

030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

แผนปฏิบัติการสอนสัปดาห์ที่ 11

วิชา 030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

ระดับ: ปริญญาตรี

เรื่อง 2D และ 3D Graphics

เวลา: บรรยาย 120 นาที

ปฏิบัติ 120 นาที

ก. วัตถุประสงค์การสอน	รายละเอียดตามที่จะพบไว้ใน
1. ใช้ Canvas วาดภาพ 2D 2. สร้าง 2D และ 3D Graph 3. สร้าง 3D Objects 4. ควบคุมการหมุน และการเคลื่อนที่ของวัตถุ 5. แสดงผลการทำงานจริงของเครื่องจักรด้วยภาพเคลื่อนไหว	} หน้า 156-165

ข. การจัดการเรียนการสอน

เวลา - นาที		0	60	120	180	240
วัตถุประสงค์		1, 2, 3, 4, 5				
การนำเข้าสู่บทเรียน						
ให้เนื้อหา						
สรุปเนื้อหา						
พัก						
ทำแบบฝึกหัด						
ให้เนื้อหา						
สรุปเนื้อหา						
ทดสอบและเก็บคะแนน						
ประเมินผล		พิจารณาจากผลการทดสอบในช่วงโมดสุดท้าย				
วิธีการสอน:	บรรยาย					
	ถาม - ตอบ					
	ทำแบบฝึกหัด					
	บรรยาย และแสดงให้ดู					
	ทดสอบ					
สื่อการสอน:	คอมพิวเตอร์					
	Presentations					

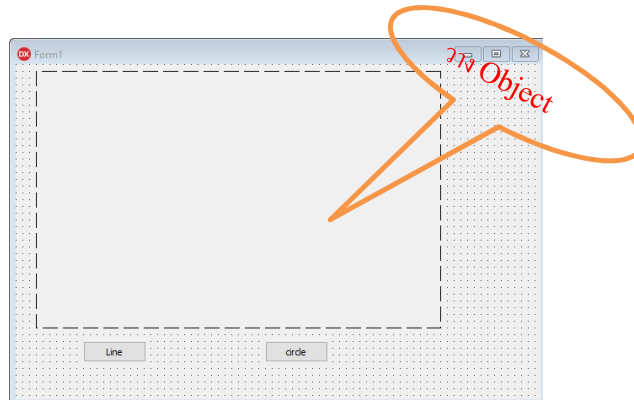
Content

1. ใช้ Canvas วาดภาพ 2D

การแสดงผลภาพวาดหรือกราฟิกที่ได้จากการคำนวณพิกัดสามารถทำได้ด้วยคำสั่ง Canvas สำหรับ Object ที่มีคุณสมบัติรองรับการวาดภาพเท่านั้น เช่น TForm, TImage

http://docs.embarcadero.com/products/rad_studio/delphiAndcpp2009/HelpUpdate2/EN/html/delphivclwin32/Graphics.html

Example



Code

```
unit Unit1;

interface

uses
  Winapi.Windows, Winapi.Messages, System.SysUtils,
  System.Variants, System.Classes, Vcl.Graphics,
  Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls,
  Vcl.ExtCtrls;

type
  TForm1 = class(TForm)
    Image1: TImage;
    Button1: TButton;
    Button2: TButton;
  procedure Button1Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);
  private
    { Private declarations }
```

```
public
  { Public declarations }
end;

var
  Form1: TForm1;

implementation

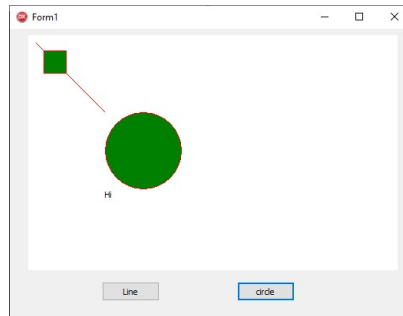
{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
begin
  Image1.Canvas.Pen.Color:=ClRed;
  Image1.Canvas.MoveTo(10, 10);
  Image1.Canvas.LineTo(100, 100);
  Image1.Canvas.TextOut(100, 200, 'Hi');
end;
```

```
procedure TForm1.Button2Click(Sender: TObject);  
begin  
  Image1.Canvas.Pen.Color:=ClRed;  
  Image1.Canvas.Brush.Color:=ClGreen;  
  Image1.Canvas.Brush.Style:=bsSolid;
```

```
Image1.Canvas.Ellipse(100, 100, 200,200);  
Image1.Canvas.Rectangle(20, 20, 50, 50);  
end;  
  
end.
```

