```
UserForm1 - 1

Private Sub Frame2_Click()

End Sub

Private Sub TextBox10_Change()

End Sub

Private Sub TextBox14_Change()

End Sub

Private Sub TextBox17_Change()

End Sub

Private Sub TextBox2_Change()

End Sub

Private Sub TextBox20_Change()

End Sub

Private Sub TextBox21_Change()

End Sub

Private Sub TextBox22_Change()

End Sub

Private Sub TextBox23_Change()

End Sub

Private Sub TextBox24_Change()

End Sub

Private Sub TextBox25_Change()

End Sub

Private Sub TextBox26_Change()

End Sub

Private Sub TextBox27_Change()

End Sub

Private Sub TextBox28_Change()

End Sub

Private Sub TextBox29_Change()

End Sub

Private Sub TextBox3_Change()

End Sub

Private Sub TextBox30_Change()

End Sub

Private Sub TextBox31_Change()

End Sub

Private Sub TextBox32_Change()
```

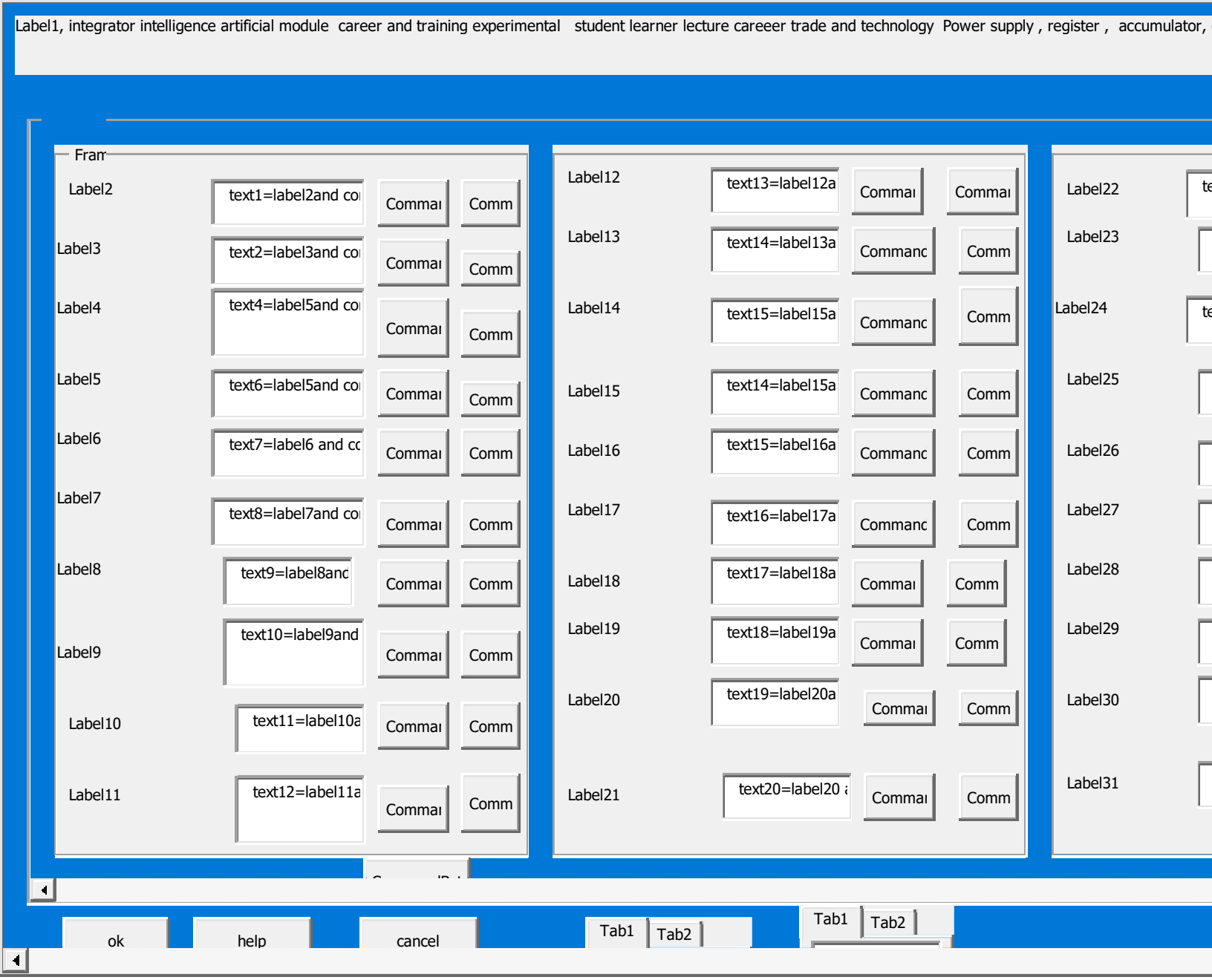
```
UserForm1 - 2

End Sub

Private Sub TextBox33_Change()

End Sub

Private Sub TextBox34_Change()

End Sub

Private Sub TextBox35_Change()

End Sub

Private Sub TextBox36_Change()

End Sub

Private Sub TextBox37_Change()

End Sub

Private Sub TextBox38_Change()

End Sub

Private Sub TextBox4_Change()

End Sub

Private Sub TextBox40_Change()

End Sub

Private Sub TextBox5_Change()

End Sub

Private Sub TextBox6_Change()

End Sub

Private Sub TextBox7_Change()

End Sub

Private Sub TextBox8_Change()

End Sub

Private Sub TextBox9_AfterUpdate()

End Sub

Private Sub TextBox9_Change()

End Sub

Private Sub UserForm_Click()

End Sub

Private Sub UserForm_DblClick(ByVal Cancel As MSForms.ReturnBoolean)

End Sub

Private Sub UserForm_Deactivate()

End Sub

Private Sub UserForm_Initialize()
```

```
UserForm1 - 3

End Sub

Private Sub UserForm_KeyDown(ByVal KeyCode As MSForms.ReturnInteger, ByVal Shift As Integer)

End Sub

Private Sub UserForm_KeyUp(ByVal KeyCode As MSForms.ReturnInteger, ByVal Shift As Integer)

End Sub

Private Sub UserForm_Layout()

End Sub

Private Sub UserForm_MouseDown(ByVal Button As Integer, ByVal Shift As Integer, ByVal x As Single, ByV
al Y As Single)

End Sub

Private Sub UserForm_MouseMove(ByVal Button As Integer, ByVal Shift As Integer, ByVal x As Single, ByV
al Y As Single)

End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)

End Sub

Private Sub UserForm_RemoveControl(ByVal Control As MSForms.Control)

End Sub

Private Sub UserForm_Resize()

End Sub



End Sub
Control Register for SCADA Switch
Dim ControlRegister As Integer
Dim PortStatus As Boolean

Sub ReadPort()
    ControlRegister = &H1A ' Example register address
    PortStatus = (ControlRegister And &H1) = &H1
    If PortStatus Then
