

การใช้ Visual Basic ติดต่อกับ Soft Ware OLE Process Control

จากบทที่ผ่านมาเราได้ทราบขั้นตอนและวิธีการใช้งานโปรแกรม Kepware V4.0 มาแล้ว และเราสามารถนำ Quick Client ในตัว Kepware ทดสอบการเข้าถึงข้อมูลในชุดควบคุมได้แล้ว แต่ข้อจำกัดของ Kepware คือ ไม่สามารถสร้างกราฟฟิกเพื่อใช้ติดต่อกับผู้ใช้โปรแกรมได้

ลำดับต่อจากนี้ไปจะขอแนะนำการใช้โปรแกรม Visual Basic ติดต่อกับ Kepware เพื่อสามารถเข้าถึงแหล่งข้อมูลของชุดควบคุม และสามารถนำข้อมูลมาแสดงผลเป็นกราฟฟิกในรูปแบบต่าง ๆ ได้

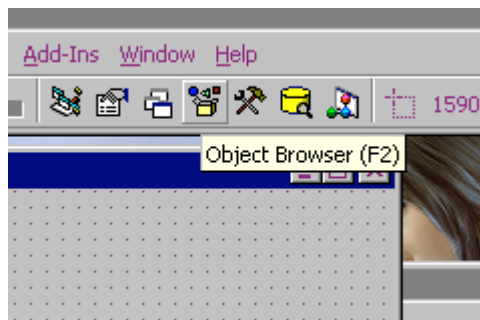
อย่างไรก็ตามก่อนที่เราจะเริ่มพัฒนาโปรแกรมงานดังกล่าวข้างต้นผู้เขียนขอให้เราทำความเข้าใจกับองค์ความรู้ที่เกี่ยวข้องซึ่งเล็กน้อยดังต่อไปนี้

8.1 OLE “ Object Linking and Embedding “ ซึ่งหมายถึงความสามารถในการเชื่อมโยงข้อมูลหรือการทำงานระหว่าง Application เข้าด้วยกัน เมื่อกกล่าวถึง OLE เราจะแบ่งออกเป็นสองส่วนคือ ส่วนของ Object Linking และ ส่วนของ Object Embedding

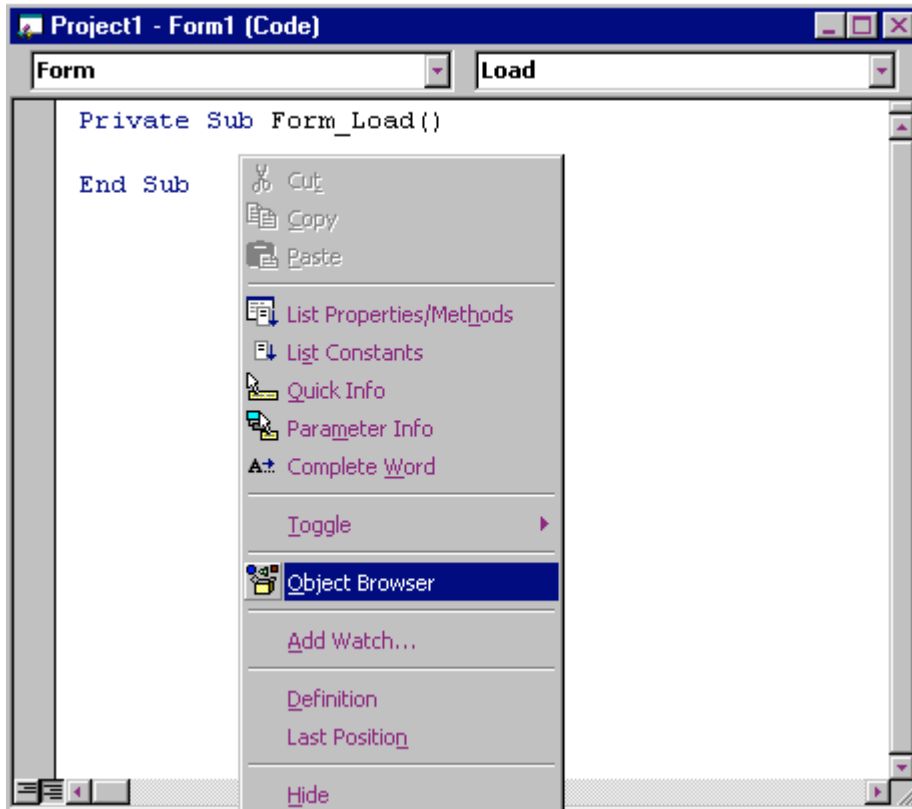
Object Linking หมายถึงความสามารถในการใช้ข้อมูลร่วมกันระหว่าง ผู้ใช้ เมื่อผู้ใช้คนหนึ่งมีการเปลี่ยนแปลงก็จะส่งผลให้ผู้ใช้อื่นได้รับข้อมูลที่ถูกแก้ไขด้วย ส่วนของ object Embedding เป็นความสามารถในการใช้ข้อมูลร่วมกันระหว่าง Application

