

LAMPIRAN

Source code

```
Option Explicit On
Imports ADODB

Public Class FormFishPond
    Dim SimPH As Double
    Dim SimSal As Double
    Dim tick As Long
    Dim InterPH As Single
    Dim SlopePH As Single
    Dim InterSalin As Single
    Dim SlopeSalin As Single
    Dim SensBacaPH(5) As Long
    Dim SensCalibPH(5) As Single
    Dim SensBacaSalin(5) As Long
    Dim SensCalibSalin(5) As Single

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        FuzzyPH.GetINI()
        FuzzyPH.RefreshRuleBase()
        FuzzyPH.RefreshKurva()
        FuzzySalin.GetINI()
        FuzzySalin.RefreshRuleBase()
        FuzzySalin.RefreshKurva()
        SensorRefresh()

        udIterasi.Value = Val(GetSettingINI("Control", "Iter", "1"))
        udChartRange.Value = Val(GetSettingINI("Control", "ChartRange",
"100"))
        ScrollFilter.Value = Val(GetSettingINI("Control", "Filter", "50"))
        LabelFilter.Text = "Filter : " & FormatNumber(0.01 *
ScrollFilter.Value, 2)

        SimPH = 7
        SimSal = 10
        ControlPH.Setpoint = 7
        ControlSalin.Setpoint = 10
        ControlPH.InitControl()
        ControlSalin.InitControl()

    End Sub

    Private Sub RefreshRuleBasePH_Click(sender As Object, e As EventArgs)
Handles RefreshRuleBasePH.Click
        FuzzyPH.RefreshRuleBase()
    End Sub

    Private Sub RefreshRuleBaseSalin_Click(sender As Object, e As
EventArgs) Handles RefreshRuleBaseSalin.Click
        FuzzySalin.RefreshRuleBase()
    End Sub
End Class
```

```

Private Sub cbRunStop_CheckedChanged(sender As Object, e As EventArgs)
Handles cbRunStop.CheckedChanged
    TimerFuzz.Enabled = cbRunStop.Checked
    udJog.Enabled = Not cbRunStop.Checked
    udIterasi.Enabled = Not cbRunStop.Checked
    ScrollMin.Enabled = Not cbRunStop.Checked
    ScrollMax.Enabled = Not cbRunStop.Checked
    If cbRunStop.Checked = True Then
        ScrollMin.Minimum = 0
        ScrollMin.Maximum = tick
        ScrollMin.Value = IIf(tick > udChartRange.Value, tick -
udChartRange.Value, 0)
        ScrollMax.Maximum = tick
        ScrollMax.Value = tick
        ScrollMax.Minimum = ScrollMin.Value
        ListTX.Items.Add("*J0" & udJog.Value.ToString("000") & "#")
    Else
        ListTX.Items.Add("*J0000#")
    End If
End Sub

Private Sub ScrollMin_Scroll(sender As Object, e As ScrollEventArgs)
Handles ScrollMin.Scroll
    If ScrollMin.Enabled = True Then
        ScrollMax.Minimum = ScrollMin.Value
        ControlPH.ChartControl.ChartAreas(0).AxisX.Minimum =
ScrollMin.Value
        ControlSalin.ChartControl.ChartAreas(0).AxisX.Minimum =
ScrollMin.Value
    End If
End Sub

Private Sub ScrollMax_Scroll(sender As Object, e As ScrollEventArgs)
Handles ScrollMax.Scroll
    If ScrollMax.Enabled = True Then
        ScrollMin.Maximum = ScrollMax.Value
        ControlPH.ChartControl.ChartAreas(0).AxisX.Maximum =
ScrollMax.Value
        ControlSalin.ChartControl.ChartAreas(0).AxisX.Maximum =
ScrollMax.Value
    End If
End Sub

Public SensorPH As Single
Public SensorSalin As Single
Public MotorPH As Single
Public MotorSalin As Single
Public QueRX232 As String

```

```

Private Sub TimerFuzz_Tick(sender As Object, e As EventArgs) Handles
TimerFuzz.Tick
    If cbSimulation.Checked = True Then
        ControlPH.Sensor = SimPH
        ControlSalin.Sensor = SimSal
    Else
        'ControlPH.Sensor = SimPH
        'ControlSalin.Sensor = SimSal
        ControlPH.Sensor = SensorPH * SlopePH + InterPH
        ControlSalin.Sensor = SensorSalin * SlopeSalin + InterSalin
    End If
End Sub

