

Forum: [Hack Forums](#)

Thread: [With an animated,faded switch button](#)

prologikus - August 8, 2014, 3:17 pm

So i wanted to create a flat switch

The components of switch to be

- the switch
- the background
- the border

default size of switch is 13,33

when it's on the switch it's located at 0, it's blue and at location 19 start the background

and when it's off the switch it's located at 13 and it's red ... from 0 to 13 it's the background ..

all the things so far was successful made ... BUT here come the hard part ...

I want to make an animation ... when the switch it's on and you press it to go smooth, slow(not too slow) down and in the same time to change its color to red slowly ..

AND i did some of that BUT every time i press the switch .. the timer runs but the paint not ... i must move my mouse fast over the switch to see differences ... i don't know how to explain that but the best way to understand it's to download the component and see what's happens

REMEMBER: the size must be 13,33 !

<http://www.fileshare.ro/e30763633>

Eternal Darkness - August 8, 2014, 8:43 pm

Not Criticizing, but here:

```
Code Enum MouseState As Byte
```

```
None = 0
```

```
Over = 1
```

```
Down = 2
```

```
Block = 3
```

```
End Enum
```

```
Class uFlatOffOn : Inherits Control
```

```
Private ToggledValue As Boolean = True
```

```
Public Property Toggled As Boolean
```

```
Get
```

```
Return ToggledValue
```

```
End Get
```

```
Set(value As Boolean)
```

```
ToggledValue = value
```

```
Invalidate() 'Refreshed the Control by Re-Drawing...
```

```
End Set
```

```
End Property
```

```
Private Const Zero As Int16 = 0, SliderSize As Double = 0.6
```

```
Private TargetRectangle As Rectangle
```

```
Private DarkBrush As SolidBrush, DarkBrush2 As SolidBrush 'Technically not necessary, but  
converts non-managed memory to Managed Memory, Aka a "safe" habit
```

```

Private State As MouseState = MouseState.None
Sub New()
    Me.BackColor = Color.Gainsboro
    State = MouseState.None
    Me.Size = New Size(13, 33)
    DarkBrush = New SolidBrush(Color.FromArgb(40, Color.Black))
    DarkBrush2 = New SolidBrush(Color.FromArgb(80, Color.Black))
End Sub
Protected Overrides Sub OnClick(ByVal e As EventArgs) 'Handles Me.Click
    MyBase.OnClick(e)
    If CType(e, System.Windows.Forms.MouseEventArgs).Button =
Windows.Forms.MouseButtons.Left Then
        If ToggledValue Then ToggledValue = False Else ToggledValue = True
        Invalidate()
    End If
End Sub
Protected Overrides Sub OnMouseEnter(ByVal e As EventArgs)
    MyBase.OnMouseEnter(e)
    State = MouseState.Over
    Invalidate()
End Sub
Protected Overrides Sub OnMouseDown(ByVal e As System.Windows.Forms.MouseEventArgs)
    MyBase.OnMouseDown(e)
    If e.Button = Windows.Forms.MouseButtons.Left Then
        If ClientRectangle.Contains(e.Location) Then State = MouseState.Down Else State =
MouseState.Over
    Else
        State = MouseState.Over
    End If
End Sub
Protected Overrides Sub OnMouseLeave(ByVal e As EventArgs)
    MyBase.OnMouseLeave(e)
    State = MouseState.None
    Invalidate()
End Sub
Protected Overrides Sub OnMouseUp(ByVal e As System.Windows.Forms.MouseEventArgs)
    MyBase.OnMouseUp(e)
    State = MouseState.Over
End Sub
Protected Overrides Sub OnPaint(ByVal e As System.Windows.Forms.PaintEventArgs)
    e.Graphics.Clear(Me.BackColor) 'or maybe Parent.BackColor?
    If ToggledValue And (Height > 0 And Width > 0) Then
        TargetRectangle = New Rectangle(Zero, Zero, Width - 1, SliderSize * Height)
        e.Graphics.FillRectangle(Brushes.DodgerBlue, TargetRectangle)
        Select Case State
        Case MouseState.None
            e.Graphics.DrawRectangle(Pens.DarkSlateBlue, TargetRectangle)
        'vs
        'e.Graphics.DrawRectangle(Pens.Gainsboro, New Rectangle(Zero, Zero, Width - 1, Height - 1))
        'e.Graphics.DrawRectangle(Pens.Gainsboro, New Rectangle(Zero, 19, Width + 1, Height + 1))
        'e.Graphics.FillRectangle(Brushes.DodgerBlue, ClientRectangle)
        'e.Graphics.FillRectangle(Brushes.Gainsboro, New Rectangle(Zero, 19, Width + 11, Height +
10))

