﻿

Public Class Form1

Public ioMgr As Ivi.Visa.Interop.ResourceManager

Public DMM1 As Ivi.Visa.Interop.FormattedIO488

Public DMM2 As Ivi.Visa.Interop.FormattedIO488

Public DMM3 As Ivi.Visa.Interop.FormattedIO488

Public DMM4 As Ivi.Visa.Interop.FormattedIO488

Public AudioAnalyzer As Ivi.Visa.Interop.FormattedIO488

Public Ocilliscope As Ivi.Visa.Interop.FormattedIO488

Public PSU As Ivi.Visa.Interop.FormattedIO488

Public SignalGen As Ivi.Visa.Interop.FormattedIO488

Public Sub Form1\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

ioMgr = New Ivi.Visa.Interop.ResourceManager

DMM1 = New Ivi.Visa.Interop.FormattedIO488

DMM2 = New Ivi.Visa.Interop.FormattedIO488

DMM3 = New Ivi.Visa.Interop.FormattedIO488

DMM4 = New Ivi.Visa.Interop.FormattedIO488

AudioAnalyzer = New Ivi.Visa.Interop.FormattedIO488

Ocilliscope = New Ivi.Visa.Interop.FormattedIO488

PSU = New Ivi.Visa.Interop.FormattedIO488

SignalGen = New Ivi.Visa.Interop.FormattedIO488

End Sub

Private Sub AssignAddress()

DMM1.IO = ioMgr.Open(("TCPIP0::" & IPaddress.Text & "::" & gpibAddress.Text & "," & CDec(DMM\_gpib\_1.Text) & "::INSTR")) ' 10 is the instr. address

DMM2.IO = ioMgr.Open(("TCPIP0::" & IPaddress.Text & "::" & gpibAddress.Text & "," & CDec(DMM\_gpib\_2.Text) & "::INSTR")) ' 10 is the instr. address

DMM3.IO = ioMgr.Open(("TCPIP0::" & IPaddress.Text & "::" & gpibAddress.Text & "," & CDec(DMM\_gpib\_3.Text) & "::INSTR")) ' 10 is the instr. address

DMM4.IO = ioMgr.Open(("TCPIP0::" & IPaddress.Text & "::" & gpibAddress.Text & "," & CDec(DMM\_gpib\_4.Text) & "::INSTR")) ' 10 is the instr. address

AudioAnalyzer.IO = ioMgr.Open(("TCPIP0::" & IPaddress.Text & "::" & gpibAddress.Text & "," & CDec(AudioAnalyzer\_gpib.Text) & "::INSTR")) ' 10 is the instr. address

Ocilliscope.IO = ioMgr.Open(("TCPIP0::" & IPaddress.Text & "::" & gpibAddress.Text & "," & CDec(Ocilliscope\_gpib.Text) & "::INSTR")) ' 10 is the instr. address

PSU.IO = ioMgr.Open(("TCPIP0::" & IPaddress.Text & "::" & gpibAddress.Text & "," & CDec(PSU\_gpib.Text) & "::INSTR")) ' 10 is the instr. address

SignalGen.IO = ioMgr.Open(("TCPIP0::" & IPaddress.Text & "::" & gpibAddress.Text & "," & CDec(SignalGen\_gpib.Text) & "::INSTR")) ' 10 is the instr. address

End Sub

Private Sub NameDevices\_Click(sender As Object, e As EventArgs) Handles NameDevices.Click

AssignAddress()

DMM1.WriteString("DISP:TEXT """ & DMM\_Name\_1.Text & """")

DMM2.WriteString("DISP:TEXT """ & DMM\_Name\_2.Text & """")

DMM3.WriteString("DISP:TEXT """ & DMM\_Name\_3.Text & """")

DMM4.WriteString("DISP:TEXT """ & DMM\_Name\_4.Text & """")

AudioAnalyzer.WriteString("21.0 SP")

Ocilliscope.WriteString("DISP:TEXT """ & Ocilliscope\_name.Text & """")

PSU.WriteString("DISP:TEXT """ & PSU\_name.Text & """")

SignalGen.WriteString("DISP:TEXT """ & SignalGen\_name.Text & """")

'Set to default

DMM\_function\_1.SelectedIndex = 0

DMM\_function\_2.SelectedIndex = 0

DMM\_function\_3.SelectedIndex = 0

DMM\_function\_4.SelectedIndex = 0

Signal\_Type.SelectedIndex = 0

Signal\_Amp\_unit.SelectedIndex = 0

End Sub

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles READ\_DMM.Click

DMM\_Read\_1()

DMM\_Read\_2()

DMM\_Read\_3()

DMM\_Read\_4()

End Sub

Public Sub DMM\_Config\_1()

' clear the display

DMM1.WriteString("DISP:TEXT:CLEAR")

' 0 V DC

If DMM\_function\_1.SelectedIndex = 0 Then

DMM1.WriteString("MEAS:VOLT:DC?")

