

MainForm - 1

Option Explicit

Dim DrawerHandle As Long
Dim DrawerNumber As Byte

'DLL functions used to communicate with the USB device
Private Declare Function OpenDrawer Lib "MSPOS_USB.dll" (ByVal DeviceHandle As Long) As Integer
Private Declare Function GetDrawerStatus Lib "MSPOS_USB.dll" (ByVal DeviceHandle As Long) As Integer
Private Declare Function ReleaseDrawerHandle Lib "MSPOS_USB.dll" (ByVal DeviceHandle As Long) As Integer
Private Declare Function GetDrawerHandle Lib "MSPOS_USB.dll" (ByVal DrawerNumber As Byte) As Long

Private Sub DrawerOption_Click(Index As Integer)
 DrawerNumber = Index 'Get the new drawer ID number
 If DrawerHandle > 0 Then 'Check to see if we already have a valid handle
 Call ReleaseDrawerHandle(DrawerHandle) 'If so release the handle call "ReleaseDrawerHandle"
 DrawerHandle = 0 'Reset the handle to 0
 End If
 Call Status_Button_Click 'Call the "Get Status" button routine
End Sub

Private Sub Form_Load()
 'Set default properties
 Status.Caption = "USB Device NOT Connected!"
 Status.ForeColor = &HFF
 Shapel.BorderColor = &H0
 Shapel.FillColor = &H0
 Label2.Caption = "OFFLINE"
 Open_Button.Enabled = False
 Status_Button.Enabled = False
 mnuCashDrawer.Enabled = False
 DrawerHandle = 0
End Sub

Private Sub Form_Terminate()
 'If we have a valid handle then release it
 If DrawerHandle > 0 Then Call ReleaseDrawerHandle(DrawerHandle)
 End
End Sub

Private Sub Form_Unload(Cancel As Integer)
 'If we have a valid handle then release it
 If DrawerHandle > 0 Then Call ReleaseDrawerHandle(DrawerHandle)
 End
End Sub

Private Sub mnuAbout_Click()
 AboutForm.Show 1
End Sub

Private Sub mnuExit_Click()
 'If we have a valid handle then release it
 If DrawerHandle > 0 Then Call ReleaseDrawerHandle(DrawerHandle)
 End
End Sub

Private Sub mnuOnline_Click()
 Call Status_Button_Click
End Sub

Private Sub mnuOpenDrawer_Click()
 Call Open_Button_Click
End Sub

Private Sub mnuStatusDrawer_Click()

MainForm - 2

```
    Call Status_Button_Click
End Sub
```

```
Private Sub Open_Button_Click()
    Dim Result As Integer
```

```
    Call GetHandle                                'Get a handle for the device if we do not have one
    If DrawerHandle = 0 Then Exit Sub              'If not valid then exit
    Result = OpenDrawer(DrawerHandle)              'Else call "OpenDrawer" function
    'Catch errors on the USB Device
    If Result = 0 Then                              'If it returned a 0 we had an error
        Call USBError
    End If
End Sub
```

```
Private Sub USBError()
```

```
    'This gets called when there was a reported error getting or sending data
    'from the USB bus and the device. This sub will set the form properties to
    'not allow user interaction if the device communication failed.
    Status.Caption = "USB Device NOT Connected!"
    Status.ForeColor = &HFF
    Shape1.BorderColor = &H0
    Shape1.FillColor = &H0
    Label2.Caption = "OFFLINE"
    Open_Button.Enabled = False
    Status_Button.Enabled = False
    mnuCashDrawer.Enabled = False
    Call ReleaseDrawerHandle(DrawerHandle)
    DrawerHandle = 0
End Sub
```

```
Private Sub Status_Button_Click()
    Dim Result As Integer
```

```
    Call GetHandle                                'Get a handel for teh device if we do not have one
    If DrawerHandle = 0 Then Exit Sub              'If not valid then exit
    Result = GetDrawerStatus(DrawerHandle)        'Else call "GetDrawerStatus" function

    If Result = 0 Then                              'If it returned a 0 we had an error
        Call USBError
    ElseIf Result = 2 Then                          'If it returned a 2 we have an open cash drawer
        Label2.Caption = "OPEN"
        Shape1.FillColor = &HFF00&
        Shape1.BorderColor = &HFF00&
    Else                                            'If it returned a 1 we have a closed drawer
        Label2.Caption = "CLOSED"
        Shape1.FillColor = &HFF
        Shape1.BorderColor = &HFF
    End If
End Sub
```

```
Private Sub Timer1_Timer()
    Dim Result As Integer
```

```
    Timer1.Enabled = False                        'Timer to call the status button every 100 mS
    Call Status_Button_Click
    Timer1.Enabled = True
End Sub
```

```
Private Sub GetHandle()
```

```
    If DrawerHandle > 0 Then Exit Sub              'If we have a handle already then exit
                                                'Call the "GetDrawerHandle" function
    DrawerHandle = GetDrawerHandle(DrawerNumber)
    If DrawerHandle > 0 Then                        'If it retured a valid handle set the form
        Status.Caption = "USB Device Connected"
        Status.ForeColor = &HFF00&
        Open_Button.Enabled = True
        Status_Button.Enabled = True
    End If
End Sub
```

MainForm - 3

```
        mnuCashDrawer.Enabled = True
        Call Status_Button_Click
    Else
        Call USBError                'Else we had an error
    End If
End Sub
```