8.1.1 Expose Object โดยปกติ Application บน Windows จะแบ่งฟังก์ชันการทำงานด้านต่าง ๆ ออกเป็นส่วน ๆ ซึ่งแต่ละส่วนจะถูกเรียกว่า Object เช่น Micro soft Word จะแบ่งเป็น Object ที่เกี่ยวกับการจัดการ File ซึ่งทำหน้าที่เกี่ยวกับการ ปิด เปิด หรือ บันทึก File เป็นต้น บาง Object ก็เป็นฟังก์ชันภายในนั้นคือ Application อื่นไม่สามารถเรียกใช้ได้ Object ประเภทนี้เรียกว่า Private Object แต่ถ้า Object ที่ Application อื่นสามารถเรียกใช้ได้ในรูปแบบของ OLE เราจะเรียก Object ประเภทนี้ว่า Exposed Object เช่น Object ที่ใช้สำหรับการตรวจค่าของ Micro Soft Word เป็นต้น

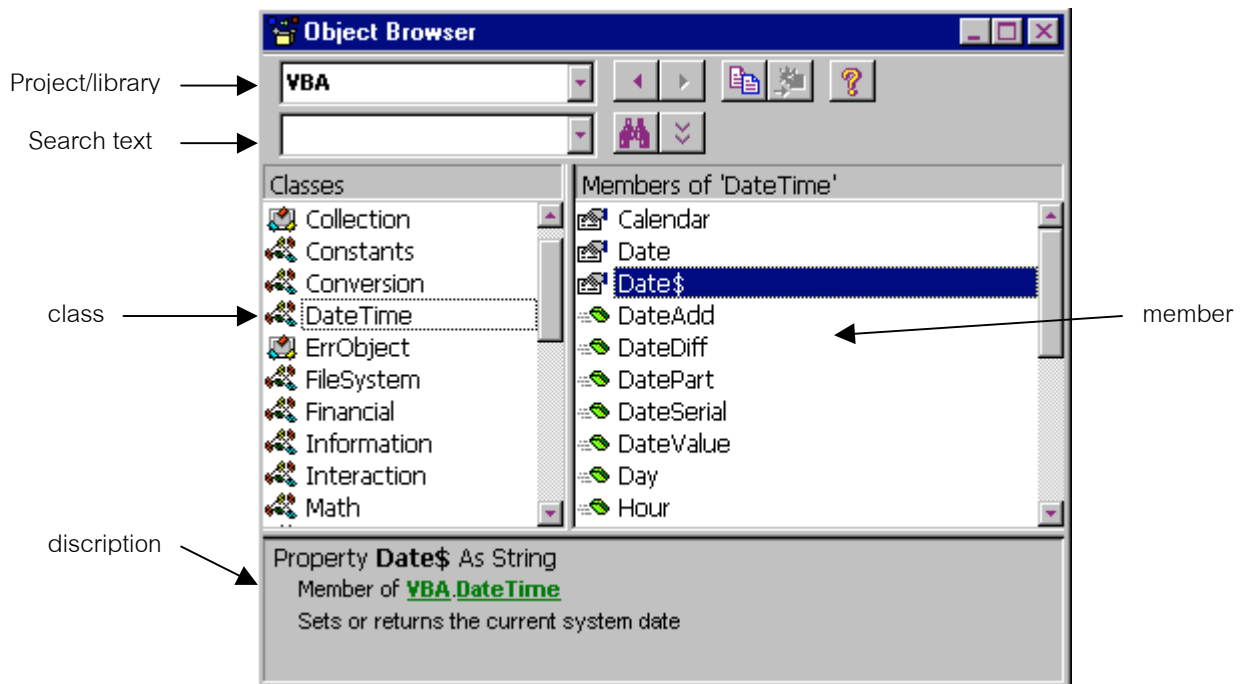
ในการเรียกใช้ Exposed Object เราจะต้องทราบโครงสร้างของ Exposed Object ว่าประกอบด้วย Class อะไรบ้างและมีวิธีการเรียกใช้อย่างไร ซึ่งสามารถเรียกดูได้จาก Object Browser



