

การกำหนดสเกลในการแสดงผลที่จอภาพ

ผศ.สรกานต์ ศรีทองอ่อน

อาร์มภบท

ปัญหาคลาสสิกปัญหาหนึ่งในการเขียนโปรแกรม ที่ผู้เขียนได้รับคำถามนี้มาเสมอคือ จะเขียนโปรแกรมให้แสดงผลทางกราฟิก (graphic) ได้อย่างไร โดยเฉพาะในงานทางวิศวกรรม ซึ่งการเห็นภาพจะชัดเจนกว่าการแสดงผลทางตัวเลข (หรือช่วยเสริมการแสดงผลทางตัวเลขให้สมบูรณ์ขึ้น) แต่การเขียนโปรแกรมทางด้านนี้ หาหนังสืออ้างอิงได้น้อยมาก เพราะส่วนใหญ่จะเน้นไปในทางทฤษฎี ส่วนในทางปฏิบัตินั้น ดูเหมือนว่าโปรแกรมเมอร์จะต้องค้นหาเทคนิคด้วยตนเอง ซึ่งในทัศนะของผู้เขียนเห็นว่า ทำให้การพัฒนาทางด้านนี้ไปได้ช้ามาก หลายกรณีอาจเรียกได้ว่า ‘พายเรือในอ่าง’ ด้วยซ้ำ

ปัญหาข้างต้น นำมาซึ่งวัตถุประสงค์ของบทความชุดนี้คือ แนวทางการพัฒนาโปรแกรมทางด้านกราฟิกในงานวิศวกรรม ซึ่งแนวคิดในการนำเสนอคือ ใช้การสาธิตโปรแกรมตัวอย่างร่วมกับความรู้ภาคทฤษฎี โดยเน้นในงานด้านวิศวกรรมโยธา ซึ่งเป็นสายงานที่ผู้เขียนทำอยู่ แต่ในงานวิศวกรรมอื่น ก็สามารถใช้นำทางไปประยุกต์ได้ เพราะใช้หลักการเดียวกัน ต่างกันที่วิธีการเท่านั้น

สำหรับการนำเสนอบทความ จะมีลักษณะต่อเนื่องกันไป โดยแสดงแนวคิดในเบื้องต้นต่อดัวยรหัสโปรแกรม (source code) ซึ่งผู้อ่านสามารถทำตามเพื่อให้เห็นผลได้ด้วยตนเอง พร้อมกับสามารถนำรหัสโปรแกรมไปประยุกต์ใช้ในงานของตนได้ แต่ทั้งนี้ ผู้อ่านควรติดตามบทความไปตามลำดับ เนื่องจากรหัสโปรแกรมจะนำมาใช้อย่างต่อเนื่องในส่วนหนึ่ง และอีกส่วนหนึ่งจะแสดงให้เห็นถึงพัฒนาการของรหัสโปรแกรมนั้น

ความรู้พื้นฐาน

การติดตามบทความนี้เพื่อให้ได้ผลเต็มที่ ผู้อ่านควรมีพื้นฐานความรู้ดังนี้

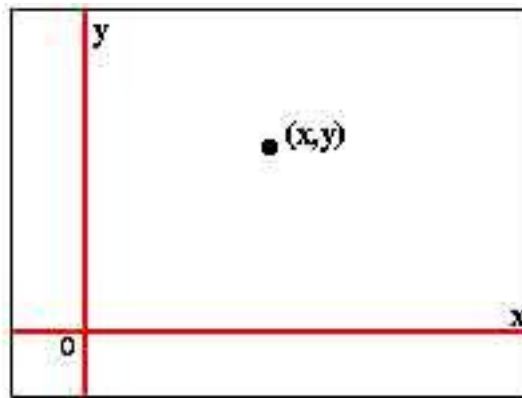
1. การเขียนโปรแกรมเบื้องต้นโดยใช้ไมโครซอฟท์วิซวลเบสิก (Microsoft Visual Basic) ซึ่งผู้เขียนใช้เวอร์ชัน 6.0 ในการสร้างโปรแกรมตัวอย่าง