' 1 I DC

ElseIf DMM\_function\_1.SelectedIndex = 1 Then

DMM1.WriteString("MEAS:CURR:DC?")

'2 V AC

ElseIf DMM\_function\_1.SelectedIndex = 2 Then

DMM1.WriteString("MEAS:VOLT:AC?")

'3 I AC

ElseIf DMM\_function\_1.SelectedIndex = 3 Then

DMM1.WriteString("MEAS:CURR:AC?")

'4 R 2W

ElseIf DMM\_function\_1.SelectedIndex = 4 Then

DMM1.WriteString("MEAS:RES?")

'5 R 4W

ElseIf DMM\_function\_1.SelectedIndex = 5 Then

DMM1.WriteString("MEAS:FRES?")

'6 Diode

ElseIf DMM\_function\_1.SelectedIndex = 6 Then

DMM1.WriteString("MEAS:DIODe?")

'7 Capacitance

ElseIf DMM\_function\_1.SelectedIndex = 7 Then

DMM1.WriteString("MEAS:FREQuency?")

'8 Freq

ElseIf DMM\_function\_1.SelectedIndex = 8 Then

DMM1.WriteString("MEAS:FREQuency?")

'9 Period

ElseIf DMM\_function\_1.SelectedIndex = 9 Then

DMM1.WriteString("MEAS:PERiod?")

'10 Continuity

ElseIf DMM\_function\_1.SelectedIndex = 10 Then

DMM1.WriteString("MEAS:CONTinuity?")

End If

DMM\_Read\_1()

End Sub

Public Sub DMM\_Config\_2()

' clear the display

DMM2.WriteString("DISP:TEXT:CLEAR")

' 0 V DC

If DMM\_function\_2.SelectedIndex = 0 Then

DMM2.WriteString("MEAS:VOLT:DC?")

' 1 I DC

ElseIf DMM\_function\_2.SelectedIndex = 1 Then

DMM2.WriteString("MEAS:CURR:DC?")

'2 V AC

ElseIf DMM\_function\_2.SelectedIndex = 2 Then

DMM2.WriteString("MEAS:VOLT:AC?")

'3 I AC

ElseIf DMM\_function\_2.SelectedIndex = 3 Then

DMM2.WriteString("MEAS:CURR:AC?")

'4 R 2W

ElseIf DMM\_function\_2.SelectedIndex = 4 Then

DMM2.WriteString("MEAS:RES?")

'5 R 4W

ElseIf DMM\_function\_2.SelectedIndex = 5 Then

DMM2.WriteString("MEAS:FRES?")

'6 Diode

ElseIf DMM\_function\_2.SelectedIndex = 6 Then

DMM2.WriteString("MEAS:DIODe?")

'7 Capacitance

ElseIf DMM\_function\_2.SelectedIndex = 7 Then

DMM2.WriteString("MEAS:FREQuency?")

'8 Freq

ElseIf DMM\_function\_2.SelectedIndex = 8 Then

DMM2.WriteString("MEAS:FREQuency?")

'9 Period

ElseIf DMM\_function\_2.SelectedIndex = 9 Then

DMM2.WriteString("MEAS:PERiod?")

'10 Continuity

ElseIf DMM\_function\_2.SelectedIndex = 10 Then

DMM2.WriteString("MEAS:CONTinuity?")

End If

'read the value

DMM\_Read\_2()

End Sub

Public Sub DMM\_Config\_3()

' clear the display

DMM3.WriteString("DISP:TEXT:CLEAR")

' 0 V DC

If DMM\_function\_3.SelectedIndex = 0 Then

DMM3.WriteString("MEAS:VOLT:DC?")

' 1 I DC

ElseIf DMM\_function\_3.SelectedIndex = 1 Then

DMM3.WriteString("MEAS:CURR:DC?")

'2 V AC

ElseIf DMM\_function\_3.SelectedIndex = 2 Then

DMM3.WriteString("MEAS:VOLT:AC?")

'3 I AC

ElseIf DMM\_function\_3.SelectedIndex = 3 Then

DMM3.WriteString("MEAS:CURR:AC?")

'4 R 2W

ElseIf DMM\_function\_3.SelectedIndex = 4 Then

DMM3.WriteString("MEAS:RES?")

