlink.
- Alarms, sounds, setting time by moving hands, time zones editor, internal time.
- Built-In digital LCD clock with different time formats ( incl. system monitor ).
- Realtime - animated cogwheels.
- Integrated Daylight Map.
- Analog days.
- Moon phases.
- Moon position in zodiacs.

**File Name**

ADVLOCK.OCX

**Object Type**

Control

**Compatibility**

VB 4.0 and up, VC++, MS Office 97 and up, Delphi

**Remarks**

Advanced Clock Control provides a way for the programmer to display a photorealistic analog clock face in the program.

---

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

---

**Events**

**Methods**

## Advanced Clock Control Properties

Properties new in this release are marked with red asterisk (\*).

All of the properties that apply to this control are in this table.

<u>AlarmTime Property</u>	<u>hwnd Property</u>
<u>Align Property</u>	<u>MinTickTexture Property</u>
<u>AnalogDayPic *</u>	<u>MinutesHandColor Property</u>
<u>AutoSize Property</u>	<u>MinuteTexture Property</u>
<u>BackColor Property</u>	<u>Mouselcon Property</u>
<u>BackgroundPicture Property</u>	<u>MousePointer Property</u>
<u>BackMode Property</u>	<u>MTexture Property</u>
<u>BevelInner Property</u>	<u>NumeralsColor Property</u>
<u>BevelOuter Property</u>	<u>NumeralsMode Property</u>
<u>BevelWidth Property</u>	<u>NumeralsOrientation Property</u>
<u>BorderStyle Property</u>	<u>NumeralsPosition Property</u>
<u>BubbleTextStyle Property</u>	<u>NumeralsStyle Property</u>
<u>BubblePic Property</u>	<u>RAMFreeColor</u>
<u>Caption Property</u>	<u>RoundControl</u>
<u>CaptionColor Property</u>	<u>SecHandColor Property</u>
<u>CaptionFont Property</u>	<u>SecHandStyle Property</u>
<u>CaptionOffsetX Property</u>	<u>Shadow Property</u>
<u>CaptionOffsetY Property</u>	<u>ShowAnalogDay *</u>
<u>CaptionStyle Property</u>	<u>ShowCenter Property</u>
<u>CenterAutoSize Property</u>	<u>ShowCPUUsage</u>
<u>CenterColor Property</u>	<u>ShowDayOfMonth *</u>
<u>CenterTexture Property</u>	<u>ShowDayOfWeek *</u>
<u>CenterWidth Property</u>	<u>ShowRAMFree</u>
<u>CircleColor Property</u>	<u>ShowSeconds Property</u>
<u>ClipControls Property</u>	<u>ShowSwapUse</u>
<u>CogWheel</u>	<u>ShowTicks Property</u>
<u>CPUUsageColor</u>	<u>ShowThreads</u>
<u>CTexture Property</u>	<u>ShowZodiac *</u>
<u>Digital Property</u>	<u>SmallSeconds Property</u>
<u>DigitalColor Property</u>	<u>SmallSecTexture Property</u>
<u>DigitalMode Property</u>	<u>Sound Property</u>
<u>DragDrop Property</u>	<u>SoundMode Property</u>
<u>DrawCaption Property</u>	<u>SoundSource Property</u>
<u>EnableAlarm Property</u>	<u>StartTime Property</u>
<u>Enabled Property</u>	<u>STexture Property</u>
<u>EngineMode Property</u>	<u>Stretch Property</u>
<u>ExtHourSound Property</u>	<u>Style Property</u>
<u>ExtQuarterSound Property</u>	<u>SwapUseColor</u>
<u>FillHand Property</u>	<u>ThreadsColor</u>
<u>Font Property</u>	<u>TickBorder Property</u>
<u>FontAutoSize Property</u>	<u>TimeFormat Property</u>
<u>FontStyle Property</u>	<u>TickHourColor Property</u>
<u>FrameDescription Property</u>	<u>TickMinColor Property</u>
<u>hdc Property</u>	<u>TickStyle Property</u>
<u>HoursHandColor Property</u>	<u>TimeZone Property</u>
<u>HourTexture Property</u>	<u>URL Property</u>
<u>HourTickTexture Property</u>	<u>URLColor Property</u>
<u>HTexture Property</u>	<u>URLStyle Property</u>

**WhatsThisHelpID Property**

**Properties**

**Methods**

## **Advanced Clock Control Events**

All of the events that apply to this control are in this table.

**Alarm Event**

**AsyncReadComplete Event**

**Click Event**

**DbClick Event**

**Hide Event**

**KeyDown Event**

**KeyPress Event**

**KeyUp Event**

**MouseDown Event**

**MouseMove Event**

**MouseUp Event**

**Resize Event**

**Show Event**



Properties

Events

## Advanced Clock Control Methods

All of the methods that apply to this control are in this table.

Refresh Method

AsyncRead Method

ShowAboutBox Method

SetSysTime Method

**Properties**

**Events**

**Methods**

## **Advanced Clock Control Tools**

All of the tools available for this control are in this table.

**FrameDesigner**



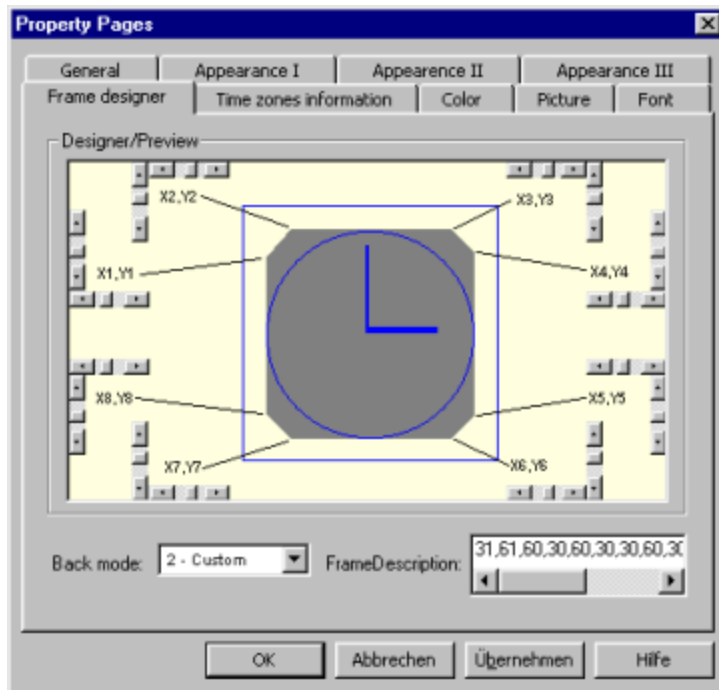
## FrameDesigner Tool

### Description

This visual tool allows to set the geometrical appearance of background  
- polygons or rounded rectangles - ( **FrameDescription** property ) with a preview.

