BAB VI

KESIMPULAN DAN SARAN

6.1. Kesimpulan

Paket program aplikasi yang ditulis dengan bahasa pemrograman *Visual Basic 6.0* dengan nama ***Alignment Analysis Program***, masih mempunyai banyak kesalahan dalam proses perhitungan maupun dalam sistematis kerja program. Program ini juga belum dapat menampilkan gambar trase jalan yang diciptakan dan tidak dapat memproses perhitungan alinemen vertikal. Kelengkapan kontrol program ini juga masih kurang untuk digunakan sebagai program perencanaan geometrik jalan raya yang mudah untuk digunakan.

Setelah dikembangkan menjadi ***Alignment Analysis Program 2.0***, program ini jauh lebih sempurna daripada program awalnya. Selain kontrol kesalahan, akurasi proses perhitungan dan kelengkapan data *input* dan *output*nya, ***Alignment Analysis Program 2.0*** juga sudah dilengkapi dengan gambar trase jalan yang dihasilkan serta hasil perhitungan perencanaan alinemen vertikal. Kesalahan – kesalahan baik yang terdapat dalam proses perhitungan maupun elemen – elemen fungsional program dalam ***Alignment Analysis Program*** telah diperbaiki. Kontrol program juga sudah dilengkapi agar program dapat dioperasikan dengan mudah dan cepat. Koordinasi antara alinemen horisontal dan alinemen vertikal sesuai dengan Standar Perencanaan Geometrik untuk Jalan Perkotaan, Maret 1992, sudah dapat diperhitungkan dalam satu paket program. Oleh karena itu, dapat disimpulkan bahwa ***Alignment Analysis Program 2.0*** dapat digunakan dalam

mempercepat dan mempermudah proses perhitungan perencanaan geometrik jalan raya dan hasil perhitungannya sangat akurat serta dapat dipercaya.

6.2. Saran

Setelah melihat proses pengerjaan beserta hasil *output* dari program maka terdapat beberapa saran untuk memperbaiki kekurangan dan pengembangan program ke tingkat lebih lanjut. Beberapa saran yang perlu diperhatikan antara lain adalah:

1. proses pemasukan data titik-titik poligon pada program ini menggunakan sistem koordinat kartesius, sedangkan pada penerapan sebenarnya menggunakan sistem jarak dan sudut,
2. program yang dibuat dalam penulisan tugas akhir ini belum dapat memperhitungkan jumlah galian dan timbunan, sehingga untuk perhitungan galian dan timbunan harus dilakukan perhitungan manual,
3. program ini dapat dikembangkan lebih lanjut agar juga dapat memproses perhitungan perencanaan perkerasan jalan, sehingga program ini dapat digunakan untuk merencanakan jalan raya secara menyeluruh.

DAFTAR PUSTAKA

Anonim, 1990, *A Policy on Geometric Design of Highway and Street*, AASHTO.

Dalimin, 1979, *Pelaksanaan Pembangunan Jalan*, Penerbit Lestari, Jakarta.

Direktorat Jenderal Bina Marga, 1992, *Standar Perencanaan Geometrik Untuk Jalan Perkotaan*, Direktorat Pembinaan Jalan Kota, Jakarta.

Garber N. J. and Hoel L. A., 1996, *Traffic and Highway Engineering*, Department of Civil Engineering University of Virginia.

Madika, Fransisco, 1999, *Program Komputer Untuk Perencanaan Balok Prategang Penuh dan Parsial dengan menggunakan Visual Basic*, Skripsi, Fakultas Teknik, Universitas Sebelas Maret, Surakarta, dikutip dari J. Lengkutoy, 1989, *Dasar-Dasar Programming*, Mutiara Sumber Widya, Jakarta.

Nelson, Ross, 1995, *Menguasai Visual Basic for Windows Versi 3*, Penerbit PT Elex Media Komputindo, Jakarta.

Oglesby C. H. & Hicks R. G., 1993, *Teknik Jalan Raya*, Edisi Ke Empat, Penerbit Erlangga, Jakarta.

Sukirman, Silvia, 1994, *Dasar-Dasar Perencanaan Geometrik Jalan Raya*, Penerbit NOVA, Bandung.

Suryadharma, Hendra & Susanto Benidiktus, 1999, *Rekayasa Jalan Raya,* Penerbit Universitas Atma Jaya, Yogyakarta.

Winarko, Hendri, 2004, *Program Visual Basic Untuk Perencanaan Alinemen Jalan Raya*

LAMPIRAN
BAHASA PROGRAM ALIGNMENT ANALYSIS PROGRAM 2.0

Lampiran Bahasa Program *Alignment Analysis Program 2.0*

MODULE

```
General Declarations
Public Const Ko = 181913.53
Public Const KonvDegToRad = 0.017453292
Public Const KonvRadToDeg = 57.29577951
Public Const KonvRdToD = 1432.394488
Public Const pi = 3.141592654
Public Const emak = 0.1
Public Gmo(5, 5)
Public eno(5), ebo(5), bo(5)
Public fmo(120), fso(120), Vjo(120), Vho(120), Mo(200)
Public IndekX, IndekY, IndekL, IndekS As Integer
Public Lajur, BedaLandai, Shenti, Smenyiap
Public Lhenti(100), Lmenyiap(100), Ldrainase(100), Lkenyamanan(100)
Public g(100), A(100), Jarak(100), Lvterpakai(100), Ev(100)
Public Vd, Vj, Vh, Rn, en, fm, eb, b, fmak
Public S1, S2, h, M0, Dpi, Dmak
Public m, Jlajur, KelasJalan, KeadaanMedan, JumlahNode, Elevasi
Public Xpi, Ypi, Rpi
Public SudutDefleksi(100)
Public OvLap(100), StaPI(100), StaTC(100), StaCT(100), SudutCircle(100)
Public Xpc(100), Ypc(100), Xpt(100), Ypt(100), SudutSpiral(100), SudutJurusan(100)
Public Arah(100) As String
Public Tc(100), Lc(100), Ec(100), p(100), k(100),  LCurve(100)
Public Lsmin(100), Soc(100), Pelebaran(100), M_Curve(100)
Public Xmin, Ymin, SkalaMak, SkalaMin
Public D(100)
Public DataText As String
Public optNol
Public RecordModified As Boolean

Sub Main()
  frmSplash.Show
  frmSplash.Refresh
  Unload frmSplash
  MDIMain.Show
End Sub

Sub UpdateSkala()
  Xmin = X(1): Ymin = Y(1)
  Xmax = X(1): Ymax = Y(1)
  For I = 1 To JumlahNode
    If Xmin > X(I) Then Xmin = X(I)
    If Ymin > Y(I) Then Ymin = Y(I)
    If Xmak < X(I) Then Xmak = X(I)
    If Ymak < Y(I) Then Ymak = Y(I)
  Next I
  SkalaMak = Xmak: If Xmak < Ymak Then SkalaMak = Ymak
  SkalaMin = Xmin: If Xmin > Ymin Then SkalaMin = Ymin
End Sub

Sub InitDataGeometrik()
'Standar geometri Bina Marga
  eno(0) = 0.02: eno(1) = 0.02: eno(2) = 0.02: eno(3) = 0.03: eno(4) = 0.04
  ebo(0) = 4: ebo(1) = 4: ebo(2) = 6: ebo(3) = 6: ebo(4) = 6
  bo(0) = 3.75: bo(1) = 3.5: bo(2) = 3.5: bo(3) = 3: bo(4) = 2.5
```

```
'Koefisien Gesekan Memanjang
fmo(100) = 0.3: fmo(80) = 0.31: fmo(60) = 0.33: fmo(50) = 0.35: fmo(40) = 0.38: fmo(30) = 0.44:
fmo(30) = 0.44
'Koefisien Gesekan Melintang
fso(100) = 0.11: fso(80) = 0.12: fso(60) = 0.13: fso(50) = 0.14: fso(40) = 0.15: fso(30) = 0.15: fso(20) = 0.15
'Kecepatan Perjalanan di Tikungan
Vjo(100) = 74: Vjo(80) = 64: Vjo(60) = 52: Vjo(50) = 45: Vjo(40) = 37: Vjo(30) = 28: Vjo(20) = 19
'Kecepatan Kendaraan dalam Keadaan Hujan
Vho(100) = 85: Vho(80) = 68: Vho(60) = 54: Vho(50) = 45: Vho(40) = 36: Vho(30) = 30: Vho(20) = 20
'Kelandaian Relatif
Mo(100) = 225: Mo(80) = 200: Mo(60) = 175: Mo(50) = 150: Mo(40) = 125: Mo(30) = 100: Mo(20) = 75
 With FrmData.CboKelasJalan
  .Clear
  .AddItem "Tipe I Kelas I"
  .AddItem "Tipe I Kelas II"
  .AddItem "Tipe II Kelas I"
  .AddItem "Tipe II Kelas II"
  .AddItem "Tipe II Kelas III"
  .AddItem "Tipe II Kelas IV"
 End With
End Sub

Sub KecepatanRencana()
 If IndekX = 0 Then
 'Jalan Tipe I Kelas I
 With FrmData.CboKecepatan
  .AddItem "100"
  .AddItem "80"
 End With
ElseIf IndekX = 1 Then
 'Jalan Tipe I Kelas II
 With FrmData.CboKecepatan
  .AddItem "80"
  .AddItem "60"
 End With
ElseIf IndekX = 2 Then
 'Jalan Tipe II Kelas I
 With FrmData.CboKecepatan
  .AddItem "60"
 End With
ElseIf IndekX = 3 Then
 'Jalan Tipe II Kelas II
 With FrmData.CboKecepatan
  .AddItem "60"
  .AddItem "50"
 End With
ElseIf IndekX = 4 Then
 'Jalan Tipe II Kelas III
 With FrmData.CboKecepatan
  .AddItem "40"
  .AddItem "30"
 End With
ElseIf IndekX = 5 Then
 'Jalan Tipe II Kelas IV
 With FrmData.CboKecepatan
  .AddItem "30"
  .AddItem "20"
 End With
 End If
End Sub
```

```
Function Sta(ByVal ZaZa)
Sta = Format(ZaZa, "##0+0##.0#")
End Function

Function Fixed(ByVal Anan)
Fixed = Format(Anan, "###0.0#")
End Function

Sub BersihList()
With FrmData.lstAbsisPI.Clear
End With
With FrmData.lstOrdinatPI.Clear
End With
With FrmData.lstRadius.Clear
End With
With FrmData.lstElevasi.Clear
End With
End Sub
```