'5 R 4W

ElseIf DMM\_function\_3.SelectedIndex = 5 Then

DMM3.WriteString("MEAS:FRES?")

'6 Diode

ElseIf DMM\_function\_3.SelectedIndex = 6 Then

DMM3.WriteString("MEAS:DIODe?")

'7 Capacitance

ElseIf DMM\_function\_3.SelectedIndex = 7 Then

DMM3.WriteString("MEAS:FREQuency?")

'8 Freq

ElseIf DMM\_function\_3.SelectedIndex = 8 Then

DMM3.WriteString("MEAS:FREQuency?")

'9 Period

ElseIf DMM\_function\_3.SelectedIndex = 9 Then

DMM3.WriteString("MEAS:PERiod?")

'10 Continuity

ElseIf DMM\_function\_3.SelectedIndex = 10 Then

DMM3.WriteString("MEAS:CONTinuity?")

End If

DMM\_Read\_3()

End Sub

Public Sub DMM\_Config\_4()

' clear the display

DMM4.WriteString("DISP:TEXT:CLEAR")

' 0 V DC

If DMM\_function\_4.SelectedIndex = 0 Then

DMM4.WriteString("MEAS:VOLT:DC?")

' 1 I DC

ElseIf DMM\_function\_4.SelectedIndex = 1 Then

DMM4.WriteString("MEAS:CURR:DC?")

'2 V AC

ElseIf DMM\_function\_4.SelectedIndex = 2 Then

DMM4.WriteString("MEAS:VOLT:AC?")

'3 I AC

ElseIf DMM\_function\_4.SelectedIndex = 3 Then

DMM4.WriteString("MEAS:CURR:AC?")

'4 R 2W

ElseIf DMM\_function\_4.SelectedIndex = 4 Then

DMM4.WriteString("MEAS:RES?")

'5 R 4W

ElseIf DMM\_function\_4.SelectedIndex = 5 Then

DMM4.WriteString("MEAS:FRES?")

'6 Diode

ElseIf DMM\_function\_4.SelectedIndex = 6 Then

DMM4.WriteString("MEAS:DIODe?")

'7 Capacitance

ElseIf DMM\_function\_4.SelectedIndex = 7 Then

DMM4.WriteString("MEAS:FREQuency?")

'8 Freq

ElseIf DMM\_function\_4.SelectedIndex = 8 Then

DMM4.WriteString("MEAS:FREQuency?")

'9 Period

ElseIf DMM\_function\_4.SelectedIndex = 9 Then

DMM4.WriteString("MEAS:PERiod?")

'10 Continuity

ElseIf DMM\_function\_4.SelectedIndex = 10 Then

DMM4.WriteString("MEAS:CONTinuity?")

End If

DMM\_Read\_4()

End Sub

Public Sub DMM\_Read\_1()

'read the value

DMM\_Value\_1.Text = DMM1.ReadNumber()

'if it's Vac calculate dBm

If DMM\_function\_1.SelectedIndex = 2 Then

DMM\_dbm\_1.Text = 10 \* Math.Log10((CDbl(DMM\_Value\_1.Text) ^ 2) / CDbl(dBm\_ref.Text) / (0.001))

Else

DMM\_dbm\_1.Text = ""

End If

End Sub

Public Sub DMM\_Read\_2()

DMM\_Value\_2.Text = DMM2.ReadNumber()

'if it's Vac calculate dBm

If DMM\_function\_2.SelectedIndex = 2 Then

DMM\_dbm\_2.Text = 10 \* Math.Log10((CDbl(DMM\_Value\_2.Text) ^ 2) / CDbl(dBm\_ref.Text) / (0.001))

Else

DMM\_dbm\_2.Text = ""

End If

End Sub

Public Sub DMM\_Read\_3()

'read the value

DMM\_Value\_3.Text = DMM3.ReadNumber()

'if it's Vac calculate dBm

If DMM\_function\_3.SelectedIndex = 2 Then

DMM\_dbm\_3.Text = 10 \* Math.Log10((CDbl(DMM\_Value\_3.Text) ^ 2) / CDbl(dBm\_ref.Text) / (0.001))

Else

DMM\_dbm\_3.Text = ""

End If

End Sub

Public Sub DMM\_Read\_4()

'read the value

DMM\_Value\_4.Text = DMM4.ReadNumber()

'if it's Vac calculate dBm

If DMM\_function\_4.SelectedIndex = 2 Then

DMM\_dbm\_4.Text = 10 \* Math.Log10((CDbl(DMM\_Value\_4.Text) ^ 2) / CDbl(dBm\_ref.Text) / (0.001))

Else

DMM\_dbm\_4.Text = ""