Result

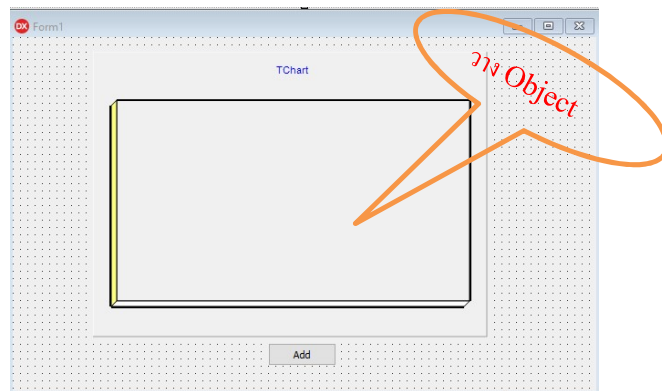


[Code\week11\Ex_Canvas_1](#)

2. สร้าง 2D และ 3D Graph

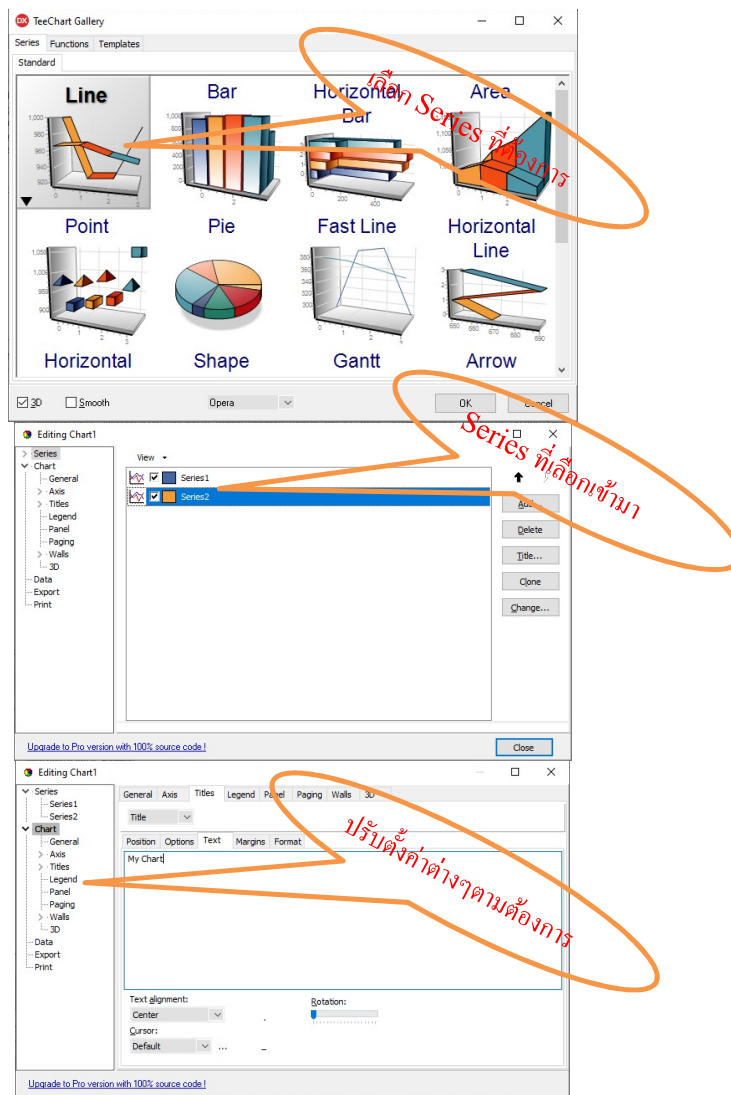
ในการสร้าง Chart ในรูปแบบต่าง ๆ นั้นสามารถสร้างได้โดยการสร้าง Series ภายใน TChart ซึ่ง Series นั้นเป็นเสมือนกราฟ หรือชุดข้อมูลหนึ่งใน Chart สามารถมีหลาย Series ได้

