

Streamlining your code

Chapter 8 - Further reduce the amount of GUI controls (Using hot-spots)

[StPendl](#) Oct 29, 2011

We have now tasted the sweetness of removing controls, so we further reduce the amount of controls by replacing the individual color selection boxes with one for each band.

For this we just have to base our color selection on the position of the pick inside the graphics box:

```
' create color fields
FOR Band = 1 to 4
    Handle$ = "#w.Band"; Band

    #Handle$ "cls;down;fill "; BackgroundColor$
    #Handle$ "when leftButtonUp MouseClick"

    FOR Color = 1 to 10
        if Band = 4 then
            if Color > 3 then exit for

            #Handle$ "backcolor "; Tolerances$(Color,2)
        else
            #Handle$ "backcolor "; Colors$(Color,2)
        end if

        #Handle$ "place 0 "; 40*(Color-1);
";boxfilled 30 "; 40*(Color-1)+30
    NEXT

    #Handle$ "flush"

    ' apply initial settings
    if Band = 4 then
        call MouseClick Handle$, 0, 95
    else
        call MouseClick Handle$, 0, 15
    end if
NEXT

[none]
WAIT
```

```
[quit]
  close #w
  END

SUB MouseClick Handle$, PosX, PosY
  ' parse handle of calling control
  Band = val(Right$(Handle$, 1))

  ' get position of color
  ColorIndex = int(PosY/40)+1

  Select Case Band
    Case 1
      Band1value = (ColorIndex-1) * 10
      ColorValue1$ = Colors$(ColorIndex,2)

    Case 2
      Band2value = ColorIndex-1
      ColorValue2$ = Colors$(ColorIndex,2)

    Case 3
      Band3value = 10 ^ (ColorIndex-1)
      ColorValue3$ = Colors$(ColorIndex,2)

    Case 4
      Band4value = ColorIndex*5
      if Band4value = 15 then Band4value = 20
      ColorValue4$ = Tolerances$(ColorIndex,2)

    Case Else
      Exit Sub
  End Select
```

The complete code now looks like this.

```
'Resistor Calculator.bas
'Author: salaion Yahoo! Group
'Date: 27.11.10
'
' List of Colors at http://en.wikipedia.org/wiki/List\_of\_colors
'
' 27.11.10 00:18:50 - Initial as posted

' 27.11.10 00:21:40 - added array for colors and reduced redundant cod
```