```

```

End If
tick += 1
ControlPH.DoControl()
ControlSalin.DoControl()
If cbSimulation.Checked = True Then
    SimPH = SimPH + 0.01 * FuzzyPH.Defuzz
    SimSal = SimSal + 0.05 * FuzzySalin.Defuzz
    MotorPH = FuzzyPH.Defuzz
    MotorSalin = FuzzySalin.Defuzz
Else
    SimPH = SimPH + 0.01 * FuzzyPH.Defuzz
    SimSal = SimSal + 0.05 * FuzzySalin.Defuzz
    MotorPH = FuzzyPH.Defuzz
    MotorSalin = FuzzySalin.Defuzz
    If MotorPH < -100 Then MotorPH = -100
    If MotorPH > 100 Then MotorPH = 100

    If MotorPH > 0 Then
        ListTX.Items.Add("*M1" & MotorPH.ToString("000") & "#")
    ElseIf MotorPH < 0 Then
        ListTX.Items.Add("*M2" & (-MotorPH).ToString("000") & "#")
    Else
        ListTX.Items.Add("*M1000#")
        ListTX.Items.Add("*M2000#")
    End If
    If MotorSalin < -100 Then MotorSalin = -100
    If MotorSalin > 100 Then MotorSalin = 100
    If MotorSalin > 0 Then
        ListTX.Items.Add("*M3" & MotorSalin.ToString("000") & "#")
    ElseIf MotorSalin < 0 Then
        ListTX.Items.Add("*M4" & (-MotorSalin).ToString("000") &
"#")
    Else
        ListTX.Items.Add("*M3000#")
        ListTX.Items.Add("*M4000#")
    End If
    While ListTX.Items.Count > 10
        ListTX.Items.RemoveAt(0)
    End While
    ListTX.TopIndex() = ListTX.Items.Count - 1
    Application.DoEvents()
End If
End Sub

Private Sub Timer232_Tick(sender As Object, e As EventArgs) Handles
Timer232.Tick
    If Not Port232.IsOpen Then Exit Sub
    Dim strTX As String
    Dim DataRX As String
    Dim strRX() As Byte
    Dim Filter As Single
    If Port232.BytesToRead > 0 Then
        ReDim strRX(Port232.BytesToRead)
        Port232.Read(strRX, 0, Port232.BytesToRead)
        DataRX = System.Text.Encoding.UTF8.GetString(strRX)
        ListRX.Items.Add(DataRX)
        While ListRX.Items.Count > 20
            ListRX.Items.RemoveAt(0)
        End While
        ListRX.TopIndex() = ListRX.Items.Count - 1

```

```

        Filter = 0.01 * ScrollFilter.Value
        Try
            If DataRX.Substring(0, 3) = "*S1" Then
                SensorPH = (Filter * SensorPH) + ((1 - Filter) *
DataRX.Substring(3, DataRX.Length - 5))
                TextSensPH.Text = SensorPH.ToString("#0")
            ElseIf DataRX.Substring(0, 3) = "*S2" Then
                SensorSalin = (Filter * SensorSalin) + ((1 - Filter) *
DataRX.Substring(3, DataRX.Length - 5))
                TextSensSalin.Text = SensorSalin.ToString("#0")
            End If
        Catch ex As Exception
            '
        End Try
        ElseIf ListTX.Items.Count > 0 Then
            strTX = ListTX.Items(0).ToString
            ListTX.Items.RemoveAt(0)
            Port232.Write(strTX)
        End If
    End Sub

    Private Sub TabPage1_SizeChanged(sender As Object, e As EventArgs)
Handles TabKontrol.SizeChanged
        Dim centerV As Short, centerH As Short
        centerV = 0.5 * (TabKontrol.Height - ScrollMin.Height)
        centerH = 0.5 * ControlPH.Width
        ControlPH.Width = TabKontrol.Width - ControlPH.Left
        ControlPH.Height = centerV
        ScrollMin.Top = centerV
        ScrollMin.Width = centerH
        ScrollMax.Top = centerV
        ScrollMax.Left = centerH + ScrollMin.Left
        ScrollMax.Width = centerH
        centerV += ScrollMin.Height
        ControlSalin.Top = centerV
        ControlSalin.Width = ControlPH.Width
        ControlSalin.Height = TabKontrol.Height - centerV
    End Sub

    Private Sub HScrollBar1_Scroll(sender As Object, e As ScrollEventArgs)
Handles ScrollFilter.Scroll
        LabelFilter.Text = "Filter : " & FormatNumber(0.01 *
ScrollFilter.Value, 2)
        Application.DoEvents()
    End Sub

    Private Sub BtnApply_Click(sender As Object, e As EventArgs) Handles
BtnApply.Click
        Dim Filter As Single
        TimerFuzz.Interval = udIterasi.Value * IIf(cbSimulation.Checked =
True, 1000, 1000)
        Filter = 0.01 * ScrollFilter.Value
        ControlPH.FilterInput = Filter
        ControlSalin.FilterInput = Filter
        ControlPH.ChartRange = udChartRange.Value
        ControlSalin.ChartRange = udChartRange.Value
        SaveSettingINI("Control", "Iter", udIterasi.Value)
        SaveSettingINI("Control", "ChartRange", udChartRange.Value)
        SaveSettingINI("Control", "Filter", ScrollFilter.Value)
    End Sub