MDI MAIN

```
General Declarations
Dim id As Integer
Dim KelasJalan, Kecepatan, Jnode, Lajur, Stasiun
Dim Absis(100), Ordinat(100), Radius(100), Elevasi(100)
Public NomorFile As Integer
Public NamaFile As String

Private Sub MDIForm_Load()
 ObjectInit
 Me.WindowState = vbMaximized
 FrmData.Show
End Sub
Private Sub MDIForm_Unload(Cancel As Integer)
 If Me.WindowState <> vbMinimized Then
   SaveSetting App.Title, "Settings", "MainLeft", Me.Left
   SaveSetting App.Title, "Settings", "MainTop", Me.Top
   SaveSetting App.Title, "Settings", "MainWidth", Me.Width
   SaveSetting App.Title, "Settings", "MainHeight", Me.Height
 End If
End Sub

Private Sub mnuFileopen_Click()
NomorFile = FreeFile
With DlgMain
 .FileName = ""
 .Filter = "Alinemen(*.tap)|*.tap"
 .Flags = cdlOFNFileMustExist Or cdlOFNPathMustExist
 .Action = 1
If .FileName = "" Then Exit Sub
Call BersihList
NamaFile = .FileName
End With
Open NamaFile For Input As NomorFile
Input #NomorFile, JNode
Input #NomorFile, KelasJalan, Kecepatan
Input #NomorFile, Lajur
Input #NomorFile, Stasiun
For z = 1 To JNode
   Input #NomorFile, X(z)
   Input #NomorFile, Y(z)
```

```
    Input #NomorFile, El(z)
    Absis(z) = X(z)
    Ordinat(z) = Y(z)
    Elevasi(z) = El(z)
Next z
For j = 1 To JNode - 2
    Input #NomorFile, Rd(j)
    Radius(j) = Rd(j)
Next j
FrmData.Show
With FrmData
    .CboKelasJalan.ListIndex = Str$(KelasJalan)
    .CboKecepatan.ListIndex = Str$(Kecepatan)
    .txtLajur.Text = Str$(Lajur)
    .txtStasiun.Text = Str$(Stasiun)
    .txtAbsisAwal = Str$(Absis(1))
    .txtOrdinatAwal = Str$(Ordinat(1))
    .txtAbsisAkhir = Str$(Absis(JNode))
    .txtOrdinatAkhir = Str$(Ordinat(JNode))
    .txtElevasiAwal = Str$(Elevasi(1))
    .txtElevasiAkhir = Str$(Elevasi(JNode))
End With
For I = 2 To JNode - 1
    With FrmData
      .txtAbsisPI = Str$(Absis(I))
      .txtOrdinatPI = Str$(Ordinat(I))
      .txtRadius = Str$(Radius(I - 1))
      .txtElevasi = Str$(Elevasi(I))
      .CmdAdd.Value = True
    End With
Next I
Close NomorFile
End Sub

Private Sub mnuFileExit_Click()
    pilih = MsgBox("Do you really want to exit the application?", 4 + 32, "Alignment")
    If pilih = 6 Then
        End
    Else
        Print ""
    End If
End Sub

Private Sub mdiForm_QueryUnload(Cancel As Integer, UnloadMode As Integer)
    Dim Msg
    If UnloadMode > 0 Then
      Msg = "Do you really want to start new project?"
    Else
      Msg = "Do you really want to exit the application?"
    End If
    If MsgBox(Msg, vbQuestion + vbYesNo, Me.Caption) = vbNo Then Cancel = True
End Sub

Private Sub mnuFileNew_Click()
  Unload Me
  ObjectInit
  Me.WindowState = vbMaximized
  FrmData.Show
End Sub

Sub ObjectInit()
    mnuFileNew.Enabled = False
```

```
    mnuFileSave.Enabled = False
    mnuFileSaveAs.Enabled = False
    mnuFilePrint.Enabled = False
    mnuWindowOutput.Enabled = False
    mnuWindowTrace.Enabled = False
    mnuWindowCurve.Enabled = False
    mnuWindowAbout.Enabled = False
End Sub

Private Sub mnuFilePrint_Click()
DlgMain.Flags = cdlPDNoWarning
DlgMain.Flags = cdlPDPrintSetup
DlgMain.Flags = cdlPDNoWarning Or cdlPDUseDevModeCopies
DlgMain.ShowPrinter
If vbOK Then
 For I = 1 To DlgMain.Copies
    Printer.FontName = "courier new"
    Printer.FontSize = 10
    Printer.Print DataText
    Printer.Print frmOutput
    Printer.EndDoc
Next I
End If
FrmTrace.picTrace.PrintForm
End Sub

Private Sub mnuFileSave_Click()
If NamaFile = "" Then
    NomorFile = 1
    DlgMain.Flags = cdlOFNPathMustExist Or cdlOFNOverwritePrompt
    DlgMain.DialogTitle = "Save"
    DlgMain.Filter = "(*tap)|*.tap"
    DlgMain.Action = 2
    If DlgMain.FileName = "" Then Exit Sub
    NamaFile = DlgMain.FileName
End If
Call WriteToDisc
End Sub

Private Sub mnuFileSaveAs_Click()
    NomorFile = 1
    DlgMain.DialogTitle = "SaveAs"
    DlgMain.Filter = "(*tap)|*.tap"
    DlgMain.Flags = cdlOFNOverwritePrompt Or cdlOFNPathMustExist
    DlgMain.Action = 2
    If DlgMain.FileName = "" Then Exit Sub
    NamaFile = DlgMain.FileName
Call WriteToDisc
End Sub

Sub WriteToDisc()
    KelasJalan = IndekX
    Kecepatan = IndekY
    Lajur = IndekL
    Stasiun = IndekS
    JNode = JumlahNode
    Open NamaFile For Output As NomorFile
    Write #NomorFile, JNode, KelasJalan, Kecepatan, Lajur, Stasiun
 For z = 1 To JNode
    Write #NomorFile, X(z)
    Write #NomorFile, Y(z)
    Write #NomorFile, El(z)
```

```
Next z
For ZaZa = 1 To JNode - 2
   Write #NomorFile, Rd(ZaZa)
Next ZaZa
 Close #NomorFile
End Sub

Private Sub mnuVieCurveDesign_Click()
   FrmMain.Caption = "Program Alinemen Horisontal" + " - " + "Output Curve Design"
   CboCurve.ListIndex = 0
   CboCurve.SetFocus
End Sub

Private Sub mnuViewAlinyemen_Click()
   FrmMain.Caption = "Program Alinemen Horisontal" + " - " + "Output Alinemen"
   FrmMain.FraAlinemen.Visible = True
   FrmMain.FraCurve.Visible = False
   FrmMain.Show
End Sub

Private Sub mnuWindowAbout_Click()
Unload FrmOutput
Unload FrmTrace
Unload FRMCurve
mnuWindowAbout.Checked = True
mnuWindowCurve.Checked = False
mnuWindowOutput.Checked = False
mnuWindowTrace.Checked = False
FrmAbout.Show
End Sub

Private Sub mnuWindowCurve_Click()
Unload FrmOutput
Unload FrmTrace
Unload FrmAbout
mnuWindowCurve.Checked = True
mnuWindowAbout.Checked = False
mnuWindowOutput.Checked = False
mnuWindowTrace.Checked = False
FRMCurve.Show
End Sub

Private Sub mnuWindowOutput_Click()
FrmOutput.TxtOutput = DataText
Unload FrmAbout
Unload FRMCurve
Unload FrmTrace
mnuWindowOutput.Checked = True
mnuWindowAbout.Checked = False
mnuWindowCurve.Checked = False
mnuWindowTrace.Checked = False
FrmOutput.Show
End Sub

Private Sub mnuWindowTrace_Click()
Unload FrmAbout
Unload FRMCurve
Unload FrmOutput
mnuWindowTrace.Checked = True
mnuWindowOutput.Checked = False
mnuWindowAbout.Checked = False
mnuWindowCurve.Checked = False
```

```
FrmTrace.Show
End Sub

Private Sub Toolbar1_ButtonClick(ByVal Button As ComctlLib.Button)
 Select Case Button.Index
   Case 1: mnuFileNew_Click
   Case 2: mnuFileopen_Click
   Case 3: mnuFileSave_Click
   Case 5: mnuFilePrint_Click
End Select
End Sub
```