```

```

Case MouseState.Over
e.Graphics.DrawRectangle(Pens.Navy, TargetRectangle)
e.Graphics.FillRectangle(DarkBrush, TargetRectangle)
'vs
'e.Graphics.FillRectangle(Brushes.DodgerBlue, New Rectangle(Zero, Zero, Width + 11, Height
+ 10))
'e.Graphics.FillRectangle(DarkBrush, ClientRectangle)
'e.Graphics.DrawRectangle(Pens.Gainsboro, New Rectangle(Zero, 19, Width + 1, Height + 1))
'e.Graphics.FillRectangle(Brushes.Gainsboro, New Rectangle(Zero, 19, Width + 11, Height +
10))
Case MouseState.Down
e.Graphics.DrawRectangle(Pens.MidnightBlue, TargetRectangle)
e.Graphics.FillRectangle(DarkBrush2, TargetRectangle)
'vs
'e.Graphics.FillRectangle(Brushes.DodgerBlue, New Rectangle(Zero, Zero, Width + 11, Height
+ 10))
'e.Graphics.FillRectangle(DarkBrush2, ClientRectangle)
'e.Graphics.DrawRectangle(Pens.Gainsboro, New Rectangle(Zero, 19, Width + 1, Height + 1))
'e.Graphics.FillRectangle(Brushes.Gainsboro, New Rectangle(Zero, 19, Width + 11, Height +
10))
End Select
Else 'Not Toggled
TargetRectangle = New Rectangle(Zero, (Height - SliderSize * Height) - 1, Width - 1, SliderSize
* Height)
e.Graphics.FillRectangle(Brushes.DarkRed, TargetRectangle)
Select Case State
Case MouseState.None
e.Graphics.DrawRectangle(Pens.Red, TargetRectangle)
'vs
'e.Graphics.DrawRectangle(Pens.Gainsboro, New Rectangle(Zero, Zero, Width - 1, Height - 1))
'e.Graphics.DrawRectangle(Pens.DarkRed, New Rectangle(Zero, 13, Width + 1, Height + 1))
'e.Graphics.FillRectangle(Brushes.Gainsboro, New Rectangle(Zero, Zero, Width + 1, Height +
1))
'e.Graphics.FillRectangle(Brushes.DarkRed, New Rectangle(Zero, 13, Width + 11, Height + 10))
Case MouseState.Over
e.Graphics.DrawRectangle(Pens.OrangeRed, TargetRectangle)
e.Graphics.FillRectangle(DarkBrush, TargetRectangle)
'vs
'e.Graphics.FillRectangle(Brushes.Gainsboro, New Rectangle(Zero, Zero, Width + 1, Height +
1))
'e.Graphics.FillRectangle(Brushes.DarkRed, New Rectangle(Zero, 13, Width + 11, Height + 10))
'e.Graphics.FillRectangle(DarkBrush, New Rectangle(Zero, 13, Width, Height))
Case MouseState.Down
e.Graphics.DrawRectangle(Pens.Red, TargetRectangle)
e.Graphics.FillRectangle(DarkBrush2, TargetRectangle)
'vs
'e.Graphics.FillRectangle(Brushes.Gainsboro, New Rectangle(Zero, Zero, Width + 1, Height +
1))
'e.Graphics.FillRectangle(Brushes.DarkRed, New Rectangle(Zero, 13, Width + 11, Height + 10))
'e.Graphics.FillRectangle(DarkBrush2, New Rectangle(Zero, 13, Width, Height))
End Select
End If
End Sub

```

End ClassWith this, you can remove all the Themebase Stuff, all those imports, etc... Changed it up a little, but if you were to comment out my drawing stuff it should preform almost exactly as yours did (Color wise)... I did not add the transition, but I would prefer it to transition automatically, not slowly moving up (personally).

I tired to show you how much extra drawing you are doing... It seems kinda unnecessary... Again not criticizing, just pointing it out and posing a possible solution as to how to optimize it (some)...

This does not require that it is 13 by 33, but will start out at that size, since it sounds like you like that size... My version of this scales properly however, to "any" size...

Feel free to grab things from this code for your control, if you wish... Don't worry about saying that I helped, if you do, idrc... Just posting what I believe to be simpler, and cooler looking...

Ideas:

Could Draw Text on the control if the "slider"/"switch" was a fixed/property Width, instead of the controls width...

Could Transition the Color as you transition the control from the bottom to the top, if you are going to add transitioning back in (An Example of How to transition between colors is in the source here: <https://hackforums.net/showthread.php?tid=4369039> , except you wish to take the syncBlendInto, and Blend Methods)... Just an interesting idea that came to mind... :)

Also whats the point of sleeping for 0 seconds (While transitioning)? You will end up lagging the form a bit in comparison to not calling the sleep method...

Nice work OP, this is in no way the worst user control. That's why I am trying to optimize it (a bit) before people use it... xD

prologikus - August 9, 2014, 2:30 am

First i want to thanks you for all you have been posted.

This was my first button/switch created with graphics so i didn't know so much about it ... i usually use what i know and what i have to reach what i want instead of search for another methods but i always learn when i found somethings that i like ...

then ... i want to show u an image with what i wanted to reach

this is made by me too so .. u don't wanna see what's behind :D lot of code ... but this is not made with graphics and because of that i got some bugs when trying to active 2 buttons in the same time and another bug is i got that border in the middle of the switch ... (with boder size 0)

Your code was really small besides mine and u did it very simple and objective but you miss some of the design :D

just saying ... :-"

Anyway thanks for the code i learned lot of stuff from it and with some edit i can make it like i wanted :D and i will leave that animation ... after i realize it's hard to make it.