```

```

Private Sub cbSimulation_CheckedChanged(sender As Object, e As
EventArgs) Handles cbSimulation.CheckedChanged
    If cbSimulation.Checked = True Then
        Timer232.Enabled = False
        If Port232.IsOpen Then Port232.Close()
        cbPort232.Enabled = True
    Else
        Timer232.Enabled = True
        If Not Port232.IsOpen Then Port232.Open()
        cbPort232.Enabled = False
    End If
End Sub

Public Sub SensorRefresh()
    Dim sql1 As String
    Dim rs1 As New ADODB.Recordset
    Dim conn As New ADODB.Connection
    Try
        conn.Open(ParamDB & FileDB)
    Catch
        MsgBox("Tidak bisa Membuka Database :" & vbCrLf & vbCrLf &
FileDB)
        Me.Close()
        Exit Sub
    End Try
    sql1 = "select * from Table_Sensor_PH order by Baca"
    rs1.Open(sql1, conn, CursorTypeEnum.adOpenStatic,
LockTypeEnum.adLockReadOnly)
    If rs1.RecordCount = 5 Then
        rs1.MoveFirst()
        GridSensPH.Rows.Clear()
        For i = 0 To 4
            GridSensPH.Rows.Add()
            GridSensPH.Rows(i).Cells(0).Value =
rs1.Fields("Baca").Value
            GridSensPH.Rows(i).Cells(1).Value =
rs1.Fields("Calib").Value
            SensBacaPH(i) = rs1.Fields("Baca").Value
            SensCalibPH(i) = rs1.Fields("Calib").Value
            rs1.MoveNext()
        Next i
    Else
        MsgBox("Error Table Sensor PH !!!" & vbCrLf & vbCrLf &
rs1.RecordCount)
        rs1.Close()
        conn.Close()
        Me.Close()
        Exit Sub
    End If
    rs1.Close()
    sql1 = "select * from Table_Sensor_Salin order by Baca"
    rs1.Open(sql1, conn, CursorTypeEnum.adOpenStatic,
LockTypeEnum.adLockReadOnly)
    If rs1.RecordCount = 5 Then
        rs1.MoveFirst()
        GridSensSalin.Rows.Clear()
        For i = 0 To 4
            GridSensSalin.Rows.Add()