FORM DATA

```
General Declarations
Option Explicit
Dim Ind As Integer
Dim ListId As Integer
Dim ChangeId As Integer
Dim n As Integer
Dim Ls, txt, I
Dim DText As String
Dim RecordValidasi As Boolean

Private Sub Form_Activate()
CboKelasJalan.SetFocus
CmdRemove.Enabled = False
CmdChange.Enabled = False
End Sub

Private Sub Form_Load()
FrmData.Height = 8010
FrmData.Width = 6090
FrmData.Top = (MDIMain.ScaleHeight - FrmData.Height) / 2
FrmData.Left = (MDIMain.ScaleWidth - FrmData.Width) / 2
Call InitDataGeometrik
End Sub

Private Sub cboKelasJalan_Click()
With MDIMain
   .mnuFileNew.Enabled = True
   .mnuFileSave.Enabled = True
   .mnuFileSaveAs.Enabled = True
   .mnuFilePrint.Enabled = True
End With
CboKecepatan.Clear
Call CurrentAddData
Call KecepatanRencana
If IndekX = 0 Then
  JLajur = Lajur
Else JLajur = 2
End If
End Sub

Private Sub cboKelasJalan_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And CboKelasJalan.Text <> "" Then CboKecepatan.SetFocus
End Sub

Private Sub CboKecepatan_Click()
CurrentAddData
End Sub
```

```
Private Sub CboKecepatan_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And CboKecepatan.Text <> "" Then txtLajur.SetFocus
End Sub

Private Sub Form_Unload(Cancel As Integer)
Me.Refresh
With MDIMain
   .mnuWindowOutput.Enabled = True
   .mnuWindowCurve.Enabled = True
   .mnuWindowTrace.Enabled = True
   .mnuWindowAbout.Enabled = True
End With
End Sub

Private Sub txtAbsisAkhir_LostFocus()
If txtAbsisAkhir.Text <> "" Then
  Validasi (txtAbsisAkhir.Text)
 If RecordValidasi = False Then
   txtAbsisAkhir.Text = ""
   txtAbsisAkhir.SetFocus
 End If
End If
End Sub

Private Sub txtAbsisAwal_Change()
CurrentAddData
GambarKoordinat
End Sub

Private Sub txtAbsisAwal_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtAbsisAwal.Text <> "" Then
  Validasi (txtAbsisAwal.Text)
 If RecordValidasi = False Then
   txtAbsisAwal.Text = ""
   txtAbsisAwal.SetFocus
 Else
   txtOrdinatAwal.SetFocus
 End If
End If
End Sub

Private Sub txtAbsisAwal_LostFocus()
If txtAbsisAwal.Text <> "" Then
  Validasi (txtAbsisAwal.Text)
 If RecordValidasi = False Then
   txtAbsisAwal.Text = ""
   txtAbsisAwal.SetFocus
 End If
End If
End Sub

Private Sub txtAbsisPI_LostFocus()
If txtAbsisPI.Text <> "" Then
  Validasi (txtAbsisPI.Text)
 If RecordValidasi = False Then
   txtAbsisPI.Text = ""
   txtAbsisPI.SetFocus
 End If
End If
End Sub
```

```
Private Sub txtElevasiPI_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtElevasiPI.Text <> "" Then
  Validasi (txtElevasiPI.Text)
 If txtElevasiPI.Text < 0 Or txtElevasiPI.Text > 1000 Then
   RecordValidasi = False
 End If
 If RecordValidasi = False Then
   txtElevasiPI.Text = ""
   txtElevasiPI.SetFocus
 Else
   CmdAdd.SetFocus
 End If
End If
End Sub

Private Sub txtElevasiPI_LostFocus()
If txtElevasiPI.Text <> "" Then
  Validasi (txtElevasiPI.Text)
 If RecordValidasi = False Then
   txtElevasiPI.Text = ""
   txtElevasiPI.SetFocus
 End If
End If
End Sub

Private Sub txtElevasiAkhir_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtElevasiAkhir.Text <> "" Then
  Validasi (txtElevasiAkhir.Text)
 If RecordValidasi = False Then
   txtElevasiAkhir.Text = ""
   txtElevasiAkhir.SetFocus
 Else
   txtAbsisPI.SetFocus
 End If
End If
End Sub

Private Sub txtElevasiAkhir_LostFocus()
If txtElevasiAkhir.Text <> "" Then
  Validasi (txtElevasiAkhir.Text)
 If RecordValidasi = False Then
   txtElevasiAkhir.Text = ""
   txtElevasiAkhir.SetFocus
 End If
End If
End Sub

Private Sub txtElevasiAwal_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtElevasiAwal.Text <> "" Then
  Validasi (txtElevasiAwal.Text)
 If RecordValidasi = False Then
   txtElevasiAwal.Text = ""
   txtElevasiAwal.SetFocus
 Else
   txtAbsisAkhir.SetFocus
 End If
End If
End Sub

Private Sub txtElevasiAwal_LostFocus()
If txtElevasiAwal.Text <> "" Then
  Validasi (txtElevasiAwal.Text)
```

```
 If RecordValidasi = False Then
   txtElevasiAwal.Text = ""
   txtElevasiAwal.SetFocus
 End If
End If
End Sub

Private Sub txtLajur_Change()
Lajur = txtLajur.Text
JLajur = Lajur
End Sub

Private Sub txtLajur_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtLajur.Text <> "" Then
  Validasi (txtLajur.Text)
 If RecordValidasi = False Then
   txtLajur.Text = ""
   txtLajur.SetFocus
 Else
   txtStasiun.SetFocus
 End If
End If
End Sub

Private Sub txtLajur_LostFocus()
If txtLajur.Text <> "" Then
  Validasi (txtLajur.Text)
 If RecordValidasi = False Then
   txtLajur.Text = ""
   txtLajur.SetFocus
 End If
End If
End Sub
Private Sub txtOrdinatAkhir_LostFocus()
If txtOrdinatAkhir.Text <> "" Then
  Validasi (txtOrdinatAkhir.Text)
 If RecordValidasi = False Then
   txtOrdinatAkhir.Text = ""
   txtOrdinatAkhir.SetFocus
 End If
End If
End Sub

Private Sub txtOrdinatAwal_Change()
CurrentAddData
GambarKoordinat
End Sub

Private Sub txtOrdinatAwal_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtOrdinatAwal.Text <> "" Then
  Validasi (txtAbsisAwal.Text)
 If RecordValidasi = False Then
   txtOrdinatAwal.Text = ""
   txtOrdinatAwal.SetFocus
 Else
   txtElevasiAwal.SetFocus
 End If
End If
End Sub

Private Sub txtAbsisAkhir_Change()
CurrentAddData
```

```
GambarKoordinat
End Sub

Private Sub txtAbsisAkhir_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtAbsisAkhir.Text <> "" Then
  Validasi (txtAbsisAkhir.Text)
 If RecordValidasi = False Then
   txtAbsisAkhir.Text = ""
   txtAbsisAkhir.SetFocus
 Else
   txtOrdinatAkhir.SetFocus
 End If
End If
End Sub

Private Sub txtOrdinatAkhir_Change()
CurrentAddData
GambarKoordinat
End Sub

Private Sub txtOrdinatAkhir_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtOrdinatAkhir.Text <> "" Then
  Validasi (txtOrdinatAkhir.Text)
 If RecordValidasi = False Then
   txtOrdinatAkhir.Text = ""
   txtOrdinatAkhir.SetFocus
 Else
   txtElevasiAkhir.SetFocus
 End If
End If
End Sub

Private Sub txtAbsisPI_Change()
ChekDataPI
End Sub

Private Sub txtAbsisPI_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtAbsisPI.Text <> "" Then
  Validasi (txtAbsisPI.Text)
 If txtAbsisPI.Text < -10000 Or txtAbsisPI.Text > 10000 Then
   RecordValidasi = False
 End If
 If RecordValidasi = False Then
   txtAbsisPI.Text = ""
   txtAbsisPI.SetFocus
 Else
   txtOrdinatPI.SetFocus
 End If
End If
End Sub

Private Sub txtOrdinatAwal_LostFocus()
If txtOrdinatAwal.Text <> "" Then
  Validasi (txtOrdinatAwal.Text)
 If RecordValidasi = False Then
   txtOrdinatAwal.Text = ""
   txtOrdinatAwal.SetFocus
 End If
End If
End Sub
```

```
Private Sub txtOrdinatPI_Change()
ChekDataPI
End Sub

Private Sub txtOrdinatPI_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtOrdinatPI.Text <> "" Then
  Validasi (txtOrdinatPI.Text)
 If txtOrdinatPI.Text < -10000 Or txtOrdinatPI.Text > 10000 Then
   RecordValidasi = False
 End If
 If RecordValidasi = False Then
   txtOrdinatPI.Text = ""
   txtOrdinatPI.SetFocus
 Else
   txtRadius.SetFocus
 End If
End If
End Sub

Private Sub txtOrdinatPI_LostFocus()
If txtOrdinatPI.Text <> "" Then
  Validasi (txtOrdinatPI.Text)
 If RecordValidasi = False Then
   txtOrdinatPI.Text = ""
   txtOrdinatPI.SetFocus
 End If
End If
End Sub

Private Sub txtRadius_Change()
ChekDataPI
End Sub

Private Sub txtRadius_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtRadius.Text <> "" Then
  Validasi (txtRadius.Text)
 If RecordValidasi = False Then
   txtRadius.Text = ""
   txtRadius.SetFocus
 Else
   txtElevasiPI.SetFocus
 End If
End If
End Sub

Private Sub LstAbsisPI_Click()
Ind = lstAbsisPI.ListIndex
Select  all  List
End Sub

Private Sub lstOrdinatPI_Click()
 Ind = lstOrdinatPI.ListIndex
 Select  all  List
End Sub

Private Sub lstRadius_Click()
 Ind = lstRadius.ListIndex
 Select_all_List
End Sub

Private Sub lstElevasiPI_Click()
 Ind = lstElevasiPI.ListIndex
```

```
 Select_all_List
End Sub

Private Sub cmdAdd_Click()
Dim Rmin
If CboKelasJalan.Text = "" Then
  MsgBox "Masukkan Kelas_Perencanaan Jalan", vbOKOnly + vbExclamation, "Error Message"
  CboKelasJalan.SetFocus
  Exit Sub
End If
If CboKecepatan.Text = "" Then
  MsgBox "Masukkan Kecepatan Rencana", vbOKOnly + vbInformation, "Error Message"
  CboKccpatan.SctFocus
  Exit Sub
End If
fmak = fso(Vd)
Rmin = Vd ^ 2 / (127 * (emak + fmak))
If Val(txtRadius.Text) < Rmin Then
  MsgBox "Jari-jari tikungan tidak boleh lebih kecil dari Rmin" & vbCrLf & "Rmin = " & Rmin, vbOKOnly
+ vbExclamation, "Error Message"
  txtRadius.SetFocus
  Exit Sub
End If
If txtElevasiPI.Text = "" Then
  MsgBox "Masukkan elevasi Point of Intersection", vbOKOnly + vbExclamation, "Error Message"
  txtElevasiPI.SetFocus
  Exit Sub
End If
If cmdChange.Caption = "Change" Then
  lstAbsisPI.AddItem txtAbsisPI.Text, ChangeId
  lstOrdinatPI.AddItem txtOrdinatPI.Text, ChangeId
  lstRadius.AddItem txtRadius.Text, ChangeId
  lstElevasiPI.AddItem txtElevasiPI.Text, ChangeId
  CurrentChangeData
  Combo1.Locked
  lblPI.Caption = "PI" & ListId + 1
Else
  lstAbsisPI.AddItem txtAbsisPI.Text
  lstOrdinatPI.AddItem txtOrdinatPI.Text
  lstRadius.AddItem txtRadius.Text
  lstElevasiPI.AddItem txtElevasiPI.Text
  CurrentAddData
  Combo1.Locked
  lblPI.Caption = "PI" & ListId + 1
End If
'Set Object Properties
txtAbsisPI = ""
txtOrdinatPI = ""
txtRadius = ""
txtElevasiPI = ""