รูป 8.1 การเรียก Object Browser จาก ICON



รูป 8.2 เรียก Object Browser จาก Editor



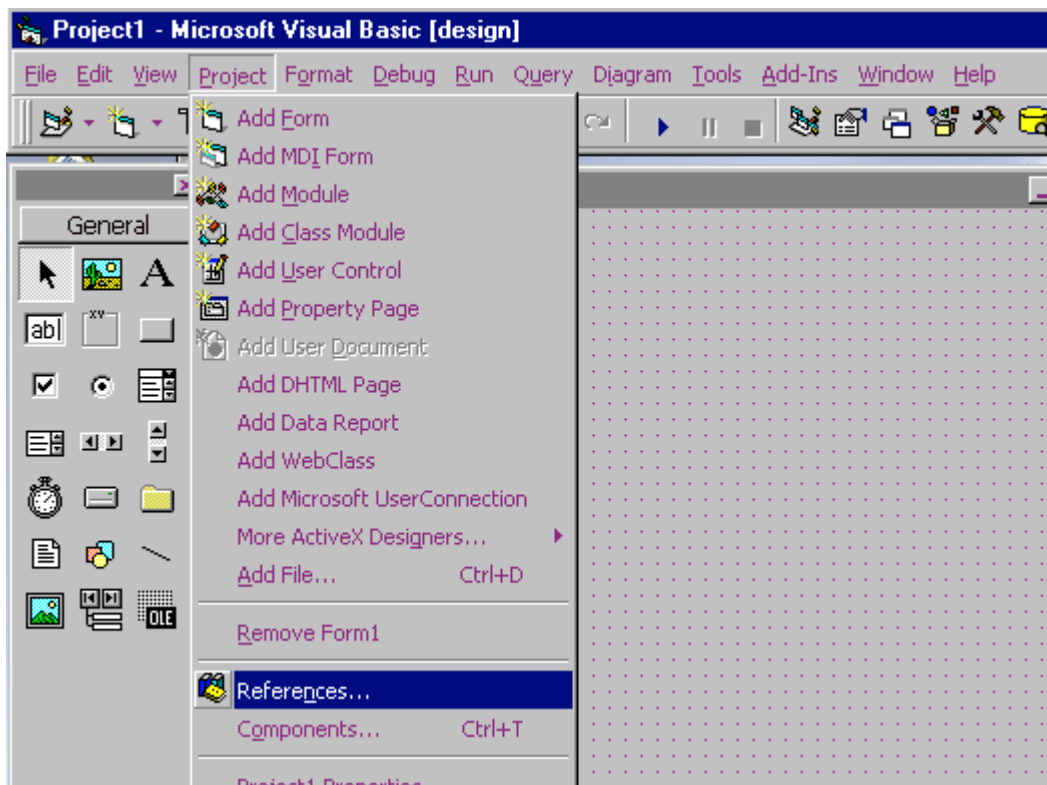
รูป 8.3 หน้าต่าง Object Browser

รายละเอียดจากรูป 8.3

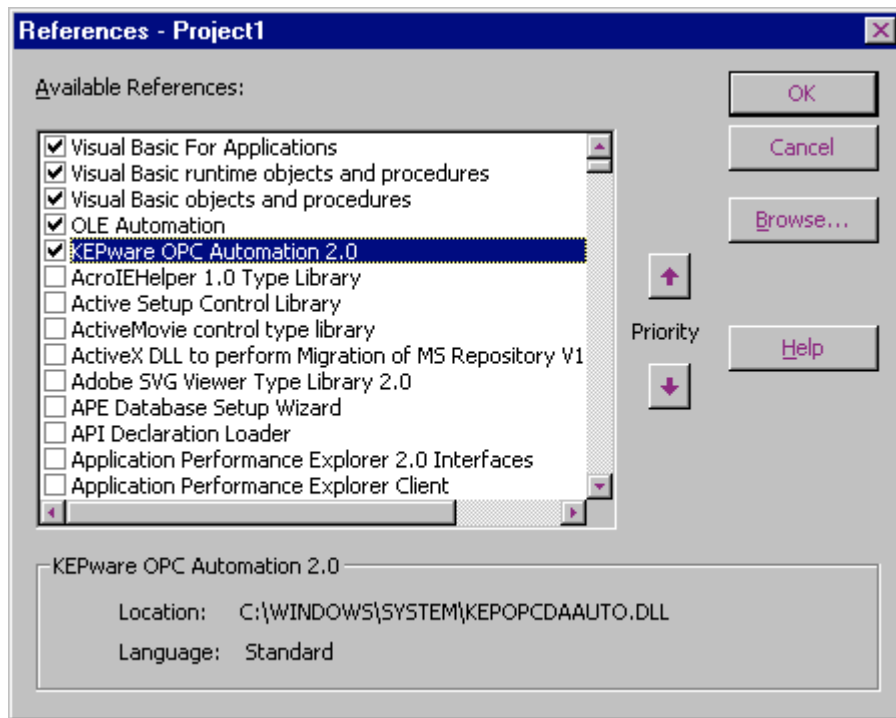
- Project/Libraries หมายถึงทุก ๆ Libraries
 - Project1 Library ที่เก็บรายละเอียดโปรแกรมปัจจุบัน
 - Stdole Library ที่เก็บ Property,Event,Method และ ค่าคงที่ ที่ใช้ทำ OLE Automation
 - VB Library ที่เก็บ Property,Event,Method และ ค่าคงที่ ที่ใช้สร้างโปรแกรมทั่ว ๆ ไป
 - VBA Library ที่เก็บ Property,Event,Method และ ค่าคงที่ ที่ใช้เชื่อมโยงกับ Application อื่น
 - VB RUN Library ที่เก็บ Property,Event,Method และ ค่าคงที่ ที่ใช้เมื่อทำการ RUN
- Search Text ใช้สำหรับค้นหา class
- Class เป็น List Box ใช้แสดง Class ต่างๆ ภายใน Library
- Member เป็น List Box ใช้แสดง Property,Event,Method และ ค่าคงที่ของ Class
- Discription ใช้แสดงรายละเอียดโดยย่อของ CLASS

8.1.2 การเพิ่ม Library ให้กับ Visual Basic

เมื่อต้องการใช้ Object ของ Application อื่นเราจะต้องเพิ่ม Library ของ Application นั้นเข้ามาใน Visual Basic ก่อนโดยเลือกจาก Menu Project -> Reference ตามรูป 8.4



รูป 8.4 เมนูสำหรับ Add Library