```

```

        GridSensSalin.Rows(i).Cells(0).Value =
rs1.Fields("Baca").Value
        GridSensSalin.Rows(i).Cells(1).Value =
rs1.Fields("Calib").Value
        SensBacaSalin(i) = rs1.Fields("Baca").Value
        SensCalibSalin(i) = rs1.Fields("Calib").Value
        rs1.MoveNext()
    Next i
Else
MsgBox("Error Table Sensor Salinitas !!!" & vbCrLf & vbCrLf &
rs1.RecordCount)
    rs1.Close()
    conn.Close()
    Me.Close()
    Exit Sub
End If
rs1.Close()
conn.Close()
Dim SigX As Double = 0
Dim SigY As Double = 0
Dim SigX2 As Double = 0
Dim SigXY As Double = 0
For i = 0 To 4
    SigX += SensBacaPH(i)
    SigY += SensCalibPH(i)
    SigX2 += (SensBacaPH(i) * SensBacaPH(i))
    SigXY += SensBacaPH(i) * SensCalibPH(i)
Next i
InterPH = ((SigY * SigX2) - (SigX * SigXY)) / ((5 * SigX2) - (SigX
* SigX))
SlopePH = ((5 * SigXY) - (SigX * SigY)) / ((5 * SigX2) - (SigX *
SigX))
SigX = 0
SigY = 0
SigX2 = 0
SigXY = 0
For i = 0 To 4
    SigX += SensBacaSalin(i)
    SigY += SensCalibSalin(i)
    SigX2 += (SensBacaSalin(i) * SensBacaSalin(i))
    SigXY += SensBacaSalin(i) * SensCalibSalin(i)
Next i
InterSalin = ((SigY * SigX2) - (SigX * SigXY)) / ((5 * SigX2) -
(SigX * SigX))
SlopeSalin = ((5 * SigXY) - (SigX * SigY)) / ((5 * SigX2) - (SigX *
SigX))
End Sub

Public Sub SensorSimpan()
    Dim i As Short
    For i = 0 To 4
        SensBacaPH(i) = Val(GridSensPH.Rows(i).Cells(0).Value)
        SensCalibPH(i) = Val(GridSensPH.Rows(i).Cells(1).Value)
    Next i
    For i = 0 To 4
        SensBacaSalin(i) = Val(GridSensSalin.Rows(i).Cells(0).Value)
        SensCalibSalin(i) = Val(GridSensSalin.Rows(i).Cells(1).Value)
    Next i
    Dim conn As New ADODB.Connection
    Dim rs1 As New ADODB.Recordset

```

```

Dim sql As String
Try
    conn.Open(ParamDB & FileDB)
Catch
    MsgBox("Tidak bisa Membuka Database :" & vbCrLf & vbCrLf &
FileDB)
    Me.Close()
    Exit Sub
End Try
sql = "select * from Table_Sensor_PH order by Baca"
rs1.Open(sql, conn, CursorTypeEnum.adOpenStatic,
LockTypeEnum.adLockOptimistic)
If rs1.RecordCount = 5 Then
    rs1.MoveFirst()
    For i = 0 To 4
        rs1.Fields("Baca").Value = SensBacaPH(i)
        rs1.Fields("Calib").Value = SensCalibPH(i)
        rs1.MoveNext()
    Next i
    rs1.UpdateBatch()
Else
    MsgBox("Error Table !!!" & vbCrLf & vbCrLf & rs1.RecordCount)
    rs1.Close()
    conn.Close()
    Me.Close()
    Exit Sub
End If
rs1.Close()
sql = "select * from Table_Sensor_Salin order by Baca"
rs1.Open(sql, conn, CursorTypeEnum.adOpenStatic,
LockTypeEnum.adLockOptimistic)
If rs1.RecordCount = 5 Then
    rs1.MoveFirst()
    For i = 0 To 4
        rs1.Fields("Baca").Value = SensBacaSalin(i)
        rs1.Fields("Calib").Value = SensCalibSalin(i)
        rs1.MoveNext()
    Next i
    rs1.UpdateBatch()
Else
    MsgBox("Error Table !!!" & vbCrLf & vbCrLf & rs1.RecordCount)
    rs1.Close()
    conn.Close()
    Me.Close()
    Exit Sub
End If
rs1.Close()
conn.Close()
End Sub

Private Sub BtnRefreshSensor_Click(sender As Object, e As EventArgs)
Handles BtnRefreshSensor.Click
    SensorRefresh()
End Sub

Private Sub BtnSimpanSensor_Click(sender As Object, e As EventArgs)
Handles BtnSimpanSensor.Click
    SensorSimpan()
    SensorRefresh()
End Sub

```

```

Private Sub TextSensPH_TextChanged(sender As Object, e As EventArgs)
Handles TextSensPH.TextChanged
Try
TextCalibPH.Text = (Val(TextSensPH.Text) * SlopePH +
InterPH).ToString("#0.00")
Catch ex As Exception
TextCalibPH.Text = ""
End Try
End Sub

Private Sub TextSensSalin_TextChanged(sender As Object, e As EventArgs)
Handles TextSensSalin.TextChanged
Try
TextCalibSalin.Text = (Val(TextSensSalin.Text) * SlopeSalin +
InterSalin).ToString("#0.0")
Catch ex As Exception
TextCalibSalin.Text = ""
End Try
End Sub

End Class

```