2. วิชากลศาสตร์วิศวกรรม (engineering mechanics) และกำลังวัสดุ (strength of materials)

แนวคิด

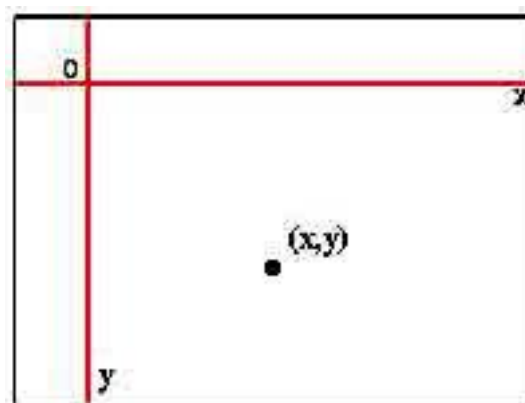
ในการโปรแกรมรุ่นเก่า โดยปกติหากจะสร้างกราฟขึ้นมา จะต้องทำการแปลงพิกัด โดยเรียกว่าเปลี่ยนพิกัดโลก (world coordinate) มาเป็นพิกัดจอภาพ (screen coordinate) ซึ่งพิกัดโลก หมายถึงพิกัดที่เราใช้วัดกันทั่วไปเช่น เมตร, ฟุต เป็นต้น พิกัดจอภาพ หมายถึง พิกัดที่จอภาพใช้อ้างอิง มีหน่วยเป็นจุดภาพ (pixel)

ปัญหาในการแปลงคือ โดยทั่วไป พิกัดโลก แกน X จะวัดจากซ้ายไปขวา ส่วนแกน Y จะวัดจากล่างขึ้นบน ดังภาพที่ 1



ภาพที่ 1 แกนพิกัดโลก

แต่พิกัดจอภาพ แกน X จะวัดจากซ้ายไปขวา ส่วนแกน Y จะวัดจากบนลงล่าง ดังภาพที่ 2



ภาพที่ 2 แกนพิกัดจอภาพ

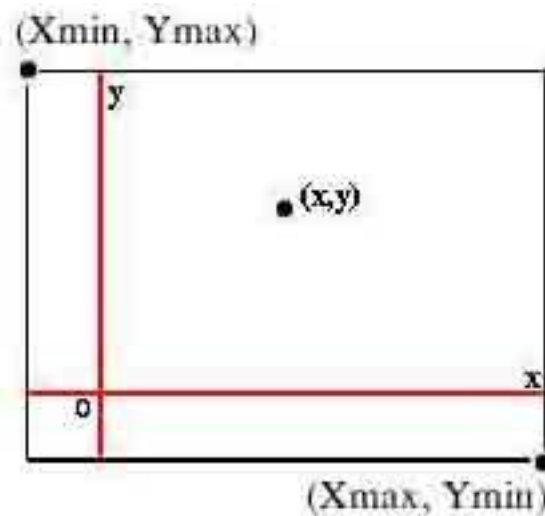
ซึ่งการแปลงจะใช้ 2 วิธี คือ

วิธีแรก กำหนด Y ให้สัมพันธ์กับพิกัดจอภาพ (Y ในพิกัดโลก คูณด้วย -1 จะเป็น Y ในพิกัดจอภาพ)

วิธีที่สอง ใช้การคำนวณแปลงให้พิกัด Y ในจอภาพ กลับเป็นชี้ขึ้นเหมือน Y ในพิกัดโลก

อย่างไรก็ตาม ในวิชาเวกเตอร์ ได้ช่วยในการแก้ปัญหาตรงนี้ คือเราสามารถกำหนดสเกลในการแสดงผลขึ้นมาใหม่ ให้อยู่ในสภาพพิกัดโลกได้ทันที โดยการกำหนดพิกัด 2 จุด คือ มุมบนซ้าย และ มุม