### Usage

The tool is located in Property Pages - page: **Frame Designer**.



## AlarmTime Property

### Description

Determines the time value when **Alarm Event** occurs.

### Usage

```
[Form][control]AlarmTime[= String]
```

### Remarks

- Read/Write
- AlarmTime should be a valid time ( for e. "13:25:00" ).

### Data Type

String

## Align Property

### Description

Determines where the control is displayed on a form.

### Usage

[Form][control]**Align**[= Integer]

### Remarks

- Read/Write
- The value of this property determines the placement of control.

Constant	Value	Description
None	0 ( default )	no alignment
Align Top	1	alignment on top of form
Align Bottom	2	alignment on bottom of form
Align Left	3	alignment on the left side of form
Align Right	4	alignment on the right side of form

### Data Type

Integer ( enumerated )

## AnalogDayPic Property

### Description

Sets a custom picture to the analog day display.

### Usage

```
[Form][control]AnalogDayPic[= LoadPicture[String]]
```

### Remarks

- Read/Write
- To take effect the **ShowAnalogDay** property should be set to **True**.
- **The custom picture should contain the numerals in order: 0,1,2.....,9 .**
- Picture should be a valid picture in format: \*.bmp, \*.gif, \*.jpg, \*.ico.
- As string must be specified the full path -otherwise the picture should be located in
- application directory.

### Data Type

Picture

## AutoSize Property

### Description

Determines whether the clock size is stretched to **BackgroundPicture** size.

### Usage

[Form][control]**AutoSize**[= Boolean]

### Remarks

- Read/Write
- AutoSize = **True** will set **Stretch** to **False** and vice versa.

Constant	Value	Description
<b>False</b>	0 ( default )	no stretching
<b>True</b>	1	clock size is stretched to background picture size

### Data Type

Boolean

## BackColor Property

### Description

Determines the color of clock face background.

### Usage

```
[Form][control]BackColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- BackColor is set to **vbButtonFace** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )

## BackgroundPicture Property

### Description

Sets a custom picture to the clock faces background.

### Usage

```
[Form][control]BackgroundPicture[= LoadPicture[String]]
```

### Remarks

- Read/Write
- Picture should be a valid picture in format: \*.bmp, \*.gif, \*.jpg, \*.ico.
- As string must be specified the full path -otherwise the picture should be located in
- application directory.

### Data Type

Picture

## BackMode Property

### Description

Determines the appearance of clock faces background.

### Usage

[Form][control]**BackMode**[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
<b>Rectangle</b>	0 ( default )	the entire area of control's background is visible
<b>FullCircle</b>	1	a circle masked out the control's background, the inner circle shows the control's background - the mask can be set to custom color by <b>CircleColor</b> property
<b>Custom</b>	2	the geometrical appearance of the background can be determined by user with <b>FrameDescription</b> property or, more comfortable, with <b>FrameDesigner</b> in Property Pages.

### Data Type

Integer ( enumerated )



## BevellInner Property

### Description

Determines the 3D appearance of clock faces inner bevel.

### Usage

[Form][control]**BevellInner**[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
<b>None</b>	0 ( default )	no bevel is drawn
<b>Inset</b>	1	inner bevel appearance = inset
<b>Raised</b>	2	inner bevel appearance = raised

### Data Type

Integer ( enumerated )

## BevelOuter Property

### Description

Determines the 3D appearance of clock faces outer bevel.

### Usage

[Form][control]**BevelOuter**[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
<b>None</b>	0 ( default )	no bevel is drawn
<b>Inset</b>	1	outer bevel appearance = inset
<b>Raised</b>	2	outer bevel appearance = raised

### Data Type

Integer ( enumerated )

## BevelWidth Property

### Description

Determines the distance between inner and outer bevel in pixels.

### Usage

```
[Form][control]BevelWidth[= Integer]
```

### Remarks

- Read/Write
- If NumeralsPosition **Middle** or **Outer** is selected a minimal required BevelWidth is calculated by the control.

### Data Type

Integer

## BorderStyle Property

### Description

Determines the appearance of control's border.

### Usage

[Form][control]**BorderStyle**[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
<b>None</b>	0 ( default )	no border is drawn
<b>FixedSingle</b>	1	a simple black frame is drawn as border
<b>Etched</b>	2	
<b>Bump</b>	3	

### Data Type

Integer ( enumerated )

From author:

The BorderStyle property is completely new implemented. The property is no longer mapped to the VB standard procedure: BorderStyle. This made it possible to incorporate two new border styles - **Etched** and **Bump**.

## BubblePic Property

### Description

Sets a custom picture to the BubbleHelp background.

### Usage

```
[Form][control]BubblePic[= LoadPicture[String]]
```

### Remarks

- Read/Write
- Picture should be a valid picture in format: \*.bmp, \*.gif, \*.jpg, \*.ico.
- This property has affect only to the URLStyle styles **Bubble** and **Roundrec**.
- As string must be specified the full path -otherwise the picture should be located in
- application directory.

### Data Type

Picture

## BubbleTextStyle Property

### Description

Determines the appearance of bubble help's text.

### Usage

[Form][control]**BubbleTextStyle**[= Integer]

### Remarks

- Read/Write

<b>Constant</b>	<b>Value</b>	<b>Description</b>
<b>Default</b>	0 ( default )	normal text is drawn
<b>Inset</b>	1	text is drawn inset
<b>Raised</b>	2	text is drawn raised
<b>Shadowed</b>	3	text is drawn with a dark grey shadow

### Data Type

Integer ( enumerated )

## Caption Property

### Description

Determines the text of control's caption.

### Usage

```
[Form][control]Caption[= String]
```

### Remarks

- Read/Write
- & - character is interpreted as normal character.
- Text can be shown in multiple lines using **vbCrLf** ( Chr(10) + Chr(13) ).
- The caption's hyperlink function is available only and box or bubble help is displayed only when **Caption** is not empty.

### Data Type

String

## CaptionColor Property

### Description

Determines the color of control's caption.

### Usage

```
[Form][control]CaptionColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- CaptionColor is set to **vbBlack** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )



## CaptionFont Property

### Description

Determines the control's caption font.

### Usage

```
[Form][control]CaptionFont[= FontName]
```

### Remarks

- Read/Write
- FontBold, FontItalic, FontUnderline, ... properties can be set **only by Property Page**.
- If **FontAutoSize** is **False** - the size of CaptionFont can be set by user.
- If **FontAutoSize** is **True** - the size of CaptionFont is determined by control.
- Default: MS Sans Serif

### Data Type

Font

## CaptionOffsetX Property

### Description

Determines the X-position of control's Caption.

### Usage

[Form][control]**CaptionOffsetX**[= Single]

### Remarks

- Read/Write
- Valid values: from: **0 to 2.0**.
- Default: **1.0**

### Data Type

Single

## CaptionOffsetY Property

### Description

Determines the Y-position of control's Caption.