Example



เมื่อวาง Component TChart ลงบน Form แล้วให้ DbClick ที่ Chart1





Code

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Winapi.Windows, Winapi.Messages, System.SysUtils,  
System.Variants, System.Classes, Vcl.Graphics,  
Vcl.Controls, Vcl.Forms, Vcl.Dialogs, VclTee.TeeGDIPlus,  
VCLTee.TeEngine,  
VCLTee.Series, Vcl.StdCtrls, Vcl.ExtCtrls,  
VCLTee.TeeProcs, VCLTee.Chart;
```

```
type
```

```
TForm1 = class(TForm)
```

```
Chart1: TChart;
```

```
Add: TButton;
```

```
Series1: TLineSeries;
```

```
Series2: TLineSeries;
```

```
Button1: TButton;
```

```
procedure AddClick(Sender: TObject);
```

```
procedure Button1Click(Sender: TObject);
```

```
private
```

```
{ Private declarations }  
  
public  
  
{ Public declarations }  
  
end;  
  
var  
    Form1: TForm1;  
  
implementation  
  
{$R *.dfm}  
  
procedure TForm1.AddClick(Sender: TObject);
```

[Code\week11\Ex_2D_3D_Graph_1](#)

```
begin  
    Series1.Add(Random(100));  
    Series2.Add(Random(100));  
end;  
  
procedure TForm1.Button1Click(Sender: TObject);  
begin  
    Series1.Clear;  
    Series2.Clear;  
end;  
  
end.
```

3. สร้าง 3D Objects

Delphi รองรับการสร้าง 3D Application ผู้เขียนโปรแกรมสามารถสร้างโปรแกรมที่แสดงผลใน 3 มิติ และรองรับ Multi-Devices ด้วย

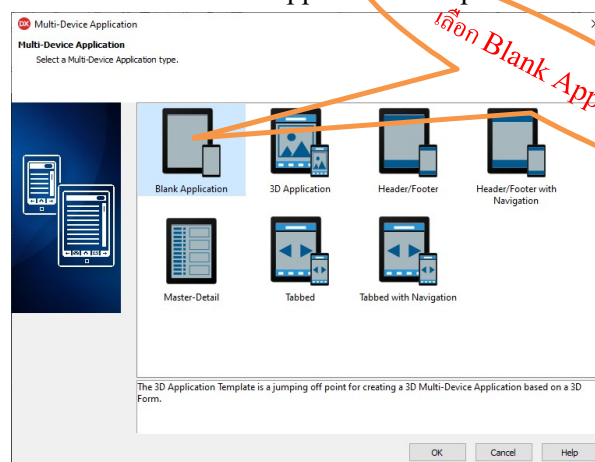
<https://www.youtube.com/watch?v=3VR9bA1xiN0>

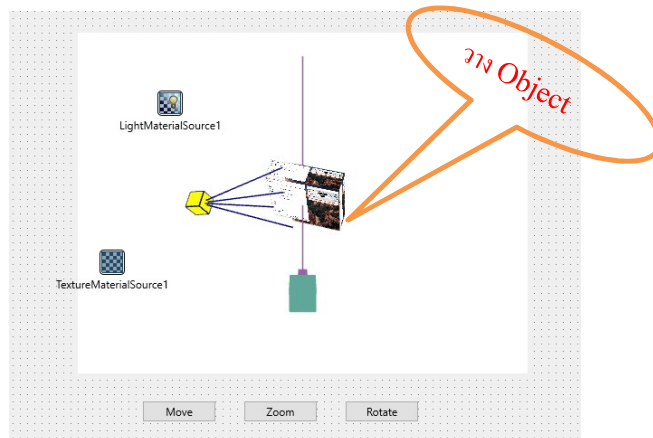
[http://docwiki.embarcadero.com/RADStudio/Rio/en/Creating_the_User_Interface_\(FireMonkey_3D_Tutorial\)](http://docwiki.embarcadero.com/RADStudio/Rio/en/Creating_the_User_Interface_(FireMonkey_3D_Tutorial))

[http://docwiki.embarcadero.com/RADStudio/Rio/en/Creating_a_2D_Interface_in_a_3D_Application_\(FireMonkey_3D_Tutorial\)](http://docwiki.embarcadero.com/RADStudio/Rio/en/Creating_a_2D_Interface_in_a_3D_Application_(FireMonkey_3D_Tutorial))

Example

สร้างโปรแกรมใหม่ด้วย File/New/Multi-Device-Application-Delphi

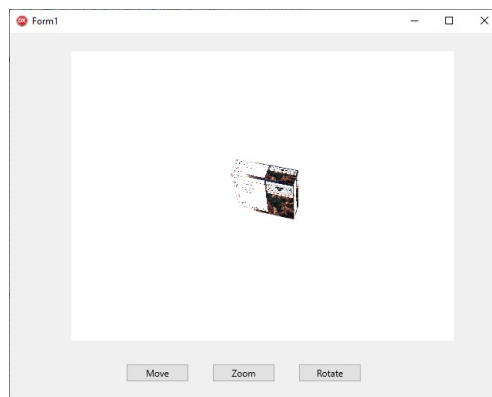




กำหนด Properties Position ของ Tcube ตามความเหมาะสม หรือใช้ Mouse ลากที่ Object Cube1 เพื่อปรับขนาดและทิศทาง

Result

เมื่อกด Run จะได้ดังนี้



4. ควบคุมการหมุน และการเคลื่อนที่ของวัตถุ

จาก Program ในหัวข้อที่ 3 ทดลองควบคุมการหมุน และการเคลื่อนที่ของวัตถุ 3 มิติได้ด้วย Code ตัวอย่างดังนี้

Code

