



## Help for Alarm

Alarm is a Visual Basic custom control that lets you set multiple alarms to go off (i.e., fire an event) at various times during the day. This control makes it really easy to schedule events to happen at different times during the day. Just set the properties and wait for Alarm to notify you.

Alarm even tells you when the date has changed. You can use this to remove all of the current alarms and set new ones for the day. A must if you're writing a PIM.

[Registration Information](#)

[Order Form](#)

[Getting Custom Controls Written](#)

[Properties](#)

[Events](#)

[Methods](#)

## Properties

Alarm supports the following standard properties: Name, Tag, Left, Top, and Enabled. Of these, only the enabled property is really useful. It determines whether or not Alarm sends its event to your program.

Alarm also supports the following custom properties:

AlarmTime

DateFormat

## AlarmTime Property

### Description

Determines the time(s) that the Alarm event is fired.

### Usage

```
[form.][label.]AlarmTime(AlarmIndex)[ = time$]
```

### Remarks

Setting this property sets an alarm. *AlarmIndex* determines the identifier of the alarm, which is passed to the Alarm event.

If this property is set to blank, the alarm identified is disabled.

The time specified must be in this format: HH:MM:SS Where HH, MM, and SS are either digits or question marks/wildcards '?'. The colons are required.

The Alarm control checks the time every second. If the time string set in the various alarms matches the current time (including any wildcards used).

### Examples

Let's say you want an alarm every hour on the hour. And, you want the alarm identifier to be 60. You would do the following:

```
Alarm1.AlarmTime(60) = "?:?:00:00"
```

If you wanted the control to go off every 10 minutes where the minutes ended in 5 (i.e. 5, 15, 25, 35, 45, and 55 minutes after the hour), you would use:

```
Alarm1.AlarmTime(23) = "?:?:?5:00"
```

If you wanted it to go off once per minute during the 10pm hour:

```
Alarm1.AlarmTime(0) = "22:?:?:00"
```

If you want it to go off once per second:

```
Alarm1.AlarmTime(0) = "?:?:?:??"
```

If you want it to go off every ten seconds:

```
Alarm1.AlarmTime(0) = "?:?:?:?0"
```

You can also use more than one alarm to simulate anything else you might need. Let's say you want it to go off 15 minutes after the hour, but only from 6pm to midnight. You would use:

```
Alarm1.AlarmTime(0) = "18:15:00"    ' get 6pm  
Alarm1.AlarmTime(1) = "19:15:00"    ' get 7pm  
Alarm1.AlarmTime(2) = "2?:15:00"    ' get 8pm - midnight
```

### Data Type

String

## DateFormat Property

### Description

Determines the format of the date passed by the [NewDate](#) event.

### Usage

```
[form.][[label.]]DateFormat[ = date$]
```

### Remarks

Setting this property determines the format of the date sent with the [NewDate](#) event. This can be used to override the international settings found in WIN.INI. Using this property can ensure that you always get a date formatted in a specific fashion. If you leave it to the format found in WIN.INI, you'll have to figure out what format (i.e., MM/DD/YY, DD/MM/YY, or YY/MM/DD) it is.

This string can consist of punctuation and symbols. The symbols are:

<b>Symbol</b>	<b>Meaning</b>
d	Day of the month: 1 - 31
dd	Day of the month: 01-31
m	Month of the year: 1 - 12
mm	Month of the year: 01-12
yy	Year: 00-99
yyyy	Year: 1970 - 2099

More information about this format can be found in the Windows Resource Kit under **[intl] Section**, and in the VB Programmer's Guide under **Format\$**.

### Examples

Let's say you want the date in an American format. You would do:

```
Alarm1.DateFormat = "mm/dd/yy"
```

Let's say you want the date in an American format with a four-digit year. You would do:

```
Alarm1.DateFormat = "mm/dd/yyyy"
```

Let's say you want the date in an European format. You would do:

```
Alarm1.DateFormat = "dd/mm/yy"
```

### Data Type

String

## Events

Alarm only has two events: Alarm and NewDate. The Alarm event fires when one of the alarms' AlarmTime matches the current time. NewDate fires when the date changes.

Alarm

NewDate

## **Alarm Event**

### **Description**

Occurs when it's time for one of the alarms.

### **Syntax**

Sub *ctlname*\_Alarm(*AlarmIndex* As Long, *TimeNow* As String)

### **Remarks**

The argument *AlarmIndex* represents array index given to Alarm for this time.

The argument *TimeNow* tells what time it was when this event was fired.

This event only occurs if the Enabled property is set to True.

## NewDate Event

### Description

Occurs when the date changes.

### Syntax

Sub *ctlname*\_NewDate(DateNow As String)

### Remarks

The argument *DateNow* holds the new date. This is formatted using the International date format setup by the control panel. This can be overridden using the DateFormat property.

Alarm uses the *sShortDate* entry found in the [INTL] section of WIN.INI. Depending on what the user prefers, *DateNow* can be any of (for May 5th, 1993):

06/05/1993	6/05/1993
06/5/1993	6/5/1993
05/06/93	5/06/93
05/6/93	5/12/93
1993/12/05	and more...

So, when parsing this date, use the '/' to get the three pieces (month, day, and year).

This event only occurs if the *Enabled* property is set to True.

## **Methods**

Alarm only supports the Clear method. The method removes all alarms.





## Alarm Order Form

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

Mail this James Shields  
form to: Mabry Software  
Post Office Box 31926  
Seattle, WA 98103-1926

Fax: 206-632-0272  
CompuServe: 71231,2066  
Internet: mabry@halcyon.com

MAKE CHECKS AND MONEY ORDERS OUT TO: **James Shields**

Where did you get this copy of Alarm?

\_\_\_\_\_

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

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

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

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

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

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

qty ordered \_\_\_\_\_ REGISTRATION  
\$10 each, postpaid (check or money order in hard currency).  
Foreign addressed add \$2.00 shipping. No additional shipping  
charges to Canada or Mexico.

qty ordered \_\_\_\_\_ SOURCE CODE AND REGISTRATION  
\$25 each, postpaid (check or money order in hard currency).  
Foreign addresses add \$5.00 shipping. No additional shipping  
charge to Canada.

## About Alarm

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

James Shields  
Mabry Software  
P.O. Box 31926  
Seattle, WA 98103-1926

You can register this program by sending \$10 (\$12 for international orders) and your address. CompuServe members may register by sending \$5 and their account number (the registered version will be E-mailed to you). CompuServe members may also register this package by going to the SWREG forum. Alarm is registered there. Alarms registration ID number is 1475.

Source code (which includes a registered copy) to this control is available for \$25 (\$30 for international orders). With source code you get a registered version of the control. If you are a CompuServe member, you may get the source code in the Software Registration forum (GO SWREG) for \$20. Its registration number is 1476.

To make things easier for you, you can print out and fill in the order form included with this help file.

(C) Copyright 1993 by James Shields



## **Getting Custom Controls Written**

If you or your organization would like to have custom controls written, contact either me or Zane Thomas (co-author of the Waite Group's VB How-To, 2nd Edition). Here's how we can be reached:

James Shields  
CompuServe: 71231,2066  
US Mail: P.O. Box 31926; Seattle, WA 98103-1926

Zane Thomas:  
CompuServe: 72060,3327  
US Mail: P.O. Box 300; Indianola, WA 98342