### Usage

[Form][control]**CaptionOffsetY**[= Single]

### Remarks

- Read/Write
- Valid values: from: **-1.5 to 1.5**.
- Default: **-1.0**

### Data Type

Single

## CaptionStyle Property

### Description

Determines the 3D appearance of control's caption.

### Usage

[Form][control]**CaptionStyle**[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
<b>None</b>	0 ( default )	simple text is drawn
<b>Inset</b>	1	inset text is drawn
<b>Raised</b>	2	raised text is drawn
<b>Shadowed</b>	3	gray shadow is drawn

### Data Type

Integer ( enumerated )

## CenterAutoSize Property

### Description

Determines whether the size of center is calculated by control.

### Usage

```
[Form][control]CenterAutoSize[= Boolean]
```

### Remarks

- Read/Write

Constant	Value	Description
False	0	size of center is user defined
True	1 ( default )	size of center is calculated by control

### Data Type

Boolean

## CenterColor Property

### Description

Determines the color of control's center.

### Usage

```
[Form][control]CenterColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- CenterColor is set to **vbBlack** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )

## CenterTexture Property

### Description

Sets a custom picture to the clock faces center.

### Usage

```
[Form][control]CenterTexture[= LoadPicture[String]]
```

### Remarks

- Read/Write
- Picture should be a valid picture in format: \*.bmp, \*.gif, \*.jpg, \*.ico.
- As string must be specified the full path -otherwise the picture should be located in application directory.
- The picture will be stretched to **CenterWidth** !
- The picture is visible only when **CTexture** is **True** !

### Data Type

Picture

## CenterWidth Property

### Description

Determines the diameter of control's center in pixels.

### Usage

```
[Form][control]CenterWidth[= Integer]
```

### Remarks

- Read/Write
- CenterWidth has an effect only when CenterAutoSize is **False**.

### Data Type

Integer



## CircleColor Property

### Description

Determines the color of control's circle mask.

### Usage

```
[Form][control]CircleColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- CircleColor is set to **vbBlack** by default.
- CircleColor has an effect only to **BackMode = FullCircle (1) and Custom (2)**.

### Data Type

OLE\_COLOR ( Unsigned Long )

## ClipControls Property

### Description

Determines whether a part or the entire area of control is redrawn by system.

### Usage

[Form][control]**ClipControls**[= Boolean]

### Remarks

- Read

Constant	Value	Description
False	0	entire area is redrawn
True	1 ( default )	part of area is redrawn

### Data Type

Boolean

## CogWheel Property

### Description

Determines whether animated cogwheels are displayed.

### Usage

[Form][control]**CogWheel**[= Boolean]

### Remarks

- Read/Write
- The performance of animation depends on the hardware used.
- To enhance performance set **Shadow** to **False**.
- CogWheel = True will disable **SmallSeconds** property, **Digital** property, **ShowZodiac** and vice versa.

Constant	Value	Description
False	0	cogwheels are displayed
True	1 ( default )	no cogwheels are displayed

### Data Type

Boolean

## CPUUsageColor Property

### Description

Determines the color of CPUUsage in TimeFormat : SysInfo (6).

### Usage

[Form][control]CPUUsageColor[= OLE\_COLOR]

### Remarks

- Read/Write
- CPUUsageColor is set to **&H800080** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )

## CTexture Property

### Description

Determines whether a texture is drawn to the control's center.

### Usage

[Form][control]**CTexture**[= Boolean]

### Remarks

- Read/Write
- Texture can be specified by set the CenterTexture property.

Constant	Value	Description
False	0 ( default )	no texture is drawn
True	1	texture is drawn

### Data Type

Boolean

## Digital Property

### Description

Determines whether a digital clock is displayed.

### Usage

[Form][control]**Digital**[= Boolean]

### Remarks

- Read/Write
- The size and position of digital clock is determined by control.
- The appearance of digital clock can be specified by **DigitalMode** property.
- Content of digital clock can be specified by **TimeFormat** property.
- Font in **DigitalMode = Default** is determined by **Font** property.
- Digital = True will disable **SmallSeconds** property, **CogWheel** property **ShowZodiac** property and vice versa.

Constant	Value	Description
False	0 ( default )	no digital clock is displayed
True	1	digital clock is displayed

### Data Type

Boolean

## DigitalColor Property

### Description

Determines the color of digital clock background.