CmdOK.Enabled = True
txtAbsisPI.SetFocus
End Sub

Private Sub cmdChange_Click()
ChangeId = Ind
lblPI.Caption = "PI" & Ind + 1
txtAbsisPI.Text = lstAbsisPI.Text
txtOrdinatPI.Text = lstOrdinatPI.Text
txtRadius.Text = lstRadius.Text
txtElevasiPI.Text = lstElevasiPI.Text
lstAbsisPI.RemoveItem Ind
```

```
lstOrdinatPI.RemoveItem Ind
lstRadius.RemoveItem Ind
lstElevasiPI.RemoveItem Ind
CmdChange.Enabled = False
CmdRemove.Enabled = False
CmdOK.Enabled = False
txtAbsisPI.SetFocus
CmdChange.Caption = "Change"
End Sub

Private Sub cmdRemove_Click()
If Ind >= 0 Then
  lstAbsisPI.RemoveItem Ind
  lstOrdinatPI.RemoveItem Ind
  lstRadius.RemoveItem Ind
  lstElevasiPI.RemoveItem Ind
Else
  Beep
End If
CurrentRemoveData
GambarKoordinat
lblPI.Caption = "PI" & ListId + 1
CmdRemove.Enabled = True
CmdChange.Enabled = True
End Sub

Private Sub cmdOk_Click()
Dim z
If CboKelasJalan.Text = "" Then
  MsgBox "Masukkan Kelas Perencanaan Jalan", _
  vbOKOnly + vbExclamation, "Error Message"
  CboKelasJalan.SetFocus
  Exit Sub
End If
If CboKecepatan.Text = "" Then
  MsgBox "Masukkan Kecepatan Rencana", _
  vbOKOnly + vbExclamation, "Error Message"
  CboKecepatan.SetFocus
  Exit Sub
End If
If txtLajur.Text = "" Or txtLajur.Text < 0 Or txtLajur.Text > 6 Then
  MsgBox "Masukkan Jumlah Lajur (nilai maksimal = 6)", _
  vbOKOnly + vbExclamation, "Error Message"
  txtLajur.SetFocus
  Exit Sub
End If
If txtStasiun.Text = "" Or txtStasiun.Text < 0 Or txtStasiun > 10000 Then
  MsgBox "Masukkan Stationing Awal (nilai < 10000 dan tidak bernilai negatif)",
  vbOKOnly + vbExclamation, "Error Message"
  txtStasiun.SetFocus
  Exit Sub
End If
If txtAbsisAwal.Text = "" Or txtAbsisAwal.Text < -10000 Or txtAbsisAwal.Text > 10000 Then
  MsgBox "Masukkan Absis Titik Awal Proyek (-10000 < nilai < 10000)",
  vbOKOnly + vbExclamation, "Error Message"
  txtAbsisAwal.SetFocus
  Exit Sub
End If
If txtOrdinatAwal.Text = "" Or txtOrdinatAwal.Text < -10000 Or txtOrdinatAwal.Text > 10000 Then
  MsgBox "Masukkan Ordinat Titik Awal Proyek (-10000 < nilai < 10000)",
  vbOKOnly + vbExclamation, "Error Message"
  txtOrdinatAwal.SetFocus
```

```
    Exit Sub
  End If
  If txtElevasiAwal.Text = "" Or txtElevasiAwal.Text < 0 Or txtElevasiAwal.Text > 1000 Then
    MsgBox "Masukkan Elevasi Titik Awal Proyek (nilai < 1000 dan tidak bernilai negatif)", _
    vbOKOnly + vbExclamation, "Error Message"
    txtElevasiAwal.SetFocus
    Exit Sub
  End If
  If txtAbsisAkhir.Text = "" Or txtAbsisAkhir.Text < -10000 Or txtAbsisAkhir.Text > 10000 Then
    MsgBox "Masukkan Absis Titik Akhir Proyek (-10000 < nilai < 10000)", _
    vbOKOnly + vbExclamation, "Error Message"
    txtAbsisAkhir.SetFocus
    Exit Sub
  End If
  If txtOrdinatAkhir.Text = "" Or txtOrdinatAkhir.Text < -10000 Or txtOrdinatAkhir.Text > 10000 Then
    MsgBox "Masukkan Ordinat Titik Akhir Proyek (-10000 < nilai < 10000)", _
    vbOKOnly + vbExclamation, "Error Message"
    txtOrdinatAkhir.SetFocus
    Exit Sub
  End If
  If txtElevasiAkhir.Text = "" Or txtElevasiAkhir.Text < 0 Or txtElevasiAkhir.Text > 1000 Then
    MsgBox "Masukkan Elevasi Titik Akhir Proyek (nilai < 1000 dan tidak bernilai negatif)"
    vbOKOnly + vbExclamation, "Error Message"
    txtElevasiAkhir.SetFocus
    Exit Sub
  End If
  If CmdAdd.Enabled = True Then
    Call Add_Data
  ElseIf lstRadius.ListCount = 0 Then
    MsgBox "Masukkan Data PI"
    vbOKOnly + vbExclamation, "Error Message"
    txtJmlPI.SetFocus
    Exit Sub
  ElseIf lstRadius.ListCount = 0 Then
    MsgBox "Masukkan Data PI", _
    vbOKOnly + vbExclamation, "Error Message"
    txtAbsisPI.SetFocus
    Exit Sub
  End If
  'Proses Perhitungan
  Jumlah = 1
  HitungSudutDefleksi
  For n = 1 To JumlahKurva - 1
   If Rd(n) < Rn Then
     HitungTitikPerpotongan
     HitungPanjangLengkungPeralihan
   Else
     e(n) = -0.02
     Lsmin(n) = 0
   End If
  PilihJenisTikungan
   If (Lc(n) = 0) And Ls < Lsmin(n) Then
     MsgBox "Jari-jari Rencana Pada Tikungan No. " & n & _
     "Kurang Besar", vbOKOnly + vbExclamation, "Error Message"
     lstRadius.Selected(n - 1) = True
     CmdChange.Value = True
     txtRadius.SetFocus
     Exit Sub
   ElseIf Lc(n) = 0 And Ls > Lsmin(n) Then
     Lsmin(n) = Ls
   End If
   Call KoordinatPC
```

```
 Call HitungPelebaran
 Call HitungM_Curve
Next n
HitungVertikal
'Kontrol Overlaping
For z = 1 To JumlahNode - 1
    OvLap(z) = D(z) - (Tc(z) + Tc(z - 1))
    StaPI(z) = StaPI(z - 1) + LCurve(z - 1) + D(z) - 2 * Tc(z - 1)
 If OvLap(z) < 3 * Vd / 3.6 Then
  If z = 1 Then
    MsgBox "Terjadi Overlapping antara Awal Proyek dan PI 1" _
     & vbCrLf & "Chek Kembali Data Anda", vbOKOnly + vbExclamation, "Error Message"
  ElseIf z = JumlahNode - 1 Then
    MsgBox "Terjadi Overlapping antara PI " & z - 1 & " dan Akhir Proyek " & vbCrLf & "Chek Kembali
Data Anda", vbOKOnly + vbExclamation, "Error Message"
    txtAbsisPI.SetFocus
    Exit Sub
  Else
    MsgBox "Terjadi Overlapping antara PI " & z - 1 & " dan PI " & z & vbCrLf & "Chek Kembali Data
Anda", _
    vbOKOnly + vbExclamation, "Error Message"
    txtAbsisPI.SetFocus
    Exit Sub
  End If
 End If
Next z
Call WriteDataText
Unload Me
End Sub

Private Sub cmdCancel_Click()
Unload Me
End Sub

Sub Initialisasi()
en = eno(IndekX)
eb = ebo(IndekX)
b = bo(IndekX)
fm = fmo(Vd)
fmak = fso(Vd)
Vj = Vjo(Vd)
Vh = Vho(Vd)
m = Mo(Vd)
Rn = (Vd ^ 2) / (127 * 0.015)
Dmak = (Ko * (emak + fmak)) / (Vd ^ 2)
h = emak * (Vd / Vj) ^ 2 - emak
Dpi = Ko * emak / Vj ^ 2
S1 = h / Dpi
S2 = (fmak - h) / (Dmak - Dpi)
M0 = Dpi * (Dmak - Dpi) * (S2 - S1) / (2 * Dmak)
End Sub

Sub HitungVertikal()
For n = 1 To JumlahNode - 1
   StaPI(n) = StaPI(n - 1) + LCurve(n - 1) + D(n) - 2 * Tc(n - 1)
Next n
For n = 1 To JumlahNode - 1
   Jarak(n) = StaPI(n) - StaPI(n - 1)
   g(n) = ((El(n + 1) - El(n)) / Jarak(n)) * 100
Next n
For n = 1 To JumlahNode - 2
   A(n) = g(n) - g(n + 1)
```

```
Next n
For n = 1 To JumlahNode - 2
PilihJenisTanjakan
If LkenyamananEX(n) > 0 And LdrainaseEX(n) > 0 And LhentiEX(n) > 0 And LmenyiapEX(n) > 0 Then
 If LkenyamananEX(n) < LdrainaseEX(n) And LkenyamananEX(n) < LmenyiapEX(n) And
LkenyamananEX(n) < LhentiEX(n) Then
   Lvterpakai(n) = Lkenyamanan(n)
 ElseIf LdrainaseEX(n) < LmenyiapEX(n) And LdrainaseEX(n) < LhentiEX(n) And LdrainaseEX(n) <
LkenyamananEX(n) Then
   Lvterpakai(n) = Ldrainase(n)
 ElseIf LhentiEX(n) < Lmenyiap(n) And LhentiEX(n) < LkenyamananEX(n) And LhentiEX(n) <
LdrainaseEX(n) Then
   Lvterpakai(n) = Lhenti(n)
 ElseIf LmenyiapEX(n) > LhentiEX(n) And LmenyiapEX(n) < LdrainaseEX(n) And LmenyiapEX(n) <
LkenyamananEX(n) Then
   Lvterpakai(n) = Lmenyiap(n)
 End If
ElseIf LdrainaseEX(n) > 0 And LhentiEX(n) > 0 And LmenyiapEX(n) > 0 Then
 If LdrainaseEX(n) < LmenyiapEX(n) And LdrainaseEX(n) < LhentiEX(n) Then
   Lvterpakai(n) = Ldrainase(n)
 ElseIf LhentiEX(n) < Lmenyiap(n) And LhentiEX(n) < LdrainaseEX(n) Then
   Lvterpakai(n) = Lhenti(n)
 ElseIf LmenyiapEX(n) > LhentiEX(n) And LmenyiapEX(n) < LdrainaseEX(n) Then
   Lvterpakai(n) = Lmenyiap(n)
 End If
ElseIf LkenyamananEX(n) > 0 And LhentiEX(n) > 0 And LmenyiapEX(n) > 0 Then
 If LkenyamananEX(n) < LmenyiapEX(n) And LkenyamananEX(n) < LhentiEX(n) Then
   Lvterpakai(n) = Lkenyamanan(n)
 ElseIf LhentiEX(n) < Lmenyiap(n) And LhentiEX(n) < LkenyamananEX(n) Then
   Lvterpakai(n) = Lhenti(n)
 ElseIf LmenyiapEX(n) > LhentiEX(n) And LmenyiapEX(n) < LkenyamananEX(n) Then
   Lvterpakai(n) = Lmenyiap(n)
 End If
ElseIf LkenyamananEA(n) > 0 And LdrainaseEA(n) > 0 And LmenyiapEA(n) > 0 Then
 If LkenyamananEX(n) < LdrainaseEX(n) And LkenyamananEX(n) < LmenyiapEX(n) Then
   Lvterpakai(n) = Lkenyamanan(n)
 ElseIf LdrainaseEX(n) < LmenyiapEX(n) And LdrainaseEX(n) < LkenyamananEX(n) Then
   Lvterpakai(n) = Ldrainase(n)
 ElseIf LmenyiapEX(n) < LdrainaseEX(n) And LmenyiapEX(n) < LkenyamananEX(n) Then
   Lvterpakai(n) = Lmenyiap(n)
 End If
ElseIf LkenyamananEA(n) > 0 And LdrainaseEA(n) > 0 And LhentiEA(n) > 0 Then
 If LkenyamananEX(n) < LdrainaseEX(n) And LkenyamananEX(n) < LhentiEX(n) Then
   Lvterpakai(n) = Lkenyamanan(n)
 ElseIf LdrainaseEX(n) < LhentiEX(n) And LdrainaseEX(n) < LkenyamananEX(n) Then
   Lvterpakai(n) = Ldrainase(n)
 ElseIf LhentiEX(n) < LkenyamananEX(n) And LhentiEX(n) < LdrainaseEX(n) Then
   Lvterpakai(n) = Lhenti(n)
 End If
ElseIf LhentiEA(n) > 0 And LmenyiapEX(n) > 0 Then
 If LhentiEX(n) < Lmenyiap(n) Then
   Lvterpakai(n) = Lhenti(n)
 ElseIf LmenyiapEX(n) > LhentiEX(n) Then
   Lvterpakai(n) = Lmenyiap(n)
 End If
ElseIf LdrainaseEX(n) > 0 And LmenyiapEX(n) > 0 Then
 If LdrainaseEX(n) < LmenyiapEX(n) Then
   Lvterpakai(n) = Ldrainase(n)
 ElseIf LmenyiapEX(n) < LdrainaseEX(n) Then
   Lvterpakai(n) = Lmenyiap(n)
 End If
ElseIf LdrainaseEX(n) > 0 And LhentiEX(n) > 0 Then
```

```
 If LdrainaseEX(n) < LhentiEX(n) Then
   Lvterpakai(n) = Ldrainase(n)
 ElseIf LhentiEX(n) < LdrainaseEX(n) Then
   Lvterpakai(n) = Lhenti(n)
 End If
ElseIf LmenyiapEX(n) > 0 Then
 If LdrainaseEX(n) < LmenyiapEX(n) And LdrainaseEX(n) < LhentiEX(n) Then
   Lvterpakai(n) = Ldrainase(n)
 ElseIf LhentiEX(n) < Lmenyiap(n) And LhentiEX(n) < LdrainaseEX(n) Then
   Lvterpakai(n) = Lhenti(n)
 ElseIf LmenyiapEX(n) > LhentiEX(n) And LmenyiapEX(n) < LdrainaseEX(n) Then
   Lvterpakai(n) = Lmenyiap(n)