ล่างขวา ดังภาพที่ 3

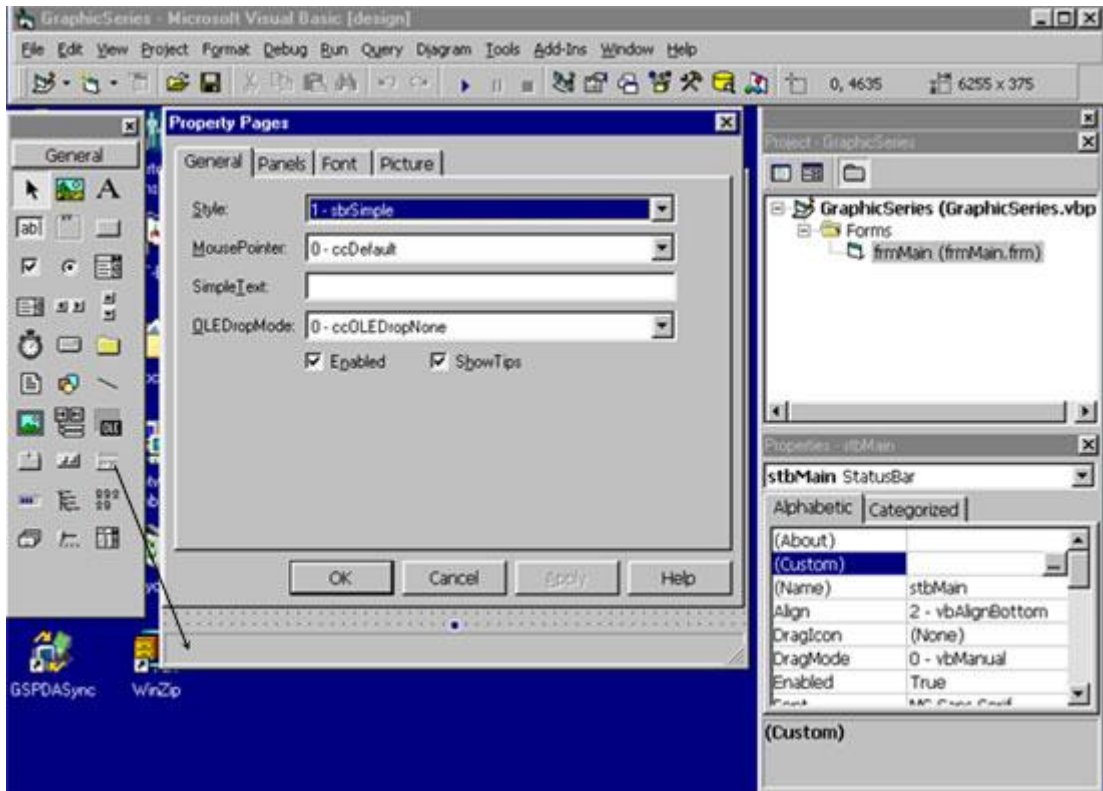


ภาพที่ 3 การกำหนดแกนพิกัดในวิชาเวกเตอร์

โปรแกรม

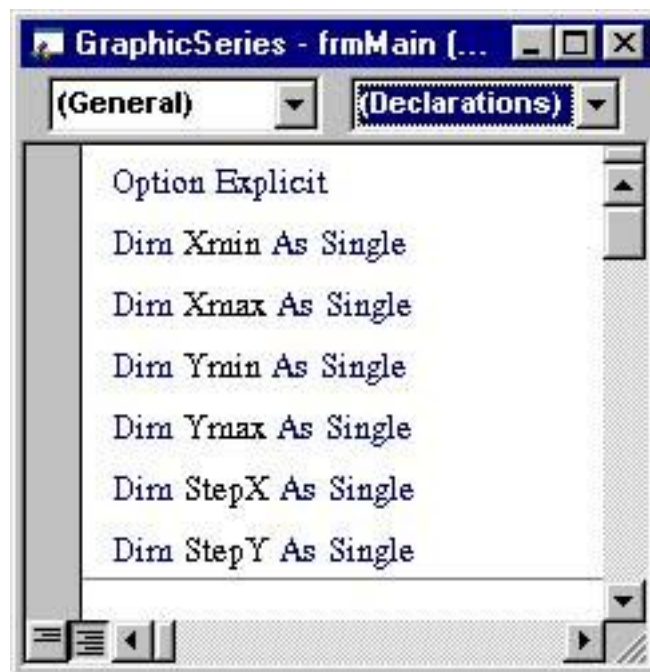
เราจะใช้หลักการนี้ทดลองสร้างโปรแกรมขึ้นมา โดยกำหนดสเกลให้เป็นแบบพิกัดโลก แล้วลงจุดกริดทุกๆ ระยะ 1 m และให้มีการแสดงตำแหน่งพิกัดตามเมาส์ที่เคลื่อนไป มีขั้นตอนดังนี้