### Usage

```
[Form][control]DigitalColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- DigitalColor is set to **vbWhite** by default.
- DigitalColor has an effect only when **DigitalMode = Default**.
- DigitalColor takes effect also to background of system monitor - **TimeFormat : 6**

### Data Type

OLE\_COLOR ( Unsigned Long )

## DigitalMode Property

### Description

Determines the style digital clock is displayed.

### Usage

[Form][control]**DigitalMode**[= Integer]

### Remarks

- Read/Write
- The size and position of digital clock is determined by control.
- Content of digital clock can be specified by **TimeFormat** property.
- Font in **DigitalMode = Default** is determined by **Font** property.
- The DigitalMode property does not effect to **TimeFormat = 4** ( Daylight Map ).

Constant	Value	Description
<b>Default</b>	0 ( default )	digital clock is displayed as simple text
<b>LCD</b>	1	digital clock is displayed with LCD segments

### Data Type

Integer ( enumerated )



## DragDrop Property

### Description

Determines whether Drag&Drop is allowed.

### Usage

[Form][control]**DragDrop**[= Boolean]

### Remarks

- Read/Write
- Via Drag&Drop texture can be set to minute hand, hour hand, center, SmallSeconds and background.
- For displaying textures the appropriate property ( **CTexture**, **HTexture**, **MTexture**, **STexture** ) should be set to **True**.
- For texture will be accepted files in \*.bmp, \*.ico, \*.gif, and \*.jpg format.
- Path to pictures added by Drag&Drop will be stored in registry and pictures will be loaded on next start if DragDrop = **True** !!!

Constant	Value	Description
<b>False</b>	0 ( default )	Drag&Drop is allowed
<b>True</b>	1	Drag&Drop is not allowed

### Data Type

Boolean

From author:

It is not the standard OLE Drag&Drop implementation of VB5 and 6 ! This works great in VB5 and 6, really !

Try it in VB4 IDE and you will raise instead of a trappable error an invalid page fault at OLE32.dll ( don't ask me - ask Microsoft ). So I made it the API way.

## DrawCaption Property

### Description

Determines whether a caption is drawn.

### Usage

[Form][control]**DrawCaption**[= Boolean]

### Remarks

- Read/Write

Constant	Value	Description
False	0	no caption is drawn
True	1 ( default )	caption is drawn

### Data Type

Boolean

## EnableAlarm Property

### Description

Determines whether a **Alarm** Event is raised when system time reaches **AlarmTime**.

### Usage

[Form][control]**EnableAlarm**[= Boolean]

### Remarks

- Read/Write
- **This property will not work when [StartTime](#) property is set !**

Constant	Value	Description
False	0 ( default )	no Alarm Event is raised
True	1	Alarm Event is raised

### Data Type

Boolean

## Enabled Property

### Description

Determines whether the control can respond to user-generated events.

### Usage

[Form][control]**Enabled**[= Boolean]

### Remarks

- Read/Write
- This property is **not** the standard Enabled - procedure of VB. This implementation draws a grayed background ( picture ) in addition to the common Enabled/Disabled features to tell the user that he can not generate events.

Constant	Value	Description
<b>False</b>	0	control can not respond to user-generated events
<b>True</b>	1 ( default )	control can respond to user-generated events

### Data Type

Boolean

## EngineMode Property

### Description

Determines the type of clock timer engine.

### Usage

[Form][control]**EngineMode**[= Integer]

### Remarks

- Read/Write
- EngineMode = **Internal** is available only if **StartTime** is not empty !
- EngineMode = **Internal** is available only if no special **TimeZone** is selected !

Constant	Value	Description
<b>System</b>	0 ( default )	clock displays system ( PC's ) time
<b>Internal</b>	1	clock displays actual time offset from <b>StartTime</b>

### Data Type

Integer ( enumerated )

## ExtHourSound Property

### Description

Determines the external sound file played every hour.

### Usage

[Form][control]**ExtHourSound**[= String]

### Remarks

- Read/Write
- String should be a valid sound file in format: \*.wav.
- As string must be specified the full path -otherwise the sound file should be located in application directory.
- To play external hour sound **Sound** property should be set to **True**.
- To play external hour sound **SoundSource** property should be set to **1**.
- To play external hour sound **SoundMode** property should be set to **0** or **2**.

### Data Type

String

## ExtQuarterSound Property

### Description

Determines the external sound file played every quarter of an hour.

### Usage

[Form][control]**ExtQuarterSound**[= String]

### Remarks

- Read/Write
- String should be a valid sound file in format: \*.wav.
- As string must be specified the full path -otherwise the sound file should be located in application directory.
- To play external quarter sound **Sound** property should be set to **True**.
- To play external quarter sound **SoundSource** property should be set to **1**.
- To play external quarter sound **SoundMode** property should be set to **1** or **2**.

### Data Type

String

## FillHand Property

### Description

Determines whether hands are displayed filled with color.

### Usage

[Form][control]**FillHand**[= Boolean]

### Remarks

- Read/Write
- FillHand property does not effect to hand Style : **Floating, OldFashion, Chronometer** and **Modern**.
- FillHand property takes effect to SecondHandStyle : **Psychedelic**

Constant	Value	Description
<b>False</b>	0	hands are not filled with color
<b>True</b>	1 ( default )	hands will be filled with appropriate color

### Data Type

Boolean



## Font Property

### Description

Determines the control's numerals font.

### Usage

```
[Form][control]Font[= FontName]
```

### Remarks

- Read/Write
- This property applies also to font of URL.
- FontSize, FontBold, FontItalic, FontUnderline, ... properties can be set **only by Property Page**.
- Default: MS Arial

### Data Type

Font

From author:

Because rotated text can be created only from True Typ Fonts - after initializing the control will select the first True Type Font from VB's font enumeration if default font ( Arial ) is missing on the system. This also happens when a Bitmap Font is selected by user.

## FontAutoSize Property

### Description

Determines whether the size of numerals **Font** and **CaptionFont** is calculated by control ( depending on control's size ) or user-defined.

### Usage

[Form][control]**FontAutoSize**[= Boolean]

### Remarks

- Read/Write

Constant	Value	Description
False	0	user-defined font size
True	1 ( default )	font size is calculated by system

### Data Type

Boolean

## FontStyle Property

### Description

Determines the 3D appearance of control's numerals.

### Usage

[Form][control]**FontStyle**[= Integer]

### Remarks

- Read/Write

<b>Constant</b>	<b>Value</b>	<b>Description</b>
<b>None</b>	0 ( default )	simple numerals are drawn
<b>Inset</b>	1	inset numerals are drawn
<b>Raised</b>	2	raised numerals are drawn

### Data Type

Integer ( enumerated )

## FrameDescription Property

### Description

Determines the geometrical appearance of custom made background in two styles.

### Usage

```
[Form][control]FrameDescription[= String]
```

#### Style: Polygon

The string **must** contain the X- and Y coordinates ( **in Pixel** ) for eight points on the control in the following manner:

```
AdvClockControl.FrameDescription = "X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5,X6,Y6,X7,Y7,X8,Y8"
```

#### Style: RoundedRectangle

The string **must** contain a leading "R," ( not case-sensitive ), the X- and Y coordinates ( **in Pixel** ) for two points and in X3 and Y3 the parameters for roundings in the following manner:

```
AdvClockControl.FrameDescription = "R,X1,Y1,X2,Y2,X3,Y3"
```

### Remarks

- Read/Write
- this property has effect only with **BackMode : 2 ( Custom )**
- **It is highly recommended to make all settings ( with preview ) with [FrameDesigner](#) from Property Pages and, if necessary, edit manually**

### Data Type

String

## hdc Property

### Description

Returns a handle to the control's device context.

### Usage

```
[Form][control]hdc[= Long]
```

### Remarks

- Read

### Data Type

Long

## HoursHandColor Property

### Description

Determines the color used to display the hour hand.

### Usage

[Form][control]HoursHandColor[= OLE\_COLOR]

### Remarks

- Read/Write
- HoursHandColor is set to &H808000 by default.
- If HTexture is set to **True** HoursHandColor is set to &H00C0C0C0 ( gray ) by control to avoid color effects.

### Data Type

OLE\_COLOR ( Unsigned Long )

## HourTexture Property

### Description

Sets a custom picture to the hour hand.

### Usage

```
[Form][control]HourTexture[= LoadPicture[String]]
```

### Remarks

- Read/Write
- Picture should be a valid picture in format: \*.bmp, \*.gif, \*.jpg, \*.ico.
- As string must be specified the full path -otherwise the picture should be located in application directory.
- The picture will be stretched to hour hand size !
- The picture is visible only when HTexture is **True** !

### Data Type

Picture

## HourTickTexture Property

### Description

Sets a custom picture to the hour ticks.

### Usage

```
[Form][control]HourTickTexture[= LoadPicture[String]]
```

### Remarks

- Read/Write
- Picture should be a valid picture in format: \*.bmp, \*.gif, \*.jpg, \*.ico.
- As string must be specified the full path -otherwise the picture should be located in application directory.
- The picture will be displayed in original size !
- The picture is visible only when TickStyle is set to **1** or **2**.
- The picture is visible only when ShowTicks is set to **2** or **3**.

### Data Type

Picture



## HTexture Property

### Description

Determines whether a texture is drawn to the hour hand.

### Usage

[Form][control]**HTexture**[= Boolean]

### Remarks

- Read/Write
- Texture can be specified by set the HourTexture property.

Constant	Value	Description
False	0 ( default )	no texture is drawn
True	1	texture is drawn

### Data Type

Boolean

## hwnd Property

### Description

Returns a handle to the control's window.

### Usage