 End If
End If
Ev(n) = (((A(n) * Lvterpakai(n)) / 800) ^ 2) ^ 0.5
Next n
End Sub

Sub PilihJenisTanjakan()
If IndekX = 0 Then
 If IndekY = 0 Then
   LandaiCembung = 60, LandaiCekung = 88, Shenti = 165, Smenyiap = 670, Lvmin = 85
 Else
   LandaiCembung = 45, LandaiCekung = 72, Shenti = 110, Smenyiap = 550, Lvmin = 70
 End If
ElseIf IndekX = 1 Then
 If IndekY = 0 Then
   LandaiCembung = 45, LandaiCekung = 72, Shenti = 110, Smenyiap = 550, Lvmin = 70
 Else
   LandaiCembung = 35, LandaiCekung = 50, Shenti = 75, Smenyiap = 350, Lvmin = 50
 End If
ElseIf IndekX = 2 Then
   LandaiCembung = 35, LandaiCekung = 50, Shenti = 75, Smenyiap = 350, Lvmin = 50
ElseIf IndekX = 3 Then
 If IndekY = 0 Then
   LandaiCembung = 35, LandaiCekung = 50, Shenti = 75, Smenyiap = 350, Lvmin = 50
 Else
   LandaiCembung = 30, LandaiCekung = 44, Shenti = 55, Smenyiap = 250, Lvmin = 40
 End If
ElseIf IndekX = 4 Then
 If IndekY = 0 Then
   LandaiCembung = 22.5, LandaiCekung = 36, Shenti = 40, Smenyiap = 150, Lvmin = 35
 Else
   LandaiCembung = 18, LandaiCekung = 26, Shenti = 30, Smenyiap = 100, Lvmin = 25
 End If
ElseIf IndekX = 5 Then
   LandaiCembung = 18, LandaiCekung = 20, Shenti = 30, Smenyiap = 100, Lvmin = 25
End If
If A(n) >= 0 Then
 If Shenti >= LandaiCembung Then
   Lhenti(n) = (2 * Shenti) - ((200 * ((1.2 ^ 0.5 + 0.1 ^ 0.5) ^ 2)) / A(n))
   Lmenyiap(n) = (2 * Smenyiap) - ((200 * ((1.2 ^ 0.5 + 1.2 ^ 0.5) ^ 2)) / A(n))
 Else
   Lhenti(n) = (2 * Shenti) - ((A(n) * Shenti ^ 2) / (100 * ((2.4 ^ 0.5) + (0.2 ^ 0.5)) ^ 2))
   Lmenyiap(n) = (2 * Smenyiap) - ((A(n) * Smenyiap ^ 2) / (100 * ((2.4 ^ 0.5) + (2.4 ^ 0.5)) ^ 2))
 End If
 Lkenyamanan(n) = Vd * 3 * (1000 / 3600)
 Ldrainase(n) = 50 * A(n)
ElseIf A(n) < 0 Then
 If Shenti >= LandaiCekung Then
   Lhenti(n) = ((2 * Shenti) - (((120 + (3.5 * Shenti)) / (A(n) * -1))))
   Lmenyiap(n) = ((2 * Smenyiap) - (((120 + (3.5 * Smenyiap)) / (A(n) * -1))))
```

```
 Else
   Lhenti(n) = ((A(n) * Shenti ^ 2) / (120 + (3.5 * Shenti))) * -1
   Lmenyiap(n) = ((A(n) * Smenyiap ^ 2) / (120 + (3.5 * Smenyiap))) * -1
 End If
 Lkenyamanan(n) = ((A(n) * Vd ^ 2) / 380) * -1
 Ldrainase(n) = (50 * A(n)) * -1
End If
LhentiEX(n) = Lhenti(n) - Lvmin
LmenyiapEX(n) = Lmenyiap(n) - Lvmin
LkenyamananEX(n) = Lkenyamanan(n) - Lvmin
LdrainaseEX(n) = Ldrainase(n) - Lvmin
End Sub

Sub PilihJenisTikungan()
   SudutSpiral(n) = 90 * Lsmin(n) / (pi * Rd(n))
   SudutCircle(n) = SudutDefleksi(n) - 2 * SudutSpiral(n)
   Lc(n) = Rd(n) * KonvDegToRad * SudutCircle(n)
If e(n) < 0.03 Then
 'Tikungan Full Circle
 SudutSpiral(n) = 0
 SudutCircle(n) = SudutDefleksi(n)
 Tc(n) = Rd(n) * Tan(KonvDegToRad * SudutCircle(n) / 2)
 Ec(n) = Tc(n) * Tan(KonvDegToRad * SudutCircle(n) / 4)
 Lc(n) = Rd(n) * KonvDegToRad * SudutCircle(n)
 LCurve(n) = Lc(n)
Else
If Lc(n) > 20 Then
 'Tikungan SCS
 p(n) = Lsmin(n) ^ 2 / (6 * Rd(n)) - (Rd(n) * (1 - Cos(KonvDegToRad * SudutSpiral(n))))
 k(n) = Lsmin(n) - Lsmin(n) ^ 3 / (40 * Rd(n) ^ 2) - Rd(n) * Sin(KonvDegToRad * SudutSpiral(n))
 Ec(n) = (Rd(n) + p(n)) / Cos(KonvDegToRad * SudutDefleksi(n) / 2) - Rd(n)
 Tc(n) = (Rd(n) + p(n)) * Tan(KonvDegToRad * SudutDefleksi(n) / 2) + k(n)
 LCurve(n) = Lc(n)
Else
 'Tikungan Full Spiral
 SudutCircle(n) = 0
 Lc(n) = 0
 SudutSpiral(n) = SudutDefleksi(n) / 2
 Ls = SudutSpiral(n) * pi * Rd(n) / 90
 p(n) = Ls ^ 2 / (6 * Rd(n)) - (Rd(n) * (1 - Cos(KonvDegToRad * SudutSpiral(n))))
 k(n) = Ls - Ls ^ 3 / (40 * Rd(n) ^ 2) - Rd(n) * Sin(KonvDegToRad * SudutSpiral(n))
 Ec(n) = (Rd(n) + p(n)) / Cos(KonvDegToRad * SudutSpiral(n)) - Rd(n)
 Tc(n) = (Rd(n) + p(n)) * Tan(KonvDegToRad * SudutSpiral(n)) + k(n)
 LCurve(n) = 2 * Ls
End If
End If
End Sub

Sub HitungPanjangLengkungPeralihan()
Dim Ls1, Ls2, Ls3, I
'Syarat Kenyamanan
Ls1 = 3 * Vd / 3.6
'Syarat gaya sentrifugal
Ls2 = 2 * (Vd / 3.6) ^ 3 / Rd(n)
'Syarat landai relatif
Ls3 = m * b * (e(n) + en)
I = Ls1
If I < Ls2 Then I = Ls2
If I < Ls3 Then I = Ls3
If I Mod 5 = 0 Then
  Lsmin(n) = I
Else
```

```
  Lsmin(n) = 5 * (1 + Int(I / 5))
End If
End Sub

Sub HitungTingkatSuperelevasi()
Dim DoC, fs
DoC = KonvRdToD / Rd(n)
If DoC < Dpi Then
  fs = M0 * (DoC / Dpi) ^ 2 + S1 * DoC
Else
  fs = M0 * ((Dmak - DoC) / (Dmak - Dpi)) ^ 2 + h + S2 * (DoC - Dpi)
End If
e(n) = DoC * Vd ^ 2 / Ko - fs
End Sub

Sub HitungSudutDefleksi()
Dim Node, dx, dy, Sudut, I, delta
For Node = 1 To JumlahNode - 1
   dx = X(Node + 1) - X(Node)
   dy = Y(Node + 1) - Y(Node)
If dy = 0 Then
  Sudut = 90
Else
  Sudut = KonvRadToDeg * Atn(dx / dy)
End If
If dx <= 0 And dy <= 0 Then
  SudutJurusan(Node) = Sudut + 180
ElseIf dx <= 0 And dy >= 0 Then
  SudutJurusan(Node) = Sudut + 360
ElseIf dx >= 0 And dy < 0 Then
  SudutJurusan(Node) = Sudut + 180
ElseIf dx >= 0 And dy >= 0 Then
  SudutJurusan(Node) = Sudut
End If
  D(Node) = Sqr(dx ^ 2 + dy ^ 2)
Next Node
For I = 1 To JumlahNode - 2
   delta = SudutJurusan(I + 1) - SudutJurusan(I)
   Soc(I) = Sgn(delta)
If delta > 180 Then
  Soc(I) = -1
  SudutDefleksi(I) = 360 - delta
ElseIf delta < -180 Then
  Soc(I) = 1
  SudutDefleksi(I) = 360 + delta
Else
  SudutDefleksi(I) = Abs(delta)
End If
If Soc(I) = -1 Then
  Arah(I) = "Kiri"
Else
  Arah(I) = "Kanan"
End If
Next I
End Sub

Sub KoordinatPC()
Xpc(n) = X(n + 1) - Tc(n) * Sin(KonvDegToRad * SudutJurusan(n))
Ypc(n) = Y(n + 1) - Tc(n) * Cos(KonvDegToRad * SudutJurusan(n))
Xpt(n) = X(n + 1) + Tc(n) * Sin(KonvDegToRad * SudutJurusan(n + 1))
Ypt(n) = Y(n + 1) + Tc(n) * Cos(KonvDegToRad * SudutJurusan(n + 1))
End Sub
```

```
Sub ChekDataPI()
CmdAdd.Enabled = Len(txtAbsisPI.Text) > 0 And Len(txtOrdinatPI.Text) > 0
End If
End Sub
Sub CurrentChangeData()
ListId = lstAbsisPI.ListCount
JumlahNode = ListId + 2
X(ChangeId + 2) = Val(txtAbsisPI.Text)
Y(ChangeId + 2) = Val(txtOrdinatPI.Text)
Rd(ChangeId + 1) = Val(txtRadius.Text)
El(ChangeId + 2) = Val(txtElevasiPI.Text)
End Sub

Sub CurrentRemoveData()
ListId = lstAbsisPI.ListCount
JumlahNode = ListId + 2
For I = Ind + 1 To JumlahNode - 1
  X(I + 1) = X(I + 2)
  Y(I + 1) = Y(I + 2)
  Rd(I + 1) = Rd(I + 2)
  El(I + 1) = El(I + 2)
Next I
End Sub

Sub Deselect()
lstAbsisPI.Selected(Ind) = False
lstOrdinatPI.Selected(Ind) = False
lstRadius.Selected(Ind) = False
lstElevasiPI.Selected(Ind) = False
CmdRemove.Enabled = False
CmdChange.Enabled = False
End Sub

Sub Select_all_List()
lstAbsisPI.Selected(Ind) = True
lstOrdinatPI.Selected(Ind) = True
lstRadius.Selected(Ind) = True
lstElevasiPI.Selected(Ind) = True
CmdRemove.Enabled = True
CmdChange.Enabled = True
lblPI.Caption = "PI" & Ind + 1
End Sub

Sub CurrentAddData()
ListId = lstAbsisPI.ListCount
JumlahNode = ListId + 2
IndekX = CboKelasJalan.ListIndex
IndekY = CboKecepatan.ListIndex
Vd = Val(CboKecepatan.Text)
IndekL = Val(txtLajur.Text)
StaPI(0) = Val(txtStasiun.Text)
Lajur = Val(txtLajur.Text)
'Koordinat Titik
X(1) = Val(txtAbsisAwal.Text)
Y(1) = Val(txtOrdinatAwal.Text)
El(I) = Val(txtElevasiAwal.Text)
X(JumlahNode) = Val(txtAbsisAkhir.Text)
Y(JumlahNode) = Val(txtOrdinatAkhir.Text)
El(JumlahNode) = Val(txtElevasiAkhir.Text)
If txtRadius <> "" Then
  X(ListId + 1) = Val(txtAbsisPI.Text)
  Y(ListId + 1) = Val(txtOrdinatPI.Text)
```

```
  Rd(ListId) = Val(txtRadius.Text)
  El(ListId + 1) = Val(txtElevasiPI.Text)
End If
End Sub

Sub GambarKoordinat()
Dim cir
PicAlinemen.Cls
UpdateSkala
PicAlinemen.Scale (SkalaMin - 100, SkalaMak + 100)-(SkalaMak + 100, SkalaMin - 100)
For cir = 1 To JumlahNode
   PicAlinemen.Circle (X(cir), Y(cir)), 15
Next cir
For n = 1 To JumlahNode - 1
   PicAlinemen.CurrentX = X(1)
   PicAlinemen.CurrentY = Y(1)
   PicAlinemen.Print "Awal Proyek"
   PicAlinemen.CurrentX = X(JumlahNode)
   PicAlinemen.CurrentY = Y(JumlahNode)
   PicAlinemen.Print "Akhir Proyek"
   PicAlinemen.Line (X(n), Y(n))-(X(n + 1), Y(n + 1))
 If n < JumlahNode - 1 Then PicAlinemen.Print "PI" & n
Next n
End Sub

Function WriteDataText()
Dim nd
DataText = DataText & "DATA PERENCANAAN GEOMETRIK" & vbCrLf
DataText = DataText & "Kelas Jalan" & vbTab & " = "
DataText = DataText & FrmData.CboKelasJalan.Text & vbCrLf
DataText = DataText & "Kecepatan" & vbTab & " = "
DataText = DataText & FrmData.CboKecepatan.Text & "km/jam" & vbCrLf
DataText = DataText & "Koordinat Awal" & vbTab & " = ( "
DataText = DataText & X(1) & "," & Y(1) & " )" & vbCrLf
DataText = DataText & "Elevasi Awal" & vbTab & " = "
DataText = DataText & El(1) & " m " & vbCrLf
DataText = DataText & "Koordinat Akhir" & vbTab & " = ( "
DataText = DataText & X(JumlahNode) & "," & Y(JumlahNode) & " ) " & vbCrLf
DataText = DataText & "Elevasi Akhir" & vbTab & " = "
DataText = DataText & El(JumlahNode) & " m " & vbCrLf
For nd = 1 To JumlahNode - 2
  DataText = DataText & "Koordinat PI" & nd & vbTab & " = ( "
  DataText = DataText & X(nd + 1) & "," & Y(nd + 1) & " ) " & vbCrLf
Next nd
For txt = 1 To JumlahNode - 2
 If SudutCircle(txt) <> 0 And SudutSpiral(txt) = 0 Then
   DataText = DataText & vbCrLf
   DataText = DataText & "#CURVE DESIGN PI" & txt & vbCrLf
   DataText = DataText & "Tipe Tikungan" & vbTab & " = "
   DataText = DataText & "Full Circle" & vbCrLf
 Call DTextCurve
   DataText = DataText & ">>Stationing" & vbCrLf
   DataText = DataText & "StationPI" & txt & vbTab & " = "
   DataText = DataText & Sta(StaPI(txt)) & vbCrLf