1. สร้างโปรเจกต์ใหม่ (อาจกำหนด Name ว่า GraphicSeries) มีหนึ่งฟอร์ม (อาจกำหนด Name ว่า frmMain) แล้วใส่คอนโทรล StatusBar (ไปที่เมนู ProjectComponents เลือกคอนโทรลกลุ่ม Microsoft Windows Common Control 6.0 คอนโทรล StatusBar จึงจะปรากฏที่ Toolbox) กำหนด Name ว่า stbMain แล้วเลือก Custom กำหนด Style เป็น 1-sbrSimple ดังภาพที่ 4



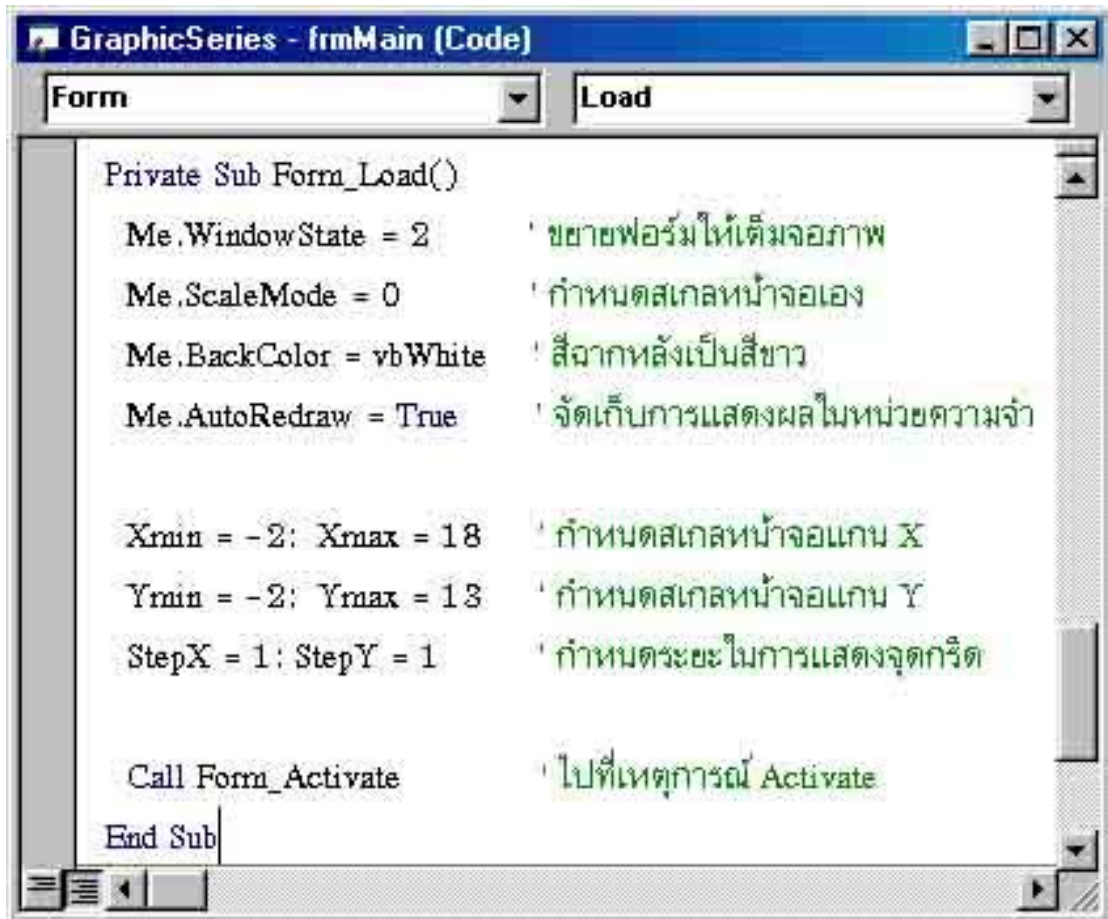
ภาพที่ 4 การใส่คอนโทรล StatusBar

2. ประกาศตัวแปรที่ General Declaration ของฟอร์ม ดังภาพที่ 5



ภาพที่ 5 การประกาศตัวแปรใน General Declaration

3. เขียนรหัสคำสั่งในเหตุการณ์ Load ของฟอร์ม ดังภาพที่ 6



```
GraphicSeries - frmMain (Code)
Form Load
Private Sub Form_Load()
    Me.WindowState = 2      ' ขยายฟอร์มให้เต็มจอภาพ
    Me.ScaleMode = 0       ' กำหนดสเกลหน้าจอเอง
    Me.BackColor = vbWhite ' สีฉากหลังเป็นสีขาว
    Me.AutoRedraw = True   ' จัดเก็บการแสดงผลในหน่วยความจำ

    Xmin = -2: Xmax = 18   ' กำหนดสเกลหน้าจอแกน X
    Ymin = -2: Ymax = 13   ' กำหนดสเกลหน้าจอแกน Y
