1. Switching between Valve and Safing Control Modes -

```
Private Sub DisplayChange_Click(sender As Object, e As EventArgs)
Handles DisplayChange.Click
    If DisplayMode.Equals("Information") Then
        For Each cntrl As Control In Me.Controls
            If TypeOf cntrl Is Button Then
                If cntrl.Name.Contains("Valve") Then
                    cntrl.Enabled = False
                End If
                If cntrl.Name.Contains("Module") Then
                    cntrl.Enabled = True
                End If
            End If
            If TypeOf cntrl Is Label Then
                If (Not cntrl.Name.Contains("Safing")) Then
                    cntrl.Visible = False
                ElseIf cntrl.Name.Contains("Safing") Then
                    cntrl.Visible = True
                End If
            End If
        Next
        ControlsLabel.Visible = True
        AirLeakToggle.Visible = True
        DisplayMode = "Safing"
        DisplayChange.Text = "Valve Mode"
        ToggleButton.Text = "Fire"
    Else
        For Each cntrl As Control In Me.Controls
            If TypeOf cntrl Is Button Then
                If cntrl.Name.Contains("Module") Then
                    cntrl.Enabled = False
                End If
                If cntrl.Name.Contains("Valve") Then
                    cntrl.Enabled = True
                End If
            End If
            If TypeOf cntrl Is Label Then
                If (Not cntrl.Name.Contains("Safing")) Then
                    cntrl.Visible = True
                ElseIf cntrl.Name.Contains("Safing") Then
                    cntrl.Visible = False
                End If
            End If
        Next
```

```
        AirLeakToggle.Visible = False
        DisplayMode = "Information"
        DisplayChange.Text = "Safing Mode"
        ToggleButton.Text = "Toggle"
    End If
End Sub
```

- Activating and deactivating an air leak -

```
Private Sub AirLeakToggle_Click(sender As Object, e As EventArgs)
Handles AirLeakToggle.Click
    If AirLeak.Equals("Undetected") Then
        AirLeak = "Detected"
        For Each cntrl As Control In Me.Controls
            If TypeOf cntrl Is Button Then
                If cntrl.Name.Contains("Module") Then
                    CallByName(cntrl, "BackColor", CallType.Set,
Color.FromArgb(199, 33, 56))
                End If
            End If
        Next
        For Each cntrl As Control In Me.Controls
            If TypeOf cntrl Is Button Then
                If cntrl.Name.Contains("Valve") Then
                    ReferenceObject = cntrl
                    CallByName(ReferenceObject.FlatAppearance,
"BorderColor", CallType.Set, Color.FromArgb(215, 166, 75))
                End If
            End If
        Next
        DisplayInfo(ReferenceObject, False, True)
        Refresh()
        Threading.Thread.Sleep(3000)
        For Each cntrl As Control In Me.Controls
            If TypeOf cntrl Is Button Then
                If cntrl.Name.Contains("Valve") Then
                    ReferenceObject = cntrl
                    CallByName(ReferenceObject.FlatAppearance,
"BorderColor", CallType.Set, Color.FromArgb(199, 33, 56))
                    ReferenceObject.ImageAlign = 64
                End If
            End If
        Next
        Refresh()
    Else
        AirLeak = "Undetected"
        For Each cntrl As Control In Me.Controls
            If TypeOf cntrl Is Button Then
                If cntrl.Name.Contains("Module") Then
```

```
                    CallByName(cntrl, "BackColor", CallType.Set,
Color.White)
                End If
            End If
        Next
        For Each cntrl As Control In Me.Controls
            If TypeOf cntrl Is Button Then
                If cntrl.Name.Contains("Valve") Then
                    ReferenceObject = cntrl
                    CallByName(ReferenceObject.FlatAppearance,
"BorderColor", CallType.Set, Color.FromArgb(215, 166, 75))
                End If
            End If
        Next
        DisplayInfo(ReferenceObject, False, True)
        Refresh()
        Threading.Thread.Sleep(3000)
        For Each cntrl As Control In Me.Controls
            If TypeOf cntrl Is Button Then
                If cntrl.Name.Contains("Valve") Then
                    ReferenceObject = cntrl
                    CallByName(ReferenceObject.FlatAppearance,
"BorderColor", CallType.Set, Color.FromArgb(48, 76, 122))
                    ReferenceObject.ImageAlign = 16
                End If
            End If
        Next
        Refresh()
    End If
End Sub
```