รูป 8.5 Add Library ของ Application Kepware OPC Automation

8.1.3 OLE Automation

เป็นการเชื่อมโยงการทำงานหรือข้อมูลระหว่าง Visual Basic กับ Application อื่นโดยการเรียก Exposed Object มาใช้งานซึ่งจะถูกแบ่งออกเป็นสองฝั่ง คือ ฝั่งทางด้าน Application ที่เรียกใช้ Exposed Object ฝั่งนี้จะเรียกว่า “OLE Client” และฝั่ง Application ที่เป็นเจ้าของ Exposed Object เรียกว่า “OLE Sserver”

เมื่อต้องการนำ Application มาใช้งานร่วมกับ Visual Basic ในรูปแบบของ OLE Automation ให้ทำตามขั้นตอนดังนี้

1 กำหนดตัวแปรประเภท Object โดยใช้รูปแบบประกาศตัวแปรดังนี้

```
Dim < variable> As <Library name> โดยที่ variable เป็น ชื่อตัวแปร
Library name เป็นชื่อ Library
```

2 การประกาศตัวแปรทำได้ 2 วิธีดังนี้

- Early Binding เป็นการ ประกาศตัวแปรที่กำหนดชื่อของ Library โดยตรงใช้ในกรณีที่ไม่ทราบชื่อของ Library เช่น Library ของ VB

```
Dim myobj as textbox
```

- Late Binding เป็นการประกาศตัวแปรในกรณีที่ไม่ทราบชื่อของ Library โดยใช้คำว่า “Object” แทนชื่อของ Library

3 การเรียก Object มาใช้งานแบ่งออกเป็น 2 กรณี

3.1 กรณีสร้าง Object ใหม่ ใช้คำสั่ง

CreateObject(appname.objecttype) โดยที่ appname หมายถึง ชื่อ application
Objecttype หมายถึง ชื่อ Object