```
unit Unit1;  
  
interface  
  
uses
```

```
System.SysUtils, System.Types, System.UITypes,  
System.Classes, System.Variants,  
FMX.Types, FMX.Controls, FMX.Forms, FMX.Graphics,  
FMX.Dialogs,  
System.Math.Vectors, FMX.Controls3D, FMX.Objects3D,  
FMX.Viewport3D,
```

```

FMX.Controls.Presentation, FMX.StdCtrls, FMX.Types3D,
FMX.MaterialSources;

type
  TForm1 = class(TForm)
    Viewport3D1: TViewport3D;
    Cube1: TCube;
    BtMove: TButton;
    BtZoom: TButton;
    BtRotate: TButton;
    Camera1: TCamera;
    LightMaterialSource1: TLightMaterialSource;
    TextureMaterialSource1: TTextureMaterialSource;
    Light1: TLight;
    procedure BtMoveClick(Sender: TObject);
    procedure BtRotateClick(Sender: TObject);
    procedure BtZoomClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var

```

```

Form1: TForm1;

implementation

{$R *.fmx}

procedure TForm1.BtMoveClick(Sender: TObject);
begin
  Cube1.Position.X:=Cube1.Position.X+1;
end;

procedure TForm1.BtRotateClick(Sender: TObject);
begin
  Cube1.RotationAngle.X:=Cube1.RotationAngle.X+10;
end;

procedure TForm1.BtZoomClick(Sender: TObject);
begin
  Cube1.Scale.X:=Cube1.Scale.X+1;
  Cube1.Scale.Y:=Cube1.Scale.Y+1;
  Cube1.Scale.Z:=Cube1.Scale.Z+1;
end;