   DataText = DataText & "StationTC" & txt & vbTab & " = "
   DataText = DataText & Sta(StaPI(txt) - Tc(txt)) & vbCrLf
   DataText = DataText & "StationCT" & txt & vbTab & " = "
   DataText = DataText & Sta(StaPI(txt) - Tc(txt) + Lc(txt)) & vbCrLf
 Call DTextSuperelevasi
 Call DTextPelebaran
 Call DTextAlinemenVertikal
 ElseIf SudutCircle(txt) <> 0 And SudutSpiral(txt) <> 0 Then
```

```
    DataText = DataText & vbCrLf
    DataText = DataText & "#CURVE DESIGN PI" & txt & vbCrLf
    DataText = DataText & "Tipe Tikungan" & vbTab & " = "
    DataText = DataText & "Spiral Circle Spiral" & vbCrLf
  Call DTextCurve
  Call DTextPK
    DataText = DataText & ">>Stationing" & vbCrLf
    DataText = DataText & "StationPI" & txt & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt)) & vbCrLf
    DataText = DataText & "StationTS" & txt & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt) - Tc(txt)) & vbCrLf
    DataText = DataText & "StationSC" & txt & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt) - Tc(txt) + Lsmin(txt)) & vbCrLf
    DataText = DataText & "StationCS" & txt & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt) - Tc(txt) + Lsmin(txt) + Lc(txt)) & vbCrLf
    DataText = DataText & "StationST" & txt & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt) - Tc(txt) + 2 * Lsmin(txt) + Lc(txt)) & vbCrLf
  Call DTextSuperelevasi
  Call DTextPelebaran
  Call DTextAlinemenVertikal
  ElseIf SudutCircle(txt) = 0 And SudutSpiral(txt) <> 0 Then
    DataText = DataText & vbCrLf
    DataText = DataText & "#CURVE DESIGN PI" & txt & vbCrLf
    DataText = DataText & "Tipe Tikungan" & vbTab & " = "
    DataText = DataText & "Spiral Spiral" & vbCrLf
  Call DTextCurve
  Call DTextPK
    DataText = DataText & ">>Stationing" & vbCrLf
    DataText = DataText & "StationPI" & txt & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt)) & vbCrLf
    DataText = DataText & "StationTS" & txt & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt) - Tc(txt)) & vbCrLf
    DataText = DataText & "StationST" & txt & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt) - Tc(txt) + 2 * Lsmin(txt)) & vbCrLf
  Call DTextSuperelevasi
  Call DTextPelebaran
  Call DTextAlinemenVertikal
  End If
Next txt
DataText = DataText & vbCrLf
DataText = DataText & "~Panjang Trace Jalan = "
DataText = DataText & Fixed(StaPI(JumlahNode - 1)) & vbCrLf
DataText = DataText & "~Kontrol Overlapping = Oke !!" & vbCrLf
End Function

Sub DTextSuperelevasi()
    DataText = DataText & ">>Diagram Superelevasi" & vbCrLf
    DataText = DataText & "Superelevasi" & vbTab & vbTab & " = "
    DataText = DataText & 100 * Fixed(e(txt)) & "%" & vbCrLf
    DataText = DataText & "Awal Superelevasi" & vbTab & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt) - Tc(txt))
Call DTextENormal
    DataText = DataText & "Awal Superelevasi  Penuh" & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt) + Lsmin(txt) - Tc(txt))
Call DTextEPenuh
    DataText = DataText & "Akhir Superelevasi Penuh" & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt) - Tc(txt) + Lc(txt) - Lsmin(txt))
Call DTextEPenuh
    DataText = DataText & "Akhir Superelevasi" & vbTab & vbTab & " = "
    DataText = DataText & Sta(StaPI(txt) - Tc(txt) + Lc(txt))
Call DTextENormal
End Sub
```

```
Sub DTextPelebaran()
   DataText = DataText & ">>Pelebaran Perkerasan" & vbCrLf
   DataText = DataText & "Pelebaran Perkerasan" & vbTab & " = "
   DataText = DataText & Fixed(Pelebaran(txt)) & " m " & vbCrLf
If Pelebaran(txt) <> 0 Then
   DataText = DataText & "Posisi Pelebaran" & vbTab & vbTab & " = "
   DataText = DataText & Arah(txt) & vbCrLf
   DataText = DataText & "Awal Pelebaran" & vbTab & vbTab & " = Sta. "
   DataText = DataText & Sta(StaPI(txt)) & vbCrLf
   DataText = DataText & "Awal Pelebaran Penuh" & vbTab & " = Sta. "
   DataText = DataText & Sta(StaPI(txt) + Lsmin(txt)) & " ("
   DataText = DataText & Arah(txt) & " +" & Fixed(Pelebaran(txt)) & " m)" & vbCrLf
   DataText = DataText & "Akhir Pelebaran Penuh" & vbTab & " = Sta. "
   DataText = DataText & Sta(StaPI(txt) + Tc(txt)) & " ("
   DataText = DataText & Arah(txt) & " +" & Fixed(Pelebaran(txt)) & " m)" & vbCrLf
   DataText = DataText & "Akhir Pelebaran" & vbTab & vbTab & " = Sta. "
   DataText = DataText & Sta(StaPI(txt) + Lc(txt)) & vbCrLf
End If
   DataText = DataText & ">>Jarak Pandang Horisontal" & vbCrLf
   DataText = DataText & "Kebebasan Samping" & vbTab & " = "
   DataText = DataText & Fixed(M_Curve(txt)) & " m " & vbCrLf
End Sub

Sub DTextCurve()
   DataText = DataText & "Sudut Defleksi" & vbTab & " = "
   DataText = DataText & SudutDefleksi(txt) & vbCrLf
   DataText = DataText & "Radius Tikungan" & vbTab & " = "
   DataText = DataText & Rd(txt) & " m" & vbCrLf
   DataText = DataText & " Design Speed" & vbTab & " = "
   DataText = DataText & Vd & " km/jam" & vbCrLf
   DataText = DataText & " Arah Tikungan" & vbTab & " = "
   DataText = DataText & Arah(txt) & vbCrLf
   DataText = DataText & " Panjang Ls" & vbTab & " = "
   DataText = DataText & Fixed(Lsmin(txt)) & " m" & vbCrLf
   DataText = DataText & " Panjang Lc" & vbTab & " = "
   DataText = DataText & Fixed(Lc(txt)) & " m" & vbCrLf
   DataText = DataText & " Panjang Tc" & vbTab & " = "
   DataText = DataText & Fixed(Tc(txt)) & " m" & vbCrLf
   DataText = DataText & " Panjang Ec" & vbTab & " = "
   DataText = DataText & Fixed(Ec(txt)) & " m" & vbCrLf
End Sub

Sub DTextPK()
DataText = DataText & " Panjang p" & vbTab & " = "
DataText = DataText & Fixed(p(txt)) & " m" & vbCrLf
DataText = DataText & " Panjang k" & vbTab & " = "
DataText = DataText & Fixed(k(txt)) & " m" & vbCrLf
End Sub

Sub DTextEPenuh()
If Soc(txt) > 0 Then
DataText = DataText & " (Ki +" & 100 * Fixed(e(txt)) & "%, Ka -" & 100 * Fixed(e(txt)) & "%)" & vbCrLf
Else
DataText = DataText & " (Ki -" & 100 * Fixed(e(txt)) & "%, Ka +" & 100 * Fixed(e(txt)) & "%)" & vbCrLf
End If
End Sub

Sub DTextENormal()
   DataText = DataText & " (Ki -" & 100 * Fixed(en) & "%, Ka -" & 100 * Fixed(en) & "%)" & vbCrLf
End Sub
```

```
Sub HitungPelebaran()
Dim Ri, Rw, Bw, Td, Zn, bt, sigma
Ri = Sqr(Rd(n) ^ 2 - 64) - 1.25
Rw = Sqr((Ri + 2.5) ^ 2 + 64)
Bw = Rw - Ri
Td = Sqr(Rd(n) ^ 2 + 21.75) - Rd(n)
Zn = 0.105 * Vd / Sqr(Rd(n))
bt = JLajur * (Bw + 0.8) + (JLajur - 1) * Td + Zn
sigma = bt - JLajur * b
If sigma < 0 Then
Pelebaran(n) = 0
Else
Pelebaran(n) = sigma
End If
End Sub

Sub HitungM_Curve()
Dim SSD, Sudut
SSD = ((Vh * 2.5) * 0.278) + ((Vh ^ 2)) / (254 * fm)
Sudut = (90 * SSD) / (pi * Rd(n))
M_Curve(n) = Rd(n) * (1 - Cos(KonvDegToRad * Sudut))
End Sub

Sub Validasi(DText)
Dim SText As String
Dim VText, FText
VText = Val(DText)
FText = Format(DText, "0.###")
SText = Format(VText, "#0.###")
 If SText <> FText Then
  RecordValidasi = False
  MsgBox "This is not a valid number", _
  vbOKOnly + vbExclamation, "Error Message"
 Else
  RecordValidasi = True
 End If
End Sub

Private Sub txtRadius_LostFocus()
If txtRadius.Text <> "" Then
  Validasi (txtRadius.Text)
 If RecordValidasi = False Then
    txtRadius.Text = ""
    txtRadius.SetFocus
 End If
End If
End Sub

Sub DTextAlinemenVertikal()
DataText = DataText & vbCrLf
DataText = DataText & "#VERTIKAL DESIGN PPV" & txt & vbCrLf
DataText = DataText & "Elevasi" & vbTab & vbTab & " = " & El(txt + 1) & " m " & vbCrLf
If txt - 1 = 0 Then
   DataText = DataText & "Jarak dari awal proyek ke PI" & txt & " = " & Fixed(D(txt)) & " m" & vbCrLf
Else
   DataText = DataText & "Jarak dari PI" & txt - 1 & " ke PI" & txt & " = " & Fixed(D(txt)) & " m" & vbCrLf
End If
DataText = DataText & "L menurut JPH" & vbTab & " = " & Fixed(Lhenti(txt)) & vbCrLf
DataText = DataText & "L menurut JPM" & vbTab & " = " & Fixed(Lmenyiap(txt)) & vbCrLf
DataText = DataText & "L (Drainase)" & vbTab & " = " & Fixed(Ldrainase(txt)) & vbCrLf
DataText = DataText & "L (Kenyamanan)" & vbTab & " = " & Fixed(Lkenyamanan(txt)) & vbCrLf
DataText = DataText & "Lv terpakai" & vbTab & " = " & Fixed(Lvterpakai(txt)) & vbCrLf
```

```
DataText = DataText & "Ev" & vbTab & vbTab & " = " & Fixed(Ev(txt)) & vbCrLf
DataText = DataText & ">>Stationing" & vbCrLf
DataText = DataText & "Station PPV" & txt & vbTab & " = " & Sta(StaPI(txt)) & vbCrLf
DataText = DataText & "Station PLV" & txt & vbTab & " = " & Sta(StaPI(txt) - (0.5 * Lvterpakai(txt))) &
vbCrLf
DataText = DataText & "Station PTV" & txt & vbTab & " = " & Sta(StaPI(txt) + (0.5 * Lvterpakai(txt))) &
vbCrLf
If A(txt) > 0 Then
  DataText = DataText & "Elevasi PPV" & txt & " pada cembung" & vbTab & " = " & Fixed(El(txt + 1) +
Ev(txt)) & vbCrLf
Else
  DataText = DataText & "Elevasi PPV" & txt & " pada cekung" & vbTab & " = " & Fixed(El(txt + 1) +
Ev(txt)) & vbCrLf
End If
End Sub

Private Sub txtStasiun_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 And txtStasiun.Text <> "" Then
  Validasi (txtStasiun.Text)
 If RecordValidasi = False Then
  txtStasiun.Text = ""
  txtStasiun.SetFocus
 Else
  txtAbsisAwal.SetFocus
 End If
End If
End Sub

Private Sub txtStasiun_LostFocus()
If txtStasiun.Text <> "" Then
  Validasi (txtStasiun.Text)
 If RecordValidasi = False Then
   txtStasiun.Text = ""
   txtStasiun.SetFocus
 End If
End If
End Sub
```