3.2 กรณีการใช้ Object ที่ปรากฏอยู่แล้ว เช่นการอ่านเอกสารของ Microsoft word

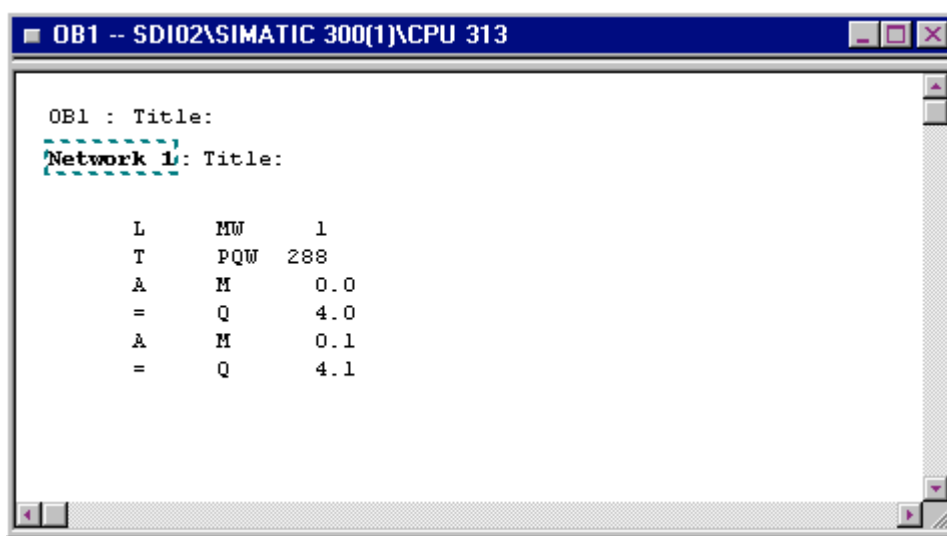
ใช้คำสั่ง GetObject(AppName.ObjectType)

4 ใช้ Method หรือ Property ของ OBJECT ที่จำเป็นต่อการใช้งาน โดยดูจาก ListBox Member

5 ปิด Object ที่ไม่ต้องการใช้ด้วย Method Close และลบตัวแปรที่ใช้แทน Object นั้นออกจากหน่วยความจำ ด้วยการกำหนดค่า “ Nothing “ เช่น Set MyObject = Nothing

8.2 การใช้ Visual Basic ติดต่อสื่อสารกับ PLC SIEMENS S7300U โดยผ่าน Kepware V4.0 โดยที่ Visual basic เปรียบเสมือนเป็น Client และ Kepware เป็น Server โดยขั้นตอนในการพัฒนาโปรแกรมมีดังนี้

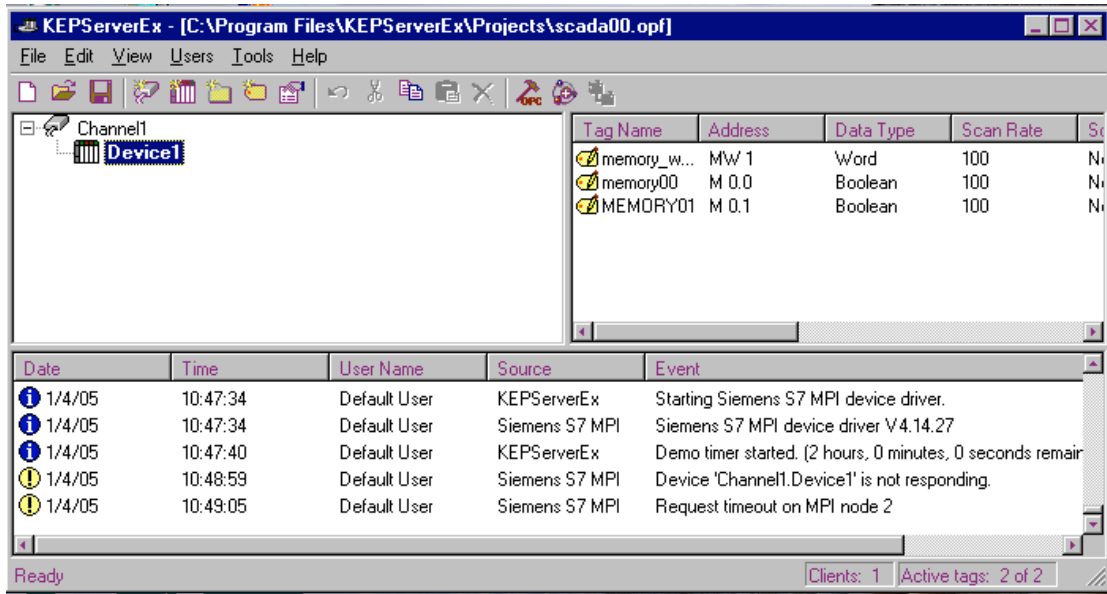
8.2.1 ขั้นตอนการพัฒนาโปรแกรมบน PLC โดยใช้โปรแกรม SIMATIC 7