        MsgBox "Port Active"
    Else
        MsgBox "Port Inactive"
    End If
End Sub
' LED connected to amplifier logic
Sub ControlLED(ByVal AmpLevel As Integer)
    If AmpLevel > 5 Then
        LEDPin = True
    Else
        LEDPin = False
    End If
End Sub
' Relay logic for fault detection
Dim FaultA, FaultB, FaultC As Boolean

Sub CheckFaults()
    If FaultA Or FaultB Or FaultC Then
        MsgBox "Fault Detected"
        ActivateBreaker()
    End If
End Sub

Sub ActivateBreaker()
    ' Simulate breaker trip
    BreakerStatus = "Tripped"
```

```
UserForm1 - 4

End Sub
[Start]
   Print
[Read Sensor Data]
   Print
[Check Threshold]
   ???(Yes)??> [Activate Output Pin 7]
   ???(No)???> [Log Data]
   Print
End
[Initialize System]
   Print
[Monitor Current Z]
   Print
[Detect Fault A/B/C]
   ???(Fault A)??> [Trip Breaker A]
   ???(Fault B)??> [Trip Breaker B]
   ???(Fault C)??> [Trip Breaker C]
   Print
[Log Fault Event]
   Print
End
Dim ControlRegister As Byte
Dim PortInput As Boolean

Sub ReadControlPort()
    ControlRegister = &H1A ' Example address
    PortInput = (ControlRegister And &H1) = &H1
    If PortInput Then
        MsgBox "Port Active"
    Else
        MsgBox "Port Inactive"
    End If
End Sub
?? 1B: SCADA Switch Control
Dim SCADASwitch As Boolean

Sub ToggleSCADASwitch()
    SCADASwitch = Not SCADASwitch
    If SCADASwitch Then
        MsgBox "SCADA Switch ON"
    Else
        MsgBox "SCADA Switch OFF"
    End If
End Sub
?? Visual Basic Processor & Relay Logic (Core Code)
?? 1A: Control Register - Lecture Port
Dim ControlRegister As Byte
Dim PortInput As Boolean


    ControlRegister = &H1A ' Example address
    PortInput = (ControlRegister And &H1) = &H1
    If PortInput Then
        MsgBox "Port Active"
    Else
        MsgBox "Port Inactive"
    End If
End Sub
?? 1B: SCADA Switch Control
Dim SCADASwitch As Boolean


    SCADASwitch = Not SCADASwitch
    If SCADASwitch Then
        MsgBox "SCADA Switch ON"
    Else
        MsgBox "SCADA Switch OFF"
    End If
End Sub
?? LED Control - Connect to Amp
vbnet
Dim LEDState As Boolean
```

```
UserForm1 - 5

Dim AmpLevel As Integer


    If AmpLevel > 5 Then
        LEDState = True
        MsgBox "LED ON"
    Else
        LEDState = False
        MsgBox "LED OFF"
    End If
End Sub
?? Output Switch - Pin 7 Logic
vbnet
Dim OutputPin7 As Boolean

Sub SwitchOffPin7()
    OutputPin7 = False
    MsgBox "Pin 7 Output OFF"
End Sub
?? Relay Current Z Logic (Z = rg + jxd)
vbnet
Structure Impedance
    Dim rg As Double
    Dim jxd As Double
End Structure


    Return New Complex(r, x)
End Function