FORM CURVE

```
General Declarations
Dim id As Integer

Private Sub cboCurve_Change()
  id = CboCurve.ListIndex + 1
  ShowData
  PicCurve.Cls
  PicCurve.AutoRedraw = True
  PaintCurve
End Sub

Private Sub cboCurve_Click()
  id = CboCurve.ListIndex + 1
  ShowData
  With PicCurve
  .Cls
  .AutoRedraw = True
  End With
  PaintCurve
End Sub
```

```
Private Sub Form_Activate()
CboCurve.Clear
 For cbo = 1 To JumlahNode - 2
  CboCurve.AddItem "Curve Design pada PI " & cbo
 Next cbo
End Sub

Private Sub Form_Load()
 Me.WindowState = vbMaximized
 With FraDataPI
  .Top = 0
  .Left = 0
  .Height = Me.ScaleHeight
 End With
 With PicCurve
  .Top = 0
  .Left = FraDataPI.Width
  .Height = Me.ScaleHeight
  .Width = Me.ScaleWidth - FraDataPI.Width
 End With
End Sub

Sub ShowData()
   If SudutCircle(id) <> 0 And SudutSpiral(id) = 0 Then
    FraCC.Visible = True
    FraSS.Visible = False
    FraSCS.Visible = False
    DataCC
   ElseIf SudutCircle(id) <> 0 And SudutSpiral(id) <> 0 Then
    FraSCS.Visible = True
    FraCC.Visible = False
    FraSS.Visible = False
    DataSCS
   ElseIf SudutCircle(id) = 0 And SudutSpiral(id) <> 0 Then
    FraSS.Visible = True
    FraSCS.Visible = False
    FraCC.Visible = False
    DataSS
   End If
End Sub

Sub PaintCurve()
 xc = Tc(id) * Cos(KonvDegToRad * SudutDefleksi(id) / 2)
 yc = Tc(id) * Sin(KonvDegToRad * SudutDefleksi(id) / 2) - (Ec(id) + Rd(id))
 Yf = Tc(id) * Sin(KonvDegToRad * SudutDefleksi(id) / 2)
 skala = 2 * xc
 If skala < Yf Then skala = Yf
 PicCurve.Scale (0, Soc(id) * skala)-(skala, 0)
 PicCurve.Scale (-10, Yf + 10)-(2 * xc + 10, -2 * xc + 10 + Yf)
 'Menggambar Tangen
 PicCurve.Line (0, 0)-(xc, Soc(id) * (yc + Ec(id) + Rd(id)))
 PicCurve.Line (xc, Soc(id) * (yc + Ec(id) + Rd(id)))-(2 * xc, 0)
 PicCurve.CurrentX = xc
 PicCurve.CurrentY = Soc(id) * (yc + Ec(id) + Rd(id))
 PicCurve.Print "PI"
 'Menggambar Lengkung Peralihan
 X1 = 0: Y1 = 0
 If SudutSpiral(id) <> 0 Then
  PicCurve.CurrentX = X1
  PicCurve.CurrentY = Y1
  PicCurve.Print "TS"
  PicCurve.CurrentX = 2 * xc
```

```
    PicCurve.CurrentY = Y1
    For spiral = 0.1 To Lsmin(id) Step 0.1
      Xs = spiral - spiral ^ 5 / (40 * Rd(id) ^ 2 * Lsmin(id) ^ 2)
      Ys = spiral ^ 3 / (6 * Lsmin(id) * Rd(id))
      alpha = KonvRadToDeg * Atn(Ys / Xs)
      z = Sqr(Xs ^ 2 + Ys ^ 2)
      Sudut = SudutDefleksi(id) / 2 - alpha
      X2 = z * Cos(KonvDegToRad * Sudut)
      Y2 = Soc(id) * (z * Sin(KonvDegToRad * Sudut))
      PicCurve.Line (X1, Y1)-(X2, Y2)
      PicCurve.Line (2 * xc - X1, Y1)-(2 * xc - X2, Y2)
      X1 = X2: Y1 = Y2
    Next spiral
    If SudutCircle(id) <> 0 Then
      PicCurve.CurrentX = X1
      PicCurve.CurrentY = Y1
      PicCurve.Print "SC"
      PicCurve.CurrentX = 2 * xc - X1
      PicCurve.CurrentY = Y1
      PicCurve.Print "CS"
     Else
      PicCurve.CurrentX = X1
      PicCurve.CurrentY = Y1
      PicCurve.Print "SS"
    End If
   End If
   'Menggambar Lengkung Melingkar
   Xs = X1: Ys = Y1
   If SudutCircle(id) <> 0 Then
    For busur = 0.1 To Lc(id) Step 0.1
      SudutTheta = KonvRadToDeg * busur / 2 - SudutTheta
      z = 2 * Rd(id) * Sin(KonvDegToRad * SudutTheta)
      SudutGlobal = SudutCircle(id) / 2 - SudutTheta
      X2 = Xs + z * Cos(KonvDegToRad * SudutGlobal)
      Y2 = Ys + Soc(id) * (z * Sin(KonvDegToRad * SudutGlobal))
      PicCurve.Line (X1, Y1)-(X2, Y2)
      X1 = X2: Y1 = Y2
    Next busur
    If SudutSpiral(id) = 0 Then
      PicCurve.CurrentX = 0
      PicCurve.CurrentY = 0
      PicCurve.Print "TC"
      PicCurve.CurrentX = 2 * xc
      PicCurve.CurrentY = 0
      PicCurve.Print "CT"
    End If
  End If
End Sub

Sub DataSCS()
    LblType1.Caption = "SCS"
    LblStaPI1.Caption = Sta(StaPI(id))
    LblStaTs1.Caption = Sta(StaPI(id) - Tc(id))
    LblStaSC1.Caption = Sta(StaPI(id) - Tc(id) + Lsmin(id))
    LblStaCS1.Caption = Sta(StaPI(id) - Tc(id) + Lsmin(id) + Lc(id))
    LblStaST1.Caption = Sta(StaPI(id) - Tc(id) + 2 * Lsmin(id) + Lc(id))
    LblRd1.Caption = Fixed(Rd(id))
    LblDelta1.Caption = Fixed(SudutDefleksi(id))
    LblSuper1.Caption = Fixed(e(id))
    LblLs1.Caption = Fixed(Lsmin(id))
    LblLc1.Caption = Fixed(Lc(id))
    LblTs1.Caption = Fixed(Tc(id))
```

```
    Lblp1.Caption = Fixed(p(id))
    LblK1.Caption = Fixed(k(id))
End Sub

Sub DataCC()
    LblType2.Caption = "CC"
    LblStaPI2.Caption = Sta(StaPI(id))
    LblstaTC2.Caption = Sta(StaPI(id) - Tc(id))
    LblStaCT2.Caption = Sta(StaPI(id) - Tc(id) + Lc(id))
    LblRd2.Caption = Fixed(Rd(id))
    LblDelta2.Caption = Fixed(SudutDefleksi(id))
    LblSuper2.Caption = Fixed(e(id))
    LblLs2.Caption = Fixed(Lsmin(id))
    LblLc2.Caption = Fixed(Lc(id))
    LblTs2.Caption = Fixed(Tc(id))
End Sub

Sub DataSS()
    LblType3.Caption = "SS"
    LblStaPI3.Caption = Sta(StaPI(id))
    LblStaTs3.Caption = Sta(StaPI(id) - Tc(id))
    LblStaST3.Caption = Sta(StaPI(id) - Tc(id) + 2 * Lsmin(id))
    LblRd3.Caption = Fixed(Rd(id))
    LblDelta3.Caption = Fixed(SudutDefleksi(id))
    LblSuper3.Caption = Fixed(e(id))
    LblLs3.Caption = Fixed(Lsmin(id))
    LblTs3.Caption = Fixed(Tc(id))
    Lblp3.Caption = Fixed(p(id))
    LblK3.Caption = Fixed(k(id))
End Sub

Private Sub Form_Resize()
With FraDataPI
  .Top = 0
  .Left = 0
  .Height = Me.ScaleHeight
End With
With PicCurve
  .Top = 0
  .Left = FraDataPI.Width
  .Height = Me.ScaleHeight
   lebar = Me.ScaleWidth - FraDataPI.Width
   If lebar > 0 Then
     .Width = lebar
   End If
End With
End Sub
```