รูป 8.6 แสดงโปรแกรมที่อยู่ในตัว PLC

จากรูปที่ 8.6 เราได้สร้างโปรแกรมเพื่อควบคุมการทำงานของ PLC ขึ้นมาเพียง NETWORK เดียวโดยจากโปรแกรมจะเห็นว่า OUTPUT Q4.0 ถูกควบคุมด้วย Memory M0.0 , OUTPUT Q4.1 ถูกควบคุมด้วย Memory M0.1 และ นำค่าใน Memory Word MW1 ส่งให้ Analog output PQW288 ตามลำดับ

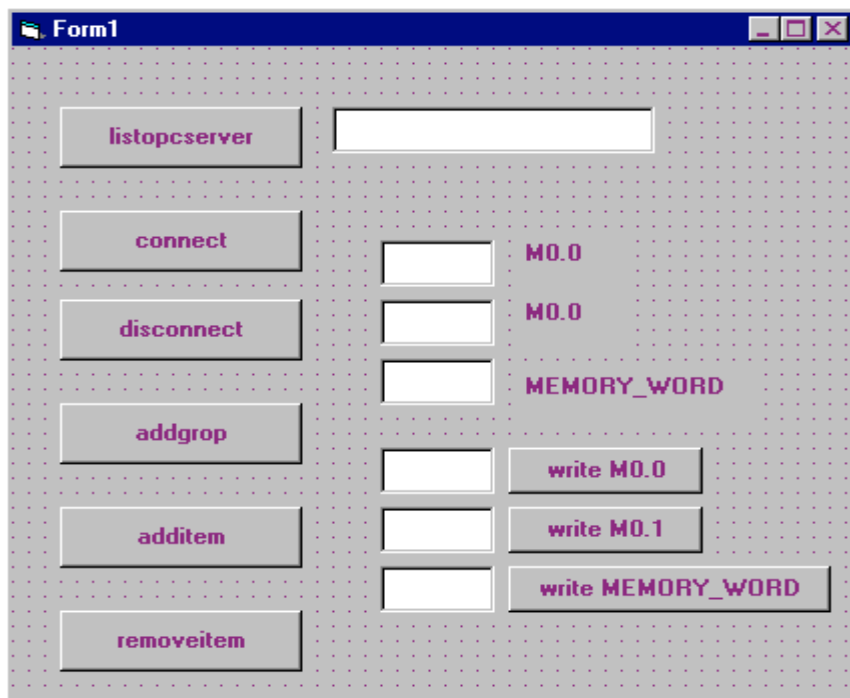
8.2.2 กำหนดให้ SOFTWARE KEPWARE V4.0 ติดต่อกับ MEMORY M0.0 และ M0.1 ใน PLC โดยขั้นตอนต่าง ๆ ตามที่แสดงไว้ในบทที่ 7



รูป 8.7 แสดงการกำหนดให้ KEPServerEx เข้าถึง Memory M0.0 , M0.1 และ MW1

8.2.3 พัฒนาโปรแกรมบน Visual Basic เพื่อเขียนและอ่านค่าในตำแหน่ง M0.0 , M0.1 และ MW1 ใน PLC โดยลำดับการทำงานมีดังนี้

- Add Library ของ Application Kepware ตามรูป 8.5
- สร้าง Project Visual Basic



รูป 8.9 วาง Tool เพื่อรองรับการพัฒนาโปรแกรม

tool	property	define
Command1	caption	listopcserver
Command2	caption	connect
Command3	caption	disconnect
Command4	caption	addgrop
Command5	caption	additem
Command6	caption	Remove item
Command7	caption	Write M0.0
Command8	caption	Write M0.1
Command9	caption	Write MEMORY_WORD
Textbox Dataopc(0)	text	ว่าง ""
Textbox Dataopc(1)	text	ว่าง ""
Textbox Dataopc(2)	text	ว่าง ""
Textbox text2	text	ว่าง ""
Textbox text4	text	ว่าง ""
Textbox text5	text	ว่าง ""
Label1	caption	M0.0
Label2	caption	M0.1
Label3	caption	MEMORY_WORD