    StepX = 1: StepY = 1   ' กำหนดระยะในการแสดงจุดกริด

    Call Form_Activate     ' ไปที่เหตุการณ์ Activate
End Sub
```

ภาพที่ 6 รหัสคำสั่งในเหตุการณ์ Load ฟอร์ม

4. เขียนรหัสคำสั่งในเหตุการณ์ Activate ของฟอร์ม ดังภาพที่ 7

```
GraphicSeries - frmMain (Code)
Form Activate
Private Sub Form_Activate()
    Dim Row As Integer: Dim Col As Integer
    Dim NumX As Single: Dim NumY As Single
    Dim X As Single: Dim Y As Single

    Me.Scale (Xmin, Ymax)-(Xmax, Ymin) ' กำหนดสเกลการแสดงผล
    NumX = (Abs(Xmax - Xmin) / StepX) + 1 ' หาจำนวนจุดกริดในแกน X
    NumY = (Abs(Ymax - Ymin) / StepY) + 1 ' หาจำนวนจุดกริดในแกน Y
    Cls ' เคลียร์หน้าจอ
    ' พล็อตจุดกริด
    X = Xmin: Y = Ymin
    For Row = 1 To NumX
        For Col = 1 To NumY
            PSet (X, Y), QBColor(4)
            Y = Y + StepY
        Next Col
        X = X + StepX: Y = Ymin
    Next Row
    ' ลากเส้นแกนอ้างอิงพิกัด (0,0)
    Line (0, 0)-(1.5, 0), QBColor(4): Line (0, 0)-(0, 1.5), QBColor(4)
End Sub
```