End If

End Sub

Private Sub DMM\_function\_1\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles DMM\_function\_1.SelectedIndexChanged

DMM\_Config\_1()

End Sub

Private Sub DMM\_function\_2\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles DMM\_function\_2.SelectedIndexChanged

DMM\_Config\_2()

End Sub

Private Sub DMM\_function\_3\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles DMM\_function\_3.SelectedIndexChanged

DMM\_Config\_3()

End Sub

Private Sub DMM\_function\_4\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles DMM\_function\_4.SelectedIndexChanged

DMM\_Config\_4()

End Sub

Private Sub Label12\_Click(sender As Object, e As EventArgs) Handles Label12.Click

End Sub

Private Sub DomainUpDown1\_SelectedItemChanged(sender As Object, e As EventArgs)

End Sub

Private Sub Button5\_Click(sender As Object, e As EventArgs)

End Sub

Private Sub Button1\_Click\_2(sender As Object, e As EventArgs) Handles Button1.Click

config\_SigGen()

SignalGen.WriteString("OUTPUT ON")

End Sub

Public Sub config\_SigGen()

SignalGen.WriteString("DISPLAY:TEXT:CLEAR")

SignalGen.WriteString("FUNC " & Signal\_Type.Text)

SignalGen.WriteString("FREQ " & Sine\_freq.Text)

SignalGen.WriteString("VOLT:UNIT " & Signal\_Amp\_unit.Text)

SignalGen.WriteString("VOLT " & Sine\_amp.Text)

End Sub

Private Sub Sine\_amp\_TextChanged(sender As Object, e As EventArgs) Handles Sine\_amp.TextChanged

End Sub

Private Sub DutyCycle\_Scroll(sender As Object, e As EventArgs) Handles DutyCycle.Scroll

If Signal\_Type.Text = "SQU" Then

SignalGen.WriteString("FUNC:SQU:DCYC " & DutyCycle.Value)

ElseIf Signal\_Type.Text = "RAMP" Then

SignalGen.WriteString("FUNC:RAMP:SYMM " & DutyCycle.Value)

Else

End If

End Sub

Private Sub Signal\_Amp\_unit\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles Signal\_Amp\_unit.SelectedIndexChanged

End Sub

Private Sub Signal\_Type\_SelectedIndexChanged(sender As Object, e As EventArgs) Handles Signal\_Type.SelectedIndexChanged

End Sub

Private Sub Sine\_freq\_TextChanged(sender As Object, e As EventArgs) Handles Sine\_freq.TextChanged

TrackBar1.Value = Sine\_freq.Text

End Sub

Private Sub DistortionChange\_Click(sender As Object, e As EventArgs) Handles DistortionChange.Click

AudioAnalyzer.WriteString("CL")

AudioAnalyzer.WriteString("FR " & Dist\_freq.Text & " HZ")

AudioAnalyzer.WriteString("AP " & Dist\_Amp.Text & " VL")

'Dist\_Dist\_measure\_DSTN

AudioAnalyzer.WriteString("S3")

'hold

AudioAnalyzer.WriteString("T3")

AudioAnalyzer.WriteString("RR")

Dist\_measure\_DSTN.Text = AudioAnalyzer.ReadString()

'Dist\_measure\_SINAD

' AudioAnalyzer.WriteString("M2")

' Dist\_measure\_SINAD.Text = AudioAnalyzer.ReadNumber()

'Dist\_measure\_Voltage

' AudioAnalyzer.WriteString("MV")

' Dist\_measure\_Voltage.Text = AudioAnalyzer.ReadNumber()

End Sub

Private Sub Dist\_measure\_SINAD\_Click(sender As Object, e As EventArgs) Handles Dist\_measure\_SINAD.TextChanged

End Sub

Private Sub Dist\_Dist\_measure\_DSTN\_click(sender As Object, e As EventArgs) Handles Dist\_measure\_DSTN.TextChanged

End Sub

Private Sub Dist\_measure\_Voltage\_Click(sender As Object, e As EventArgs) Handles Dist\_measure\_Voltage.TextChanged

End Sub

Private Sub Ocilliscope\_gpib\_TextChanged(sender As Object, e As EventArgs) Handles Ocilliscope\_gpib.TextChanged

End Sub

Private Sub PSU\_gpib\_TextChanged(sender As Object, e As EventArgs) Handles PSU\_gpib.TextChanged

End Sub

Private Sub TrackBar1\_Scroll(sender As Object, e As EventArgs) Handles TrackBar1.Scroll

Sine\_freq.Text = TrackBar1.Value

SignalGen.WriteString("FREQ " & Sine\_freq.Text)

End Sub

End Class