Contents

 The **ctClock** object is a visual representation of a clock control.

This control can be used to view a clock in an analog or digital format. The control continually scans the system time for the current time. The clock comes with its own styles for analog and digital clocks, or the programmer can override the style with a bitmap image.

The time can also be set using the clock. If the TimeType property is set to 1, the control will accept input from the mouse. Clicking and dragging on the control with the left mouse button will set the minute hand. Clicking and dragging on the mouse with the right button will set the hour hand.

Time Zones : There are three properties called SetHour, SetMinute, and SetSecond. These are offset properties for the current time. You can set the clock control to a new time zone by changing the SetHour property. In fact, the control could be used like a stopwatch by setting the above properties to the current time.

File Names

16 bit CTCLOK16.OCX
32 bit CTCLOK32.OCX

Class Name

ctClock

See Also :

Events
Properties

Properties

ctClock Property	Data Type	Description
AlarmHour	integer	Specifies the hour that the Alarm event will be fired.
AlarmMinute	integer	Specifies the minute that the Alarm event will be fired.
AlarmSecond	integer	Specifies the second that the Alarm event will be fired.
BackColor	long	Specifies the background color of the control.
BorderColor	long	Specifies the border color of the control.
BorderType	integer (enumerated)	Specifies the type of border to place around the control. Valid values include ... • 0 - Regular • 1 - None • 2 - Raised • 3 - Lowered
ClockColor	long	Specifies the color used to paint the face of an analog clock.
ClockType	integer (enumerated)	Specifies the type of clock to display. Valid values include ... • 0 - Digital • 1 - Analog
ClockXOffset	integer	Specifies the horizontal offset of the clock hands in an analog clock.
ClockYOffset	integer	Specifies the vertical offset of the clock hands in an analog clock.
CurrentHour	integer	Specifies the current hour for the control.
CurrentMinute	integer	Specifies the current minute for the control.
CurrentSecond	integer	Specifies the current second for the control.
Font	Font	Specifies the font used for all text within the control.
ForeColor	long	Specifies the color used to paint the text in the control
HourColor	long	Specifies the color used to paint the hour hand.
HourSize	integer	Specifies the offset value for the length of the hour hand.

IncludeAmPm	boolean	Specifies whether or not the AM/PM characters will be used in the time display.
IncludeSeconds	boolean	Specifies whether or not seconds will be included in the time display.
MilitaryTime	boolean	Specifies whether or not the control will display the time in a 24 hour military format.
MinuteColor	long	Specifies the color used to paint the minute hand.
MinuteSize	integer	Specifies the offset value for the length of the minute hand.
NumberOffset	integer	Specifies the offset for the calculated distance of the text from the edge of an analog clock.
Picture	Picture	The bitmap image used for the background of the control. If a bitmap is supplied to the control. The control will resize itself to fit the size of the bitmap.
SecondColor	long	Specifies the color used to paint the second hand.
SecondSize	integer	Specifies the offset value for the length of the second hand.
SetHour	integer	Specifies the offset value to the CurrentHour property.
SetMinute	integer	Specifies the offset value to the CurrentMinute property.
SetSecond	integer	Specifies the offset value to the CurrentSecond property.
StartTimer	boolean	Specifies whether or not the control will continue to update itself with the system clock.
TimeType	integer (enumerated)	Specifies how the clock will receive its information for the current time properties. Valid values include .. • 0 - Display current system time • 1 - Accept input from mouse

Events

ctClock	Occurs
Alarm	Occurs when the current time property values match the alarm time property values.
Click	Standard Event

DblClick	Standard Event
MouseDown	Standard Event
MouseMove	Standard Event
MouseUp	Standard Event
TimeChange	Occurs each time the current time property values changes.

AlarmHour Property

Description

Specifies the hour that the Alarm event will be fired.

Syntax

[*form.*]ctClock.**AlarmHour** [= *setting%*]

Data Type

Integer

Example

```
ctClock.AlarmHour = 12
```

AlarmMinute Property

Description

Specifies the minute that the Alarm event will be fired.

Syntax

[*form.*]ctClock.**AlarmMinute** [= *setting%*]

Data Type

Integer

Example

```
ctClock.AlarmMinute = 30
```

AlarmSecond Property

Description

Specifies the second that the Alarm event will be fired.

Syntax

[*form.*]ctClock.**AlarmSecond** [= *setting%*]

Data Type

Integer

Example

```
ctClock.AlarmSecond = 45
```

BackColor Property

Description

Determines the background color of the control. If a bitmap is used for the background, it will override this property.