```
[Form][control]hwnd[= Long]
```

### Remarks

- Read

### Data Type

Long

## MinTickTexture Property

### Description

Sets a custom picture to the minute ticks.

### Usage

```
[Form][control]MinTickTexture[= LoadPicture[String]]
```

### Remarks

- Read/Write
- Picture should be a valid picture in format: \*.bmp, \*.gif, \*.jpg, \*.ico.
- As string must be specified the full path -otherwise the picture should be located in application directory.
- The picture will be displayed in original size !
- The picture is visible only when TickStyle is set to **1** or **2**.
- The picture is visible only when ShowTicks is set to **1** or **3**.

### Data Type

Picture

## MinutesHandColor Property

### Description

Determines the color used to display the minute hand.

### Usage

[Form][control]**MinutesHandColor**[= OLE\_COLOR]

### Remarks

- Read/Write
- MinutesHandColor is set to &H808000 by default.
- If **MTexture** is set to **True** MinutesHandColor is set to &H00C0C0C0 ( gray ) by control to avoid color effects.

### Data Type

OLE\_COLOR ( Unsigned Long )

## MinuteTexture Property

### Description

Sets a custom picture to the minute hand.

### Usage

```
[Form][control]MinuteTexture[= LoadPicture[String]]
```

### Remarks

- Read/Write
- Picture should be a valid picture in format: \*.bmp, \*.gif, \*.jpg, \*.ico.
- As string must be specified the full path -otherwise the picture should be located in application directory.
- The picture will be stretched to minute hand size !
- The picture is visible only when **MTexture** is **True** !

### Data Type

Picture

## MouseIcon Property

### Description

Sets a custom mouse icon.

### Usage

Set [Form][control]**MouseIcon**[=Picture]

### Remarks

- Read/Write
- For further details see VB help.

### Data Type

Picture

## MousePointer Property

### Description

Sets the type of mouse pointer displayed when over part of an object.

### Usage

```
[Form][control]MousePointer[=Integer]
```

### Remarks

- Read/Write
- For further details see VB help.

### Data Type

Integer ( enumerated )

## MTexture Property

### Description

Determines whether a texture is drawn to the minute hand.

### Usage

[Form][control]**MTexture**[= Boolean]

### Remarks

- Read/Write
- Texture can be specified by set the **MinuteTexture** property.

Constant	Value	Description
<b>False</b>	0 ( default )	no texture is drawn
<b>True</b>	1	texture is drawn

### Data Type

Boolean



## NumeralsColor Property

### Description

Determines the color used to display the numerals.

### Usage

```
[Form][control]NumeralsColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- This property takes effect to the color of days in **ShowDayOfWeek** and **ShowDayOfMonth**.
- NumeralsColor is set to **vbBlack** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )

## NumeralsMode Property

### Description

Determines the mode used to display numerals.

### Usage

[Form][control]**NumeralsMode**[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
<b>AllNumerals</b>	0 ( default )	numerals 1 to 12 are displayed
<b>FourNumerals</b>	1	numerals 3, 6, 9, 12 are displayed
<b>None</b>	2	no numerals are displayed

### Data Type

Integer ( enumerated )

## NumeralsOrientation Property

### Description

Determines the orientation used to display numerals.

### Usage

[Form][control]**NumeralsOrientation**[= Integer]

### Remarks

- Read/Write
- This property affects only to **NumeralsMode** : **AllNumerals**

Constant	Value	Description
<b>Default</b>	0 ( default )	numerals are displayed in normal way
<b>Rotated</b>	1	numerals are displayed rotated

### Data Type

Integer ( enumerated )

## NumeralsPosition Property

### Description

Determines the position of numerals.

### Usage

[Form][control]NumeralsPosition[= Integer]

### Remarks

- Read/Write
- If NumeralsPosition = **Outer** or **Middle** a minimal required **BevelWidth** is calculated by control.

Constant	Value	Description
<b>Inner</b>	0 ( default )	numerals are placed in the inner circle
<b>Outer</b>	1	numerals are placed in the outer circle
<b>Middle</b>	2	numerals are placed on tick position

### Data Type

Integer ( enumerated )

## NumeralsStyle Property

### Description

Determines the style used to display numerals.

### Usage

[Form][control]**NumeralsStyle**[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
<b>Roman</b>	0 ( default )	numerals are displayed in roman style
<b>Arabic</b>	1	numerals are displayed in arabic style

### Data Type

Integer ( enumerated )

## RAMFreeColor Property

### Description

Determines the color of RAMFree in TimeFormat : **SysInfo (6)**.