e for color boxes

```
' 27.11.10 00:34:07 - replaced duplicate command strings by variables
' 27.11.10 02:22:16 - consolidated event handler
'
' always take the time for some sleep or you will make things worse ;-
)
'
' 27.11.10 10:56:24 - removed radio buttons
' 27.11.10 11:53:34 - used single graphics box for resistor
' 27.11.10 13:09:19 - now using the bare minimum of GUI controls

nomainwin
WindowWidth = 820:WindowHeight = 600
UpperLeftX=int( (DisplayWidth- WindowWidth) /2)
UpperLeftY=int( (DisplayHeight- WindowHeight) /2)
BackgroundColor$ = "green"

' define variables to be remembered between calling the sub
GLOBAL Band1value, Band2value, Band3value, Band4value
GLOBAL ColorValue1$, ColorValue2$, ColorValue3$, ColorValue4$

' set initial band color to body color
ColorValue1$ = "21 228 255"
ColorValue2$ = ColorValue1$
ColorValue3$ = ColorValue1$
ColorValue4$ = ColorValue1$

dim Colors$(10,2), Tolerances$(3,2)

[Colors]
DATA "Black",    "  0  0  0"
DATA "Brown",   "170 100 20"
DATA "Red",     "255  0  0"
DATA "Orange",  "255 180  0"
DATA "Yellow",  "255 255  0"
DATA "Green",   " 45 157 40"
DATA "Blue",    "  0  0 255"
DATA "Violet",  "170  0 180"
DATA "Gray",    "127 127 127"
DATA "White",   "255 255 255"

[Tolerances]
DATA "Gold",    "212 175  55"
DATA "Silver",  "200 200 250"
```

```
DATA "None", " 21 228 255"

'Fill the arrays
RESTORE [Colors]

FOR i = 1 to 10
    READ ColorName$, ColorValue$
    Colors$(i,1) = ColorName$
    Colors$(i,2) = ColorValue$
NEXT

RESTORE [Tolerances]

FOR i = 1 to 3
    READ ColorName$, ColorValue$
    Tolerances$(i,1) = ColorName$
    Tolerances$(i,2) = ColorValue$
NEXT

STATICTEXT #w.0hm, "" , 500, 200, 300, 20
STATICTEXT #w.Tole,"" , 500, 230, 300, 20

STATICTEXT #w,Colors$( 1,1), 10, 42, 50, 20
STATICTEXT #w,Colors$( 2,1), 10, 82, 50, 20
STATICTEXT #w,Colors$( 3,1), 10, 122, 50, 20
STATICTEXT #w,Colors$( 4,1), 10, 162, 50, 20
STATICTEXT #w,Colors$( 5,1), 10, 202, 50, 20
STATICTEXT #w,Colors$( 6,1), 10, 242, 50, 20
STATICTEXT #w,Colors$( 7,1), 10, 282, 50, 20
STATICTEXT #w,Colors$( 8,1), 10, 322, 50, 20
STATICTEXT #w,Colors$( 9,1), 10, 362, 50, 20
STATICTEXT #w,Colors$(10,1), 10, 402, 50, 20

STATICTEXT #w,Tolerances$(1,1), 420, 42, 50, 20
STATICTEXT #w,Tolerances$(2,1), 420, 82, 50, 20
STATICTEXT #w,Tolerances$(3,1), 420, 122, 50, 20

STATICTEXT #w,"K=Kilo " , 470, 300,100, 20
STATICTEXT #w,"M=Mega " , 470, 330,100, 20
STATICTEXT #w,
"Light Blue is The Default Color of Resistance " , 470, 360,400, 20

'Band1
STATICTEXT #w, "First Digit", 70, 7, 80, 20
groupbox #w, "", 70, 20, 70,420
graphicbox #w.Band1 , 90, 40, 30,390
```

```
stylebits #w.Band1, 0,_WS_BORDER, 0,0

'Band2
STATICTEXT #w, "Second Digit", 150, 7, 90, 20
groupbox #w, "", 160, 20, 70,420
graphicbox #w.Band2 , 180, 40, 30,390
stylebits #w.Band2, 0,_WS_BORDER, 0,0

'Band3
STATICTEXT #w, "Multiplier", 250, 7, 80, 20
groupbox #w, "", 250, 20, 70,420
graphicbox #w.Band3 , 270, 40, 30,390
stylebits #w.Band3, 0,_WS_BORDER, 0,0

'Band4
STATICTEXT #w, "Tolerance", 340, 7, 80, 20
groupbox #w, "", 340, 20, 70,420
graphicbox #w.Band4 , 360, 40, 30,110
stylebits #w.Band4, 0,_WS_BORDER, 0,0

'Fram
graphicbox #w.title, 488,53,306,44
stylebits #w.title, 0,_WS_BORDER, 0,0

'General Body Resistance
graphicbox #w.resistor, 80,470,270,35