- Automatically closing valves around a module with a fire -

```
Private Sub FireCheck(Reference As Object)
    FireCheckObject = ReferenceObject
    For Each cntrl As Control In Me.Controls
        If TypeOf cntrl Is Button Then
            If cntrl.Name.Contains("Valve") Then
                FireCheckString = CStr(CallByName(FireCheckObject,
"Name", CallType.Get))
                FireCheckString = FireCheckString.Substring(0,
FireCheckString.Length() - 6)
                If cntrl.Name.Contains(FireCheckString) Then
                    ReferenceObject = cntrl
                    ToggleButton.PerformClick()
                End If
            End If
        End If
    Next
```

```
End Sub
```

- Toggling a Valve / Safing Mode command -

```
Private Sub ToggleButton_Click(sender As Object, e As EventArgs)
Handles ToggleButton.Click
    If ReferenceObject.Name.Contains("Valve") Then
        StatusLabel.Text = "Transitioning"
        CallByName(ReferenceObject.FlatAppearance, "BorderColor",
CallType.Set, Color.FromArgb(215, 166, 75))
        Refresh()
        If ReferenceObject.ImageAlign = 16 Then
            Threading.Thread.Sleep(3000)
            DisplayInfo(ReferenceObject, False, True)
            Refresh()
            CallByName(ReferenceObject.FlatAppearance, "BorderColor",
CallType.Set, Color.FromArgb(199, 33, 56))
            ReferenceObject.ImageAlign = 64
        Else
            Threading.Thread.Sleep(3000)
            DisplayInfo(ReferenceObject, False, True)
            Refresh()
            CallByName(ReferenceObject.FlatAppearance, "BorderColor",
CallType.Set, Color.FromArgb(48, 76, 122))
            ReferenceObject.ImageAlign = 16
        End If
    Else
        If ReferenceObject.ImageAlign = 16 Then
            DisplayInfo(ReferenceObject, True, True)
            Refresh()
            CallByName(ReferenceObject.FlatAppearance, "BorderColor",
CallType.Set, Color.FromArgb(199, 33, 56))
            ReferenceObject.ImageAlign = 64
            Call FireCheck(ReferenceObject)
        Else
            DisplayInfo(ReferenceObject, True, True)
            Refresh()
            CallByName(ReferenceObject.FlatAppearance, "BorderColor",
CallType.Set, Color.FromArgb(174, 176, 181))
            ReferenceObject.ImageAlign = 16
            Call FireCheck(ReferenceObject)
        End If
    End If
End Sub
```

- Displaying information within Valve / Safing Modes -

```
Private Sub DisplayInfo(DisplayObj As Object, ModuleCheck As Boolean,
UnselectCheck As Boolean)
    If (ModuleCheck = False And UnselectCheck = False) Then
```

```
        ValveLabel.Text = DisplayObj.Name
        If DisplayObj.ImageAlign = 16 Then
            FanSpeed = 1000
            StatusLabel.Text = "Open"
        Else
            FanSpeed = 0
            StatusLabel.Text = "Closed"
        End If
        FanSpeedLabel.Text = CStr(FanSpeed) + " RPM"
    ElseIf (ModuleCheck = False And UnselectCheck = True) Then
        ValveLabel.Text = ""
        StatusLabel.Text = ""
        FanSpeedLabel.Text = ""
    ElseIf (ModuleCheck = True And UnselectCheck = False) Then
        ModuleSafingLabel.Text = DisplayObj.Name
        If DisplayObj.ImageAlign = 16 Then
            FireSensorSafingLabel.Text = "Undetected"
        Else
            FireSensorSafingLabel.Text = "Detected"
        End If
        AirLeakSafingLabel.Text = AirLeak
    Else
        ModuleSafingLabel.Text = ""
        FireSensorSafingLabel.Text = ""
        AirLeakSafingLabel.Text = ""
    End If
End Sub
```