### Usage

```
[Form][control]RAMFreeColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- RAMFreeColor is set to **vbGreen** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )

## RoundControl Property

### Description

Determines whether the **usercontrol window** will appear in **irregular** form.

### Usage

[Form][control]**RoundControl**[= Boolean]

### Remarks

- Read/Write
- the appearance of the irregular usercontrol window depends on the ratio Height / Width. A ratio of 1 will draw a circle, other will draw ellipses.

### Data Type

Boolean

## SecHandColor Property

### Description

Determines the color used to display the second hand.

### Usage

[Form][control]**SecHandColor**[= OLE\_COLOR]

### Remarks

- Read/Write
- SecHandColor is set to **vbBlack** by default.
- SecHandColor takes effect on color of hands in **ShowDayOfWeek** and **ShowDayOfMonth**.

### Data Type

OLE\_COLOR ( Unsigned Long )



## SecHandStyle Property

### Description

Determines the style used to display second hand.

### Usage

[Form][control]**SecHandStyle**[= Integer]

### Remarks

- Read/Write
- SecHandStyle : “Chrono” will set **ShowCenter** to **False**
- SecHandStyle : “Sport” will set **ShowCenter** to **False**

Constant	Value	Description
<b>Default</b>	0 ( default )	second hand is displayed as single line
<b>Extended</b>	1	second hand is displayed as single line with circle
<b>Chrono</b>	2	second hand is displayed in chronometer style
<b>Sport</b>	3	second hand is displayed as an arrow
<b>Psychedelic</b>	4	displays seconds as dots with changing colors

### Data Type

Integer ( enumerated )

## Shadow Property

### Description

Determines whether a **transparent** shadow of all hands and /or cogwheels is drawn.

### Usage

[Form][control]**Shadow**[= Boolean]

### Remarks

- Read/Write
- Shadow has an effect only to **Style Windows, Fat, PopArt, Chronometer, Modern, OldFashion, Floating** and **CogWheel** properties.

Constant	Value	Description
False	0 ( default )	no shadow is drawn
True	1	shadow is drawn

### Data Type

Boolean

## ShowAnalogDay property

### Description

Determines whether the day of month is displayed like in analog clocks.

### Usage

[Form][control]ShowAnalogDay[= Boolean]

### Remarks

- Read/Write
- A custom “font” can be set to the analog day display by using the AnalogDayPic property.
- From **10 PM to 12 PM** the numerals are moved smooth to the next position.

Constant	Value	Description
False	0 ( default )	no analog day is displayed
True	1	analog day is displayed

### Data Type

Boolean

## ShowCenter property

### Description

Determines whether a center is displayed.

### Usage

[Form][control]**ShowCenter**[= Boolean]

### Remarks

- Read/Write
- Color can be set to center by **CenterColor**.
- Texture can be added by **CTexture = True** and **CenterTexture**.
- CenterTexture is set to **False** when **SecHandStyle** “Chrono” or “Sport” are chosen.

<b>Constant</b>	<b>Value</b>	<b>Description</b>
<b>False</b>	0	no center is drawn
<b>True</b>	1 ( default )	center is drawn

### Data Type

Boolean

## ShowCPUUsage Property

### Description

Determines whether the actual CPU usage ( in percent ) is displayed in digital clock.

### Usage

[Form][control]ShowCPUUsage[= Boolean]

### Remarks

- Read/Write
- This property takes effect only with **Digital : True** and **TimeFormat : SysInfo (6)**.

Constant	Value	Description
False	0	no CPU usage is displayed
True	1( default )	CPU usage is displayed

### Data Type

Boolean

## ShowDayOfMonth Property

### Description

Determines whether the current day of month is displayed in a separate small clock, located right from center.

### Usage

[Form][control]**ShowDayOfMonth**[= Boolean]

### Remarks

- Read/Write
- The color of days is determined by **NumeralsColor**.
- The color of hand is determined by **SecHandColor** and texture can be added by **SmallSecTexture**.

<b>Constant</b>	<b>Value</b>	<b>Description</b>
<b>False</b>	0 ( default )	no day of month is displayed
<b>True</b>	1	day of month is displayed

### Data Type

Boolean

## ShowDayOfWeek Property

### Description

Determines whether the current day of week is displayed in a separate small clock, located left from center.

### Usage

[Form][control]**ShowDayOfWeek**[= Boolean]

### Remarks

- Read/Write
- The color of days is determined by **NumeralsColor**.
- The color of hand is determined by **SecHandColor** and texture can be added by **SmallSecTexture**.

<b>Constant</b>	<b>Value</b>	<b>Description</b>
<b>False</b>	0 ( default )	no day of week is displayed
<b>True</b>	1	day of week is displayed

### Data Type

Boolean

## ShowThreads Property

### Description

Determines whether the actual number of threads is displayed in digital clock.

### Usage

[Form][control]**ShowThreads**[= Boolean]

### Remarks

- Read/Write
- This property takes effect only with **Digital : True** and **TimeFormat : SysInfo (6)**.

Constant	Value	Description
False	0	no number of threads is displayed
True	1( default )	number of threads is displayed

### Data Type

Boolean



## ShowSwapUse Property

### Description

Determines whether the actual swapfile usage ( in percent ) is displayed in digital clock.

### Usage

[Form][control]ShowSwapUse[= Boolean]

### Remarks

- Read/Write
- This property takes effect only with **Digital : True** and **TimeFormat : SysInfo (6)**.

Constant	Value	Description
False	0	no swapfile usage is displayed
True	1( default )	swapfile usage is displayed

### Data Type

Boolean

## ShowRAMFree Property

### Description

Determines whether the actual free RAM ( in percent ) is displayed in digital clock.

### Usage

[Form][control]ShowRAMFree[= Boolean]

### Remarks

- Read/Write
- This property takes effect only with **Digital : True** and **TimeFormat : SysInfo (6)**.

Constant	Value	Description
False	0	no free RAM is displayed
True	1( default )	free EAM is displayed

### Data Type

Boolean

## ShowSeconds Property

### Description

Determines whether a second hand is displayed.

### Usage

[Form][control]**ShowSeconds**[= Boolean]

### Remarks

- Read/Write
- ShowSeconds = **True** will set **SmallSeconds** to **False** and vice versa.

Constant	Value	Description
<b>False</b>	0	no second hand is displayed
<b>True</b>	1 ( default )	second hand is displayed

### Data Type

Boolean

## ShowTicks Property

### Description

Determines the mode ticks are displayed.

### Usage

[Form][control]**ShowTicks**[= Integer]

### Remarks

- Read/Write

<b>Constant</b>	<b>Value</b>	<b>Description</b>
<b>None</b>	0	no ticks are displayed
<b>TickMinute</b>	1	minute ticks are displayed
<b>TickHour</b>	2	hour ticks are displayed
<b>Both</b>	3 ( default )	both ticks are displayed

### Data Type

Integer ( enumerated )

## ShowZodiac property

### Description

Determines whether the current position of moon in relation to zodiacs is displayed.

### Usage

[Form][control]**ShowZodiac**[= Boolean]

### Remarks

- Read/Write
- ShowZodiac works correct from January 27 1999.
- ShowZodiac is set to **False** when Digital, CogWheel or SmallSeconds are choosed ( and vice versa ).

Constant	Value	Description
<b>False</b>	0 ( default )	no zodiac is displayed
<b>True</b>	1	zodiac is displayed

### Data Type

Boolean

## SmallSeconds Property

### Description

Determines whether a separate second clock is displayed.

### Usage

[Form][control]**SmallSeconds**[= Boolean]

### Remarks

- Read/Write
- SmallSeconds = **True** will set **ShowSeconds** to **False** and vice versa.
- SmallSeconds = **True** will set **Digital**, **CogWheel** and **ShowZodiac** to **False** and vice versa.

Constant	Value	Description
<b>False</b>	0 ( default )	no separate second clock is displayed
<b>True</b>	1	a separate second clock is displayed

### Data Type

Boolean

## SmallSecTexture Property

### Description

Sets a custom picture to the separate second clock.

### Usage