Syntax

[*form.*]ctClock.**BackColor** [= *color&*]

Data Type

Long (OLE_COLOR)

Example

```
ctClock.BackColor = RGB( 128, 128, 128 )
```

BorderColor Property

Description

Specifies the color of the border painted around the control.

Syntax

[*form.*]ctClock.**BorderColor** [= *color&*]

Data Type

Long (OLE_COLOR)

Example

```
ctClock.BorderColor = RGB( 128, 128, 128 )
```

BorderType Property

Description

The type of border to place around the control. Valid values include ...

- 0 - Regular : Regular border
- 1 - None : No border
- 2 - Raised : Raised 3D border
- 3 - Lowered : Lowered 3D border

Syntax

[*form.*]ctClock.**BorderType** [= *setting%*]

Data Type

Integer (enumerated)

Example

```
ctClock.BorderType = 2
```

ClockColor Property

Description

Specifies the color used to paint the face of an analog clock. This property has no effect unless the ClockType property is set to 1.

Syntax

[*form.*]ctClock.**ClockColor** [= *color&*]

Data Type

Long (OLE_COLOR)

Example

```
ctClock.ClockColor = RGB( 128, 128, 128 )
```

ClockType Property

Description

Specifies the type of clock to display. Valid values include ...

- 0 - Digital
- 1 - Analog

Syntax

[*form.*]ctClock.**ClockType** [= *setting%*]

Data Type

Integer (enumerated)

Example

```
ctClock.ClockType = 1
```

ClockXOffset Property

Description

Specifies the horizontal offset of the clock hands in an analog clock. This property can be used to obtain a precise placement of the clock hands within the control. This property has no effect unless the ClockType property is set to 1.

Syntax

[*form.*]ctClock.**ClockXOffset** [= *setting%*]

Data Type

Integer

Example

```
ctClock.ClockXOffset = 3
```

ClockYOffset Property

Description

Specifies the vertical offset of the clock hands in an analog clock. This property can be used to obtain a precise placement of the clock hands within the control. This property has no effect unless the ClockType property is set to 1.

Syntax

[*form.*]ctClock.**ClockYOffset** [= *setting%*]

Data Type

Integer

Example

```
ctClock.ClockYOffset = -2
```

CurrentHour Property

Description

Specifies the current hour for the control. If the TimeType property is set to 0, then this property is read only.

Syntax

[*form.*]ctClock.**CurrentHour** [= *setting%*]

Data Type

Integer

Example

```
ctClock.CurrentHour = 12
```

CurrentMinute Property

Description

Specifies the current minute for the control. If the TimeType property is set to 0, then this property is read only.

Syntax

[*form.*]ctClock.**CurrentMinute** [= *setting%*]

Data Type

Integer

Example

```
ctClock.CurrentMinute = 30
```

CurrentSecond Property

Description

Specifies the current second for the control. If the TimeType property is set to 0, then this property is read only.

Syntax

[*form.*]ctClock.**CurrentSecond** [= *setting%*]

Data Type

Integer

Example

```
ctClock.CurrentSecond = 45
```

Font Property

Description

Specifies the font used for all text within the control.

In order to set the font of the control, it must be either set using the font property page of the control, the property table of the host language (if available), or by assigning a font from another control with the same font type to this one.

Syntax

[*form.*]ctClock.**Font** [= *Font*]

Data Type

Font

Example

```
ctClock.Font = lbl_Text.Font
```

ForeColor Property

Description

Specifies the color used to paint the text within the control.

Syntax

[*form.*]ctClock.**ForeColor** [= *color&*]

Data Type

Long (OLE_COLOR)

Example

```
ctClock.ForeColor = RGB( 128, 128, 128 )
```

HourColor Property

Description

Specifies the color used to the hour hand of the control. This property has no effect unless the ClockType property is set to 1.

Syntax

[*form.*]ctClock.**HourColor** [= *color&*]

Data Type

Long (OLE_COLOR)

Example

```
ctClock.HourColor = RGB( 128, 128, 128 )
```

HourSize Property

Description

Specifies the offset value for the length of the hour hand. This property has no effect unless the ClockType property is set to 1.

Syntax

[*form.*]ctClock.**HourSize** [= *setting%*]

Data Type

Integer

Example

```
ctClock.HourSize = -3
```

IncludeAmPm Property

Description

Specifies whether or not the AM/PM characters will be used in the time display. This property has no effect unless the ClockType property is set to 0.

Syntax

[*form.*]ctClock.**IncludeAmPm** [= { True | False }]

Data Type

Boolean

Example

```
ctClock.IncludeAmPm = TRUE
```

IncludeSeconds Property

Description

Specifies whether or not seconds will be included in the time display.