- ลำดับขั้นการเขียนโปรแกรม

```

Option Explicit
Option Base 1
Dim WithEvents AnOPCServer As OPCServer
Dim WithEvents ConnectedOPCServer As OPCServer
Dim ConnectedServerGroup As OPCGroups
Dim WithEvents ConnectedGroup As OPCGroup
Dim OPCItemCollection As OPCItems
Dim OneOPCItem As OPCItem
Dim ItemCount As Long
Dim OPCItemIDs(10) As String
Dim ItemServerHandles() As Long
Dim ItemServerErrors() As Long
Dim ClientHandles(10) As Long

```

รูป8.10 ประกาศตัวแปรที่ General

```
KEPServerEX_Simple OPC_Example - Form1 (Code)
Command1 Click
Private Sub Command1_Click()
    Dim AllOPCServers As Variant
    Set AnOPCServer = New OPCServer
    AllOPCServers = AnOPCServer.GetOPCServers
    Text1.Text = AllOPCServers(1)
End Sub
```

รูป 8.11 โปรแกรมที่ปุ่ม Listopc server (Sub command1_click)

```
KEPServerEX_Simple OPC_Example - Form1 (Code)
Command2 Click
Private Sub Command2_Click()
    Dim servername As String
    Set ConnectedOPCServer = New OPCServer
    servername = Text1.Text
    ConnectedOPCServer.Connect servername, ""
End Sub
```

รูป 8.12 โปรแกรมที่ปุ่ม Connect (Sub command2_click)

```
KEPServerEX_Simple OPC_Example - Form1 (Code)
Command3 Click
Private Sub Command3_Click()
    If Not ConnectedOPCServer Is Nothing Then
        ConnectedOPCServer.Disconnect
        Set ConnectedOPCServer = Nothing
    End If
End Sub
```

รูป 8.13 โปรแกรมที่ปุ่ม disConnect (Sub command3_click)


```

Command4 Click
Private Sub Command4_Click()
    Set ConnectedServerGroup = ConnectedOPCServer.OPCGr
    ConnectedServerGroup.DefaultGroupIsActive = True
    ConnectedServerGroup.DefaultGroupDeadband = 0
    Set ConnectedGroup = ConnectedServerGroup.Add("g1")
    ConnectedGroup.UpdateRate = 100
    ConnectedGroup.IsSubscribed = True
End Sub

```

รูป 8.14 โปรแกรมที่ปุ่ม addgroup (Sub command4_click)

```

Command5 Click
Private Sub Command5_Click()
    Set OPCItemCollection = ConnectedGroup.OPCItems
    OPCItemCollection.DefaultIsActive = True
    ItemCount = 3
    OPCItemIDs(1) = "channel1.device1.memory00"
    OPCItemIDs(2) = "channel1.device1.MEMORY01"
    OPCItemIDs(3) = "channel1.device1.memory_word"
    ClientHandles(1) = 0
    ClientHandles(2) = 1
    ClientHandles(3) = 2
    OPCItemCollection.DefaultIsActive = True
    OPCItemCollection.AddItem 3, OPCItemIDs, ClientHan
End Sub

```

รูป 8.15 โปรแกรมที่ปุ่ม additem (Sub command5_click)

```

Command6 Click
Private Sub Command6_Click()

    Dim RemoveItemServerHandles(10) As Long
    Dim RemoveItemServerErrors() As Long
    RemoveItemServerHandles(1) = ItemServerHandles(1)
    RemoveItemServerHandles(2) = ItemServerHandles(2)
    RemoveItemServerHandles(3) = ItemServerHandles(3)
    ItemCount = 3
    OPCItemCollection.Remove ItemCount, RemoveItemServerHandles,
End Sub

```

OPCItemCollection.Remove ItemCount, RemoveItemServerHandles, RemoveItemServerErrors