```
[Form][control]SmallSecTexture[= LoadPicture[String]]
```

### Remarks

- Read/Write
- Picture should be a valid picture in format: \*.bmp, \*.gif, \*.jpg, \*.ico.
- As string must be specified the full path -otherwise the picture should be located in application directory.
- The picture will be stretched to separate second clock size !
- The picture is visible only when **STexture** is **True** !
- This property takes effect to the texture of **ShowDayOfWeek** and **ShowDayOfMonth**.

### Data Type

Picture

## Sound Property

### Description

Determines whether sound is enabled.

### Usage

[Form][control]**Sound**[= Boolean]

### Remarks

- Read/Write

Constant	Value	Description
False	0	no sound will be played
True	1 ( default )	sound will be played

### Data Type

Boolean



## SoundMode Property

### Description

Determines the mode sound is played.

### Usage

[Form][control]**SoundMode**[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
Hour	0	sound will be played every hour
Quarter	1	sound will be played every quarter of an hour
Both	2 ( default )	both

### Data Type

Integer ( enumerated )

## SoundSource Property

### Description

Determines the source of sound.

### Usage

[Form][control]**SoundSource**[= Integer]

### Remarks

- Read/Write
- If SoundSource = 1 the source of sound should be determined by ExtHourSound and ExtQuarterSound.

Constant	Value	Description
	0 ( default )	sound from internal ressource will be played
	1	sound from external source will be played

### Data Type

Integer

## StartTime Property

### Description

Determines the time value the clock started with in EngineMode : **Internal**.

### Usage

[Form][control]**StartTime**[= String]

### Remarks

- Read/Write
- StartTime should be a valid time ( for e. "13:25:00" ).
- **Setting StartTime will disable the EnableAlarm property !**

### Data Type

String

## STexture Property

### Description

Determines whether a texture is set to the separate second clock.

### Usage

[Form][control]**STexture**[= Boolean]

### Remarks

- Read/Write
- Texture can be specified by set the **SmallSecTexture** property.

Constant	Value	Description
<b>False</b>	0 ( default )	no texture is drawn
<b>True</b>	1	texture is drawn

### Data Type

Boolean

## Stretch Property

### Description

Determines whether **BackgroundPicture** is stretched to clock size.

### Usage

[Form][control]**Stretch**[= Boolean]

### Remarks

- Read/Write
- Stretch = **True** will set **AutoSize** to **False** and vice versa.

Constant	Value	Description
<b>False</b>	0	no stretching
<b>True</b>	1 ( default )	background picture is stretched to clock size

### Data Type

Boolean

## Style Property

### Description

Determines the style hands are displayed.

### Usage

[Form][control]**Style**[= Integer]

### Remarks

- Read/Write
- If Style = **Pencil** HTexture and MTexture will be set to **False**.

Constant	Value	Description
<b>Windows</b>	0 ( default )	
<b>Fat</b>	1	
<b>PopArt</b>	2	
<b>Pencil</b>	3	
<b>OldFashion</b>	4	
<b>Floating</b>	5	
<b>ZigZag</b>	6	
<b>Chronometer</b>	7	
<b>Modern</b>	8	
<b>Custom</b>	99	<b>not available in this release</b>

### Data Type

Integer ( enumerated )

## SwapUseColor Property

### Description

Determines the color of SwapUse in TimeFormat : **SysInfo (6)**.

### Usage

```
[Form][control]SwapUseColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- SwapUseColor is set to **vbBlue** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )

## ThreadsColor Property

### Description

Determines the color of Threads in TimeFormat : **SysInfo (6)**.

### Usage

```
[Form][control]ThreadsColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- ThreadsColor is set to **vbMagenta** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )



## TickBorder Property

### Description

Determines the appearance of ticks displayed.

### Usage

[Form][control]**TickBorder**[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
<b>None</b>	0 ( default )	no border
<b>Inset</b>	1	inset border
<b>Raised</b>	2	raised border

### Data Type

Integer ( enumerated )

## TickHourColor Property

### Description

Determines the color used to display hour ticks.

### Usage

```
[Form][control]TickHourColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- TickHourColor is set to **vbBlue** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )

## TickMinColor Property

### Description

Determines the color used to display minute ticks.

### Usage

```
[Form][control]TickMinColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- TickMinColor is set to **vbBlue** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )

## TickStyle Property

### Description

Determines the style of ticks displayed.

### Usage

[Form][control]**TickStyle**[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
<b>Default</b>	0 ( default )	Windows - like
<b>CirclePicture</b>	1	circle for displaying pictures
<b>RectanglePicture</b>	2	rectangle for displaying pictures
<b>CirclePainted</b>	3	painted circle
<b>RectanglePainted</b>	4	painted rectangle

### Data Type

Integer ( enumerated )

## TimeFormat Property

### Description

Determines the time format displayed in digital clock.

### Usage

[Form][control]TimeFormat[= Integer]

### Remarks

- Read/Write
- TimeFormat: MoonPhases [5] works correct from January 18 1999.
- To display diff. Timeformats the **Digital** property should be set to **True**.
- **TimeFormat can be selected at runtime by repeated single clicks into the digital clock area !**
- **Parameters displayed in TimeFormat: SysInfo [6] can be choosed from property page.**

Constant	Value	Description
LongTime	0 ( default )	Long time ( for e. " 12:13:00" )
AMPM	1	AM/PM ( for e. "12:13 PM" in US/Engl. locale only )
Extended	2	Date ( for e. "12.12.98" )
GMT	3	GMT in format LongTime ( for e. "11:13:00 GMT " )
DaylightMap	4	a world map with approx. Daylight position is drawn
MoonPhases	5	shows the current moon phases
SysInfo	6	displays a graphical system info about: <ul style="list-style-type: none"><li>• CPU usage ( in percent )</li><li>• usage of swap file ( in percent ) *</li><li>• free RAM ( in percent )</li><li>• active threads</li></ul>

### Data Type

Integer ( enumerated )

\* Usage of swapfile is calculated according to MSInfo32. That means the usage ( in percent ) is calculated in relation to the **max. swapfilesize**. In most cases the actual size of swapfile is smaller and the usage higher. GlobalMemoryStatus returns no direct information about the **actual** swapfile size. This information can be retrieved from HKEY\_DYN\_DATA.

## TimeZone Property

### Description

Determines the current time zone for given control.

### Usage

```
[Form][control]TimeZone[= String]
```

### Remarks

- Read/Write
- If no other time zone is selected for the control - local ( from system ) time zone was returned.
- **Setting TimeZone is supported only by property page.**

### Data Type

String

## URL Property

### Description

Determines the URL for caption's hyperlink function.  
Determines the text for caption's box or bubble help.

### Usage

```
[Form][control]URL[= String]
```

### Remarks

- Read/Write
- For successful hyperlink function URL must be a valid URL ( for e. "http://www.Lycos.com" ).

### Data Type

String

## URLColor Property

### Description

Determines the color used to display the background of caption's box or bubble help.

### Usage