ภาพที่ 7 รหัสคำสั่งในเหตุการณ์ Activate ของฟอร์ม

- เขียนรหัสคำสั่งในเหตุการณ์ MouseMove ของฟอร์ม ดังภาพที่ 8

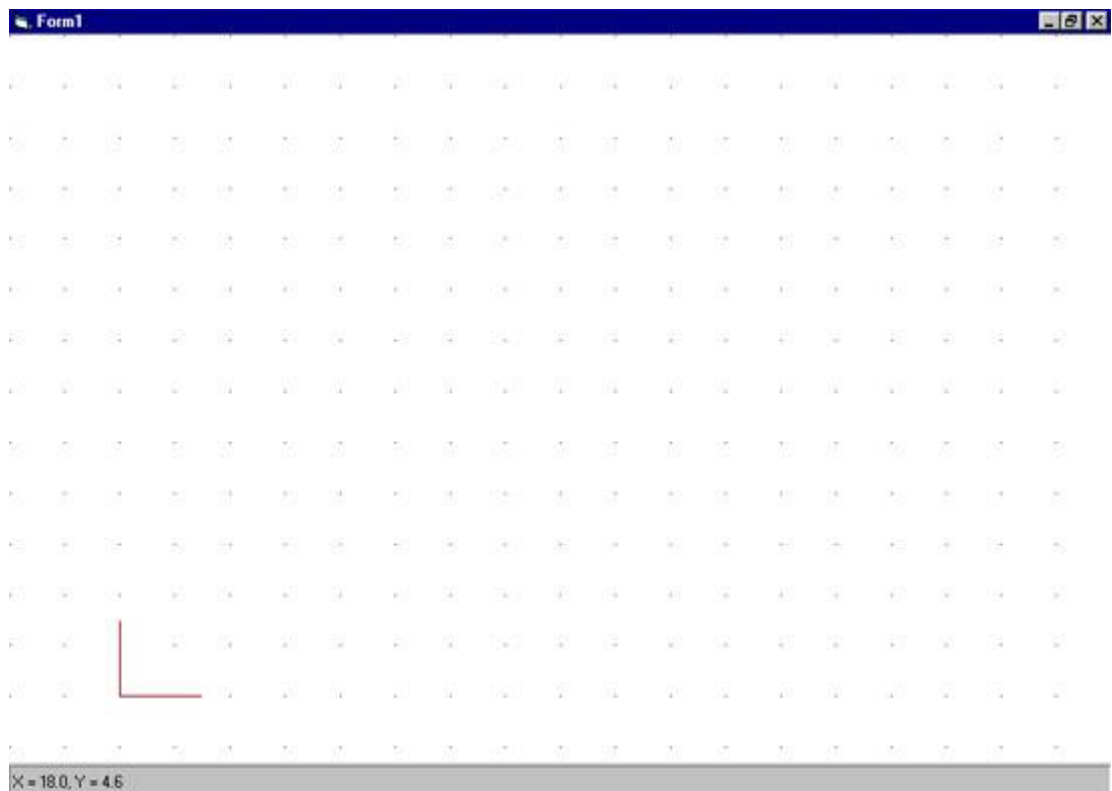
```

Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
' แสดงตำแหน่งพิกัดที่ StatusBar ทดนิยมหนึ่งตำแหน่ง
stbMain.SimpleText = "X = " & Format(X, "0.0") & ", Y = " & Format(Y, "0.0")
End Sub

```

ภาพที่ 8 รหัสคำสั่งในเหตุการณ์ MouseMove ของฟอร์ม

6. เมื่อทำการแสดงผล (run) จะได้ดังภาพที่ 9 (สังเกตเมื่อเลื่อนเมาส์ ค่าพิกัดจะเปลี่ยนตาม)



ภาพที่ 9 การแสดงผลของโปรแกรม

- สัดส่วนจอภาพของความสูงต่อความกว้าง เท่ากับ 3 : 4
- ขอให้ทดลองเปลี่ยนค่า max, min ของแกน X และ Y และเปลี่ยนค่าระยะลงจุดกริด แล้วดูผลนะครับ จะเข้าใจมากขึ้น

หนังสืออ้างอิง

สรกานต์ ศรีทองอ่อน. วิหวลเบสิกในงานวิศวกรรมโยธา. กรุงเทพฯ : สำนักพัฒนาเทคนิคศึกษา
สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2544.

Weiskamp, Keith and Heiny, Loren. Power Graphics Using Turbo Pascal 6. the United States
of America : John Wiley & Sons, Inc., 1991.

เว็บไซต์ www.geocities.com/karnprogram.