รูป 8.16 โปรแกรมที่ปุ่ม removeitem (Sub command5_click)

```

KEPServerEX_Simple OPC_Example - Form1 (Code)
Command7 Click
Private Sub Command7_Click()
    ItemCount = 1

    Dim SyncItemValues(1) As Variant
    Dim SyncItemServerHandles(1) As Long
    Dim SyncItemServerErrors() As Long
    SyncItemServerHandles(1) = ItemServerHandles(1)
    SyncItemValues(1) = Val(Text5.Text)
    ConnectedGroup.SyncWrite ItemCount, SyncItemServerHandles, SyncItemValues,

End Sub

```

ConnectedGroup.SyncWrite ItemCount, SyncItemServerHandles, SyncItemValues,
SyncItemServerErrors

รูป 8.17 โปรแกรมที่ปุ่ม write M0.0(Sub command7_click)

```

KEPServerEX_Simple OPC_Example - Form1 (Code)
Command8 Click
Private Sub Command8_Click()
    ItemCount = 1
    Dim SyncItemValues(1) As Variant
    Dim SyncItemServerHandles(1) As Long
    Dim SyncItemServerErrors() As Long
    SyncItemServerHandles(1) = ItemServerHandles(2)
    SyncItemValues(1) = Val(Text4.Text)
    ConnectedGroup.SyncWrite ItemCount, SyncItemServerHandles, SyncItemValues,

End Sub

```

ConnectedGroup.SyncWrite ItemCount, SyncItemServerHandles, SyncItemValues,
SyncItemServerErrors

รูป 8.18 โปรแกรมที่ปุ่ม write M0.1(Sub command8_click)

```

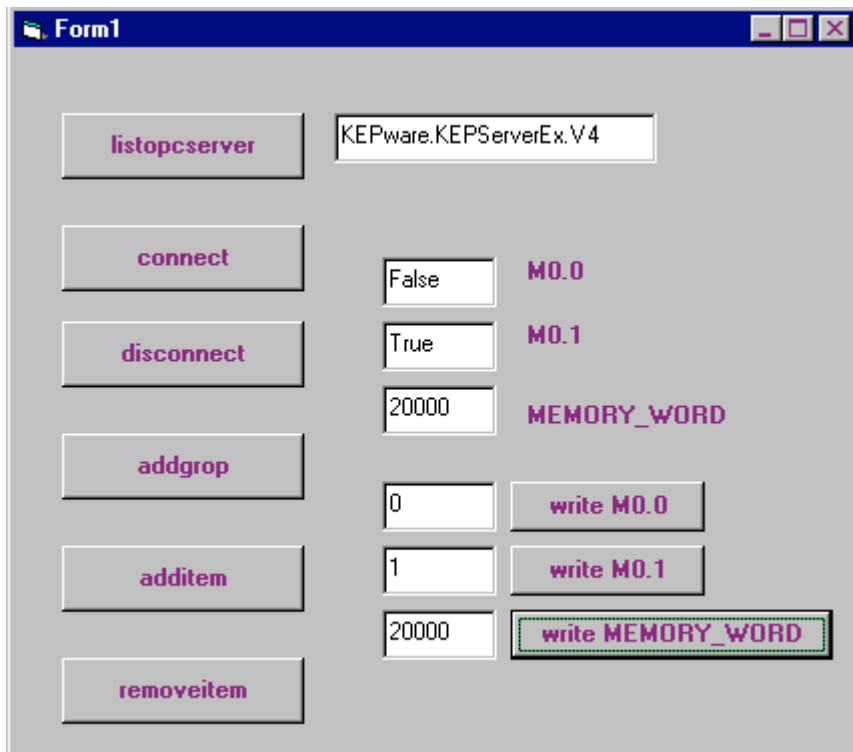
KEPServerEX_Simple OPC_Example - Form1 (Code)
Command9 Click
Private Sub Command9_Click()
    ItemCount = 1
    Dim SyncItemValues(1) As Variant
    Dim SyncItemServerHandles(1) As Long
    Dim SyncItemServerErrors() As Long
    SyncItemServerHandles(1) = ItemServerHandles(3)
    SyncItemValues(1) = Val(Text2.Text)
    ConnectedGroup.SyncWrite ItemCount, SyncItemServerHandles, SyncItemServerErrors, SyncItemValues

End Sub

```

ConnectedGroup.SyncWrite ItemCount, SyncItemServerHandles, SyncItemValues,
SyncItemServerErrors

รูป 8.19 โปรแกรมที่ปุ่ม write MEMORY_WORD(Sub command9_click)



รูป 8.20 แสดงจอภาพเมื่อ RUN PROGRAM