FORM TRACE

```
Private Sub Form_Activate()
PaintCoordinate
PaintAlinyemen
End Sub

Private Sub Form_Load()
Me.WindowState = vbMaximized
With picTrace
    .Left = 0
    .Top = 0
    .Height = Me.ScaleHeight
```

```
    .Width = Me.ScaleWidth
End With
End Sub

Private Sub Form_Resize()
With picTrace
    .Left = 0
    .Top = 0
    .Height = Me.ScaleHeight
    .Width = Me.ScaleWidth
End With
End Sub

Sub PaintCoordinate()
picTrace.Cls
UpdateSkala
picTrace.Scale (SkalaMin - 100, SkalaMak + 100)-(SkalaMak + 100, SkalaMin - 100)
For cir = 1 To JumlahNode
    picTrace.Circle (X(cir), Y(cir)), 8
Next cir
For n = 1 To JumlahNode - 1
    With picTrace
        .CurrentX = X(1)
        .CurrentY = Y(1)
        picTrace.Print "Awal Proyek" & "(" & X(1) & "," & Y(1) & ")" & " - " & El(1) & "m"
        .CurrentX = X(JumlahNode)
        .CurrentY = Y(JumlahNode)
        picTrace.Print "AkhirProyek" & "(" & X(JumlahNode) & "," & Y(JumlahNode) & ")" & " - " &
El(JumlahNode) & "m"
    End With
    picTrace.Line (X(n), Y(n))-(X(n + 1), Y(n + 1))
    If n < JumlahNode - 1 Then
    picTrace.Print "  PI" & n & "(" & X(n + 1) & "," & Y(n + 1) & ")" & " - " & El(n + 1) & "m"
    End If
Next n
End Sub

Sub PaintAlinyemen()
For n = 1 To JumlahNode - 2
X1 = Xpc(n)
Y1 = Ypc(n)
X3 = Xpt(n)
Y3 = Ypt(n)
picTrace.Circle (X1, Y1), 8
picTrace.Circle (X3, Y3), 8
If SudutSpiral(n) <> 0 Then
    For spiral = 0.1 To Lsmin(n) Step 0.1
        Xs = spiral - spiral ^ 5 / (40 * Rd(n) ^ 2 * Lsmin(n) ^ 2)
        Ys = spiral ^ 3 / (6 * Lsmin(n) * Rd(n))
        SudutTheta = KonvRadToDeg * Atn(Ys / Xs)
        z = Sqr(Xs ^ 2 + Ys ^ 2)
        'Lskiri
        SudutGlobal1 = 90 - SudutJurusan(n) - Soc(n) * SudutTheta
        Zxi = z * Cos(KonvDegToRad * SudutGlobal1)
        Zyi = z * Sin(KonvDegToRad * SudutGlobal1)
        X2 = Xpc(n) + Zxi
        Y2 = Ypc(n) + Zyi
        picTrace.Line (X1, Y1)-(X2, Y2)
        X1 = X2: Y1 = Y2
        'Ls Kanan
        SudutGlobal2 = 90 - SudutJurusan(n + 1) - Soc(n) * SudutTheta
        Zxj = z * Cos(KonvDegToRad * SudutGlobal2)
```

```
        Zyj = z * Sin(KonvDegToRad * SudutGlobal2)
        X4 = Xpt(n) - Zxj
        Y4 = Ypt(n) - Zyj
        picTrace.Line (X3, Y3)-(X4, Y4)
        X3 = X4: Y3 = Y4
    Next spiral
    picTrace.Circle (X1, Y1), 8
End If
'menggambar lengkung melingkar
Xs = X1: Ys = Y1
If SudutCircle(n) <> 0 Then
    For busur = 0.1 To Lc(n) Step 0.1
        SudutTheta = KonvRadToDeg * busur / (2 * Rd(n))
        z = 2 * Rd(n) * Sin(KonvDegToRad * SudutTheta)
        SudutGlobal = 90 - SudutJurusan(n) - Soc(n) * (SudutTheta + SudutSpiral(n))
        X2 = Xs + (z * (Cos(KonvDegToRad * SudutGlobal)))
        Y2 = Ys + (z * (Sin(KonvDegToRad * SudutGlobal)))
        picTrace.Line (X1, Y1)-(X2, Y2)
        X1 = X2: Y1 = Y2
    Next busur
    picTrace.Circle (X1, Y1), 8
End If
Next n
End Sub
```

FRM OUTPUT

```
Private Sub Form_Load()
Me.WindowState = vbMaximized
    With TxtOutput
      .Left = 15
      .Top = 15
      .Height = Me.ScaleHeight
      .Width = Me.ScaleWidth
    End With
Me.Caption = TxtOutput.Height
End Sub

Private Sub Form_Resize()
  With TxtOutput
    .Left = 0
    .Top = 0
    .Height = Me.ScaleHeight
    .Width = Me.ScaleWidth
  End With
End Sub
```

INDEKS