Syntax

[*form.*]ctClock.**IncludeSeconds** [= { True | False }]

Data Type

Boolean

Example

```
ctClock.IncludeSeconds = TRUE
```

MilitaryTime Property

Description

Specifies whether or not the control will display the time in a 24 hour military format.

Syntax

[*form.*]ctClock.**MilitaryTime** [= { True | False }]

Data Type

Boolean

Example

```
ctClock.MilitaryTime = TRUE
```

MinuteColor Property

Description

Specifies the color used to paint the minute hand of the control. This property has no effect unless the ClockType property is set to 1.

Syntax

[*form.*]ctClock.**MinuteColor** [= *color&*]

Data Type

Long (OLE_COLOR)

Example

```
ctClock.MinuteColor = RGB( 128, 128, 128 )
```

MinuteSize Property

Description

Specifies the offset value for the length of the minute hand. This property has no effect unless the ClockType property is set to 1.

Syntax

[*form.*]ctClock.**MinuteSize** [= *setting%*]

Data Type

Integer

Example

```
ctClock.MinuteSize = 5
```

NumberOffset Property

Description

Specifies the offset for the calculated distance of the text from the edge of an analog clock. It is used to move the time values closer or farther from the edge of the analog clock. This property has no effect unless the ClockType property is set to 1.

Syntax

[*form.*]ctClock.**NumberOffset** [= *setting%*]

Data Type

Integer

Example

```
ctClock.NumberOffset = 5
```

Picture Property

Description

The bitmap image used for the background of the control. If a bitmap is supplied to the control. The control will resize itself to fit the size of the bitmap.

In order to set the picture of the control, it must be either set using the picture property page of the control, the property table of the host language (if available), or by assigning a picture from another control to this one.

Syntax

[*form.*]ctClock.**Picture** [= *Picture*]

Data Type

Picture

Example

```
ctClock.Picture = OLE_Object.Picture
```

SecondColor Property

Description

Specifies the color used to paint the second hand of the control. This property has no effect unless the ClockType property is set to 1.

Syntax

[*form.*]ctClock.**SecondColor** [= *color&*]

Data Type

Long (OLE_COLOR)

Example

```
ctClock.SecondColor = RGB( 128, 128, 128 )
```

SecondSize Property

Description

Specifies the offset value for the length of the second hand. This property has no effect unless the ClockType property is set to 1.

Syntax

[*form.*]ctClock.**SecondSize** [= *setting%*]

Data Type

Integer

Example

```
ctClock.SecondSize = 5
```

SetHour Property

Description

Specifies the offset value to the CurrentHour property. Assigning a value to this property will allow you to display a time value from another time zone.

Syntax

[*form.*]ctClock.**SetHour** [= *setting%*]

Data Type

Integer

Example

```
ctClock.SetHour = 1
```

SetMinute Property

Description

Specifies the offset value to the CurrentMinute property.

Syntax

[*form.*]ctClock.**SetMinute** [= *setting%*]

Data Type

Integer

Example

```
ctClock.SetMinute = 5
```

SetSecond Property

Description

Specifies the offset value to the SetSecond property.

Syntax

[*form.*]ctClock.**SetSecond** [= *setting%*]

Data Type

Integer

Example

```
ctClock.SetSecond = 30
```

StartTimer Property

Description

Specifies whether or not the control will continue to update itself with the system clock. This property has no effect unless the TimeType property is set to 0.

Syntax

[*form.*]ctClock.**StartTimer** [= { True | False }]

Data Type

Boolean

Example

```
ctClock.StartTimer = TRUE
```

TimeType Property

Specifies how the clock will receive its information for the current time properties. Valid values include ..

- 0 - Display current system time
- 1 - Accept input from mouse

Syntax

[*form.*]ctClock.**TimeType** [= *setting%*]

Data Type

Integer (enumerated)

Example

```
ctClock.TimeType = 1
```

Alarm Event

Description

Occurs when the current time property values match the alarm time property values.

Parameters

This event has three parameters sent to it. They include ...

- 1 - nHour (integer) The value of the CurrentHour property
- 2 - nMinute (integer) The value of the CurrentMinute property
- 3 - nSecond (integer) The value of the CurrentSecond property

TimeChange Event

Description

Occurs each time the current time property values changes. This means that it will fire once each second in a clock where the TimeType property is set to 0, and it will fire each time the user move the hour or minute hand when the TimeType property is set to 1.

Parameters

This event has three parameters sent to it. They include ...

- 1 - nHour (integer) The value of the CurrentHour property
- 2 - nMinute (integer) The value of the CurrentMinute property
- 3 - nSecond (integer) The value of the CurrentSecond property