stylebits #w.resistor, 0,_WS_BORDER,0,0

open "Resistor Calculator" for window_nf as #w
#w "trapclose [quit]"
#w "font Times_New_Roman 13"

#w.title
"cls;down;color blue;backcolor yellow;size 15;place 0 0;boxfilled 306
44"
#w.title "font Times_New_Roman 16 bold;color red"
#w.title "place 13 28;\Resistor Color Code of 4 Bands"
#w.title "flush"

' create color fields
FOR Band = 1 to 4
    Handle$ = "#w.Band"; Band

    #Handle$ "cls;down;fill "; BackgroundColor$
    #Handle$ "when leftButtonUp MouseClick"
```

```
FOR Color = 1 to 10
  if Band = 4 then
    if Color > 3 then exit for

    #Handle$ "backcolor "; Tolerances$(Color,2)
  else
    #Handle$ "backcolor "; Colors$(Color,2)
  end if

  #Handle$ "place 0 "; 40*(Color-1);
";boxfilled 30 "; 40*(Color-1)+30
NEXT

#Handle$ "flush"

' apply initial settings
if Band = 4 then
  call MouseClick Handle$, 0, 95
else
  call MouseClick Handle$, 0, 15
end if
NEXT

[none]
WAIT

[quit]
close #w
END

SUB MouseClick Handle$, PosX, PosY
' parse handle of calling control
Band = val(Right$(Handle$, 1))

' get position of color
ColorIndex = int(PosY/40)+1

Select Case Band
  Case 1
    Band1value = (ColorIndex-1) * 10
    ColorValue1$ = Colors$(ColorIndex,2)

  Case 2
    Band2value = ColorIndex-1
    ColorValue2$ = Colors$(ColorIndex,2)
```

```
Case 3
Band3value = 10 ^ (ColorIndex-1)
ColorValue3$ = Colors$(ColorIndex,2)

Case 4
Band4value = ColorIndex*5
if Band4value = 15 then Band4value = 20
ColorValue4$ = Tolerances$(ColorIndex,2)

Case Else
Exit Sub
End Select

'Draw Resistor
#w.resistor
"cls;down;home;posxy CenterX CenterY;fill "; BackgroundColor$
BoxWidth = CenterX*2
BoxHeight = CenterY*2

'Draw Wire
#w.resistor "color 127 127 127;backcolor 127 127 127"
#w.resistor "place 0 "; CenterY-2; ";boxfilled "
; BoxWidth; " "; CenterY+2

'Draw Body
#w.resistor "color 21 228 255;backcolor 21 228 255"
#w.resistor "place 50 0;boxfilled "; BoxWidth-50; " "; BoxHeight

'Draw Bands
#w.resistor "color "; ColorValue1$; ";backcolor "; ColorValue1$
#w.resistor "place 70 0;boxfilled 85 "; BoxHeight

#w.resistor "color "; ColorValue2$; ";backcolor "; ColorValue2$
#w.resistor "place 100 0;boxfilled 115 "; BoxHeight

#w.resistor "color "; ColorValue3$; ";backcolor "; ColorValue3$
#w.resistor "place 130 0;boxfilled 145 "; BoxHeight

#w.resistor "color "; ColorValue4$; ";backcolor "; ColorValue4$
#w.resistor "place 160 0;boxfilled 175 "; BoxHeight

#w.resistor "flush"

vv=(Band1value+Band2value)*Band3value

Select Case
```

```
Case vv>=1000000
#w.Ohm "Resistance Value=";vv/1000000;" M Ohm"
```

```
Case vv>=1000
#w.Ohm "Resistance Value=";vv/1000;" K Ohm"
```

```
Case Else
#w.Ohm "Resistance Value=";vv;" Ohm"
End Select
```

```
if Band4value>0 then #w.Tole
"Tolerance Value =";" +/- ";Band4value;"%"
END SUB
```

[Chapter 1 - Starting the mission](#)

[Chapter 2 - Using arrays to reduce redundant code](#)

[Chapter 3 - Use variables for duplicate command strings](#)

[Chapter 4 - Consolidate event handlers](#)

[Chapter 5 - Apply mouse selection](#)

[Chapter 6 - Remove radio buttons](#)

[Chapter 7 - Use a single graphics box for the resistor display](#)

Chapter 8 - Further reduce the amount of GUI controls (Using hot-spots)

[Chapter 9 - Adding some eye-candy](#)

[Chapter 10 - Summary](#)