?? Logigramme: Fault Detection & Relay Trigger
plaintext
[Start]
   Print
[Read Current Z]
   Print
[Compare Threshold]
   ???(Above Limit)??> [Trigger Relay]
   ???(Normal)???????> [Continue Monitoring]
   Print
[Log Event]
   Print
End
?? Algorigramme: SCADA Switch + LED + Output Pin
plaintext
[Initialize System]
   Print
[Check SCADA Switch]
   Print
[If ON]
   ???> [Read Amp Level]
   ???> [If Amp > 5 ? LED ON]
   ???> [Else ? LED OFF]
   Print
[Switch OFF Pin 7]
   Print
End
?? Integration with Curriculum Text Boxes
Text Box    Functionality   Visual Basic Logic
1A  Control Register    ReadControlPort()
1B  SCADA Switch    ToggleSCADASwitch()
LED/Amp LED Control ControlLED()
Pin 7   Output Switch   SwitchOffPin7()
Relay Z Impedance Logic CalculateZ()
?? Next Steps
Electrical Equations
"   Impedance: Z=rg+jxdZ = r_g + jx_d
"   Power flow: S=V?I=P+jQS = V \cdot I = P + jQ
"   Energy saving: E=i?t1t2E = i \int_{t_1}^{t_2}
"   Maximum power transfer: Pmax=V24ZP_{max} = \frac{V^2}{4Z}
"   Signal control: x?(t)=Ax(t)+Bu(t), Y(t)=Cx(t)+Du(t)\dot{x}(t) = Ax(t) + Bu(t),\ Y(t) = Cx(t) + Du(
t)
?? Visual Basic Integration
' Relay control based on impedance
```

```
UserForm1 - 6

Dim Z As Complex
Z = New Complex(rg, xd)

If Z.Magnitude > threshold Then
    ActivateRelay()
End If

Sub ActivateRelay()
    MsgBox "Relay Triggered"
End Sub
?? Logigramme & Algorigramme Mapping
?? Logigramme: Relay Activation
[Start]
   Print
[Measure Current Z]
   Print
[Compare with Threshold]
   ???(Above)??> [Trigger Relay]
   ???(Below)??> [Continue Monitoring]
   Print
End
?? Algorigramme: SCADA Switch + LED Control
plaintext
[Initialize]
   Print
[Check SCADA Switch]
   Print
[If ON]
   ???> [Read Amp Level]
   ???> [If Amp > 5 ? LED ON]
   ???> [Else ? LED OFF]
   Print
End
```


Label1, integrator intelligence artificial module career and training experimental student learner lecture careeer trade and technology Power supply , register , accumulator,
Fram
Label2
text1=label2and co
Commai
Comm
Label3
text2=label3and co
Label4
text4=label5and co
Label5
text6=label5and co
Label6
text7=label6 and cc
Label7
text8=label7and co
Label8
text9=label8anc
Label9
text10=label9and
Label10
text11=label10a
Label11
text12=label11a
Label12
text13=label12a
Label13
text14=label13a
Commanc
Label14
text15=label15a
Label15
text14=label15a
Label16
text15=label16a
Label17
text16=label17a
Label18
text17=label18a
Label19
text18=label19a
Label20
text19=label20a
Label21
text20=label20 a
Label22
Label23
Label24
Label25
Label26
Label27
Label28
Label29
Label30
Label31
ok
help
cancel
Tab1
Tab2