```
[Form][control]URLColor[= OLE_COLOR]
```

### Remarks

- Read/Write
- URLColor is set to **vbInfoBackground** by default.

### Data Type

OLE\_COLOR ( Unsigned Long )



## URLStyle Property

### Description

Determines the style of caption's URL is displayed.

### Usage

[Form][control]URLStyle[= Integer]

### Remarks

- Read/Write

Constant	Value	Description
<b>Bubble</b>	0 ( default )	a bubble help is displayed
<b>Box</b>	1	a tooltip - like window is displayed
<b>RoundRec</b>	2	a rounded rectangle is displayed

### Data Type

Integer ( enumerated )

From author:

The Box style is not the Windows - Tooltip property ! It's an own implementation so the Tooltip can be used as usual.

Please note that the Bubble style will work properly also outside control or parent form !

## WhatsThisHelpID Property

### Description

Determines an associated context number for the control.

### Usage

[Form][control]**URLStyle**[= Long]

### Remarks

- Read/Write
- For further details see VB help.

### Data Type

Long

## Alarm Event

### Description

Occurs when system time reaches AlarmTime.

### Usage

Sub ctlname\_Alarm ( ByVal AlarmTime as Date )

### Remarks

- The AlarmTime argument will return a valid date.

### Data Type

Date

## **AsyncReadComplete Event**

### **Description**

Occurs when the data that is requested by the AsyncRead method is available.

### **Usage**

```
Sub ctlname_AsyncReadComplete ( AsyncProp As AsyncProperty )
```

## **Click Event**

### **Description**

Occurs when the user presses and then releases a mouse button over an object.

## **DbClick Event**

### **Description**

Occurs when the user presses and releases a mouse button and then presses and releases it again over an object.

## Hide Event

### Description

Occurs when the control's Visible property changes to False.

## **KeyDown Event**

### **Description**

Occurs when the user presses a key while an object has the focus.



## **KeyPress Event**

### **Description**

Occurs when the user presses and releases an ANSI key.

## KeyUp Event

### Description

Occurs when the user releases a key while an object has the focus.

## MouseDown Event

### Description

Occurs when the user presses the mouse button while an object has the focus.

### Usage

```
Sub ctlname_MouseDown ( Button As Integer, Shift As Integer, X As Single, Y As Single )
```

## MouseMove Event

### Description

Occurs when the user moves the mouse.

### Usage

Sub ctlname\_MouseMove ( Button As Integer, Shift As Integer, X As Single, Y As Single )

## MouseUp Event

### Description

Occurs when the user releases the mouse button while an object has the focus.

### Usage

Sub ctlname\_MouseUp ( Button As Integer, Shift As Integer, X As Single, Y As Single )

## Resize Event

### Description

Occurs when a form is first displayed or the size of an object changes.

### Usage

```
Sub ctlname_Resize()
```

## Show Event

### Description

Occurs when the control is displayed.

### Usage

```
Sub ctlname_Show()
```

## Refresh Method

### Description

Forces a complete repaint of a object.

### Usage

Sub ctlname[Refresh]



## AsyncRead Method

### Description

Read in data asynchronously from a path or a URL.

### Usage

```
Sub ctlname [AsyncRead(Target As String, AsyncType As Long, Optional PropertyName  
As Variant)]
```

## ShowAboutBox Method

### Description

Displays information about control.

### Usage

Sub ctlname [ShowAboutBox]

## SetSysTime Method

### Description

Sets the computer's system time.

### Usage

Sub ctlname [SetSysTime ( NewTime As String ) As Long]

### Remarks

If succeeded return value is = 1

**Moving hands is allowed only when TimeZone is default !  
Setting system time by moving hands can be switched off by set the Enabled -  
property of control to FALSE.**

From author

System time ( or Internal time ) can be set by clicking and moving hand. On clicking the hand is displayed inverted.

## Registering

### Registration

You can register this **control** by sending **\$19.95** and your address. With your order, you will receive a copy of the manual documenting the Advanced Clock Control. **Control plus sourcecode** are available for **\$65.00**. A **site license** is available for **\$250.00**.

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

Prices are subject to change without notice.

For comfortable registering please run **REGISTER.EXE** from the About screen of control.

The Advanced Clock Control was written by Frank Wallwitz.

Inquiries, tech support, comments should be sent to: **100772,1205** on CompuServe, or <http://ourworld.compuserve.com/Homepages/FrankWallwitz> on Internet.

E-mail: [FrankWallwitz@compuserve.com](mailto:FrankWallwitz@compuserve.com).

You can call me at +49 89 568098 or fax at +49 89 568098 10. If you need to send something via mail, the address is:

Frank Wallwitz

Weiss - Ferdl - Str. 9  
80689 Muenchen  
Germany

© Copyright 1998 by Frank Wallwitz.

## Order form

Mail this form to :

Frank Wallwitz  
Weiss - Ferdl - Str. 9  
80689 Muenchen  
Germany

Phone: +49 89 568098

Fax: +49 89 568098

CompuServe: 100772,1205

Internet: <http://ourworld.compuserve.com/Homepages/FrankWallwitz>

E - Mail: [FrankWallwitz@compuserve.com](mailto:FrankWallwitz@compuserve.com)

Ship to: \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

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

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

E-Mail: \_\_\_\_\_

qty ordered \_\_\_\_\_

\$19.95 each.

\$65.00 incl. sourcecode

\$250.00 for site license

Total in \$: \_\_\_\_\_

## Licensing Information

### Legalese Version

Author grants a license to use the enclosed software to the original purchaser. Copies may be made for back-up purposes only. Copies made for any other purpose are expressly prohibited, and adherence to this requirement is the sole responsibility of the purchaser.

Customer written executable applications containing embedded Advanced Clock Control may be freely distributed, without royalty payments to the author, provided that such distributed author's product is bound into these applications in such a way so as to prohibit separate use in design mode, and that such author's product is distributed only in conjunction with the customers own software product. The Advanced Clock Control may not be distributed by itself in any form.

Neither source code for Advanced Clock Control nor modified source code for Advanced Clock Control may be distributed under any circumstances, nor may you distribute .OCA, .LIB, etc. files that contain my routines. This control may not be included in constituent controls or any other product that is not a stand-alone application. This license may be transferred to a third party only if all existing copies of the software and its documentation are also transferred.

This product is licensed for use by only one developer at a time. Author expressly prohibits installing this product on more than one computer if there is any chance that both copies will be used simultaneously. This restriction also extends to installation on a network server, if more than one workstation will be accessing the product. All developers working on a project which includes Advanced Clock Control, even though not working directly with the Advanced Clock Control, are required to purchase a license for Advanced Clock Control.

This software is provided as is. Author make no warranty, expressed or implied, with regard to the software. All implied warranties, including the warranties of merchantability and fitness for a particular use, are hereby excluded.

AUTHOR'S LIABILITY IS LIMITED TO THE PURCHASE PRICE. Under no circumstances shall the author of this product be liable for any incidental or consequential damages, nor for any damages in excess of the original purchase price.

To be eligible for technical support by telephone, BBS, CompuServe, etc. and to insure that you are notified of any future updates, please complete the enclosed registration card and return it to author.

You may not include my control in a constituent control (built by VB5/6 or any other similar environment.)

You may only distribute this control with a stand-alone application.

Your license is transferable. The original purchaser of the product must make the transfer request.

Contact author for further information.