end.

```

[Code\week11\ Ex_3D_App_1](#)

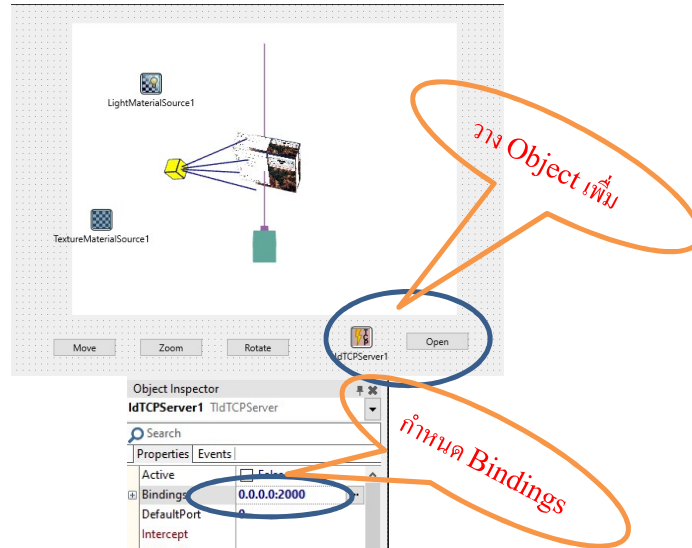
5. แสดงผลการทำงานจริงของเครื่องจักรด้วยภาพเคลื่อนไหว

สำหรับการควบคุมวัตถุ 3 มิติให้เคลื่อนที่ เปลี่ยนตำแหน่ง, ขนาด และทิศทางไปตามการเคลื่อนไหวจริงของชิ้นส่วนของเครื่องจักรนั้น จะต้องใช้ Sensor ตรวจจับค่าดังกล่าวจากชิ้นส่วนจริงแล้วส่งค่าเหล่านั้นมาให้กับโปรแกรมเพื่อแสดงผลให้เป็นไปตามตำแหน่ง และมุมจริงๆที่เกิดขึ้น

ในที่นี้ให้ชิ้นงานมี Sensor ที่สามารถวัดได้ทั้ง 6 DOF ซึ่งประกอบด้วยพิกัด X, Y, Z และมุม Roll, Pitch, Yaw เมื่อได้ค่าทั้ง 6 ค่าแล้วก็ส่งข้อมูลให้กับโปรแกรมแสดงผลด้วย Protocol ดังนี้ ผ่านทาง Network TCP/IP

```
X,Y,Z,Roll,Pitch,Yaw\r\n
```

Example



Code

```
unit Unit1;

interface

uses

    System.SysUtils, System.Types, System.UITypes,
    System.Classes, System.Variants,
    FMX.Types, FMX.Controls, FMX.Forms, FMX.Graphics,
    FMX.Dialogs,
    System.Math.Vectors, FMX.Controls3D, FMX.Objects3D,
    FMX.Viewport3D,
    FMX.Controls.Presentation, FMX.StdCtrls, FMX.Types3D,
    FMX.MaterialSources, IdBaseComponent, IdComponent,
    IdCustomTCPServer, IdTCPServer,
    IdContext;

type
    TForm1 = class(TForm)
        Viewport3D1: TViewport3D;
        Cube1: TCube;
        BtMove: TButton;
        BtZoom: TButton;
        BtRotate: TButton;
```

```
        Camera1: TCamera;
        LightMaterialSource1: TLightMaterialSource;
        TextureMaterialSource1: TTextureMaterialSource;
        Light1: TLight;
        IdTCPServer1: TIdTCPServer;
        Button1: TButton;

    procedure BtMoveClick(Sender: TObject);
    procedure BtRotateClick(Sender: TObject);
    procedure BtZoomClick(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure IdTCPServer1Execute(AContext:
        TIdContext);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    Form1: TForm1;

implementation
```


030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

```
{$R *.fmx}
```

```
procedure TForm1.BtMoveClick(Sender: TObject);
```

```
begin
```

```
    Cube1.Position.X:=Cube1.Position.X+1;
```

```
end;
```

```
procedure TForm1.BtRotateClick(Sender: TObject);
```

```
begin
```

```
    Cube1.RotationAngle.X:=Cube1.RotationAngle.X+10;
```

```
end;
```

```
procedure TForm1.BtZoomClick(Sender: TObject);
```

```
begin
```

```
    Cube1.Scale.X:=Cube1.Scale.X+1;
```

```
    Cube1.Scale.Y:=Cube1.Scale.Y+1;
```

```
    Cube1.Scale.Z:=Cube1.Scale.Z+1;
```

```
end;
```

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
begin
```

```
    IdTCPServer1.Active:=True;
```

```
end;
```

```
procedure TForm1.IdTCPServer1Execute(AContext:
```

```
TIdContext);
```

```
var
```

```
    Sl:TStringList;
```

```
    Data:String;
```

```
begin
```

```
    Data:=AContext.Connection.Socket.ReadLn;
```

```
    Sl:=TStringList.Create;
```

```
    Sl.CommaText:=Data;
```

```
    if Sl.Count = 6 then
```

```
    begin
```

```
        Cube1.Position.X:=StrToFloat(Sl[0]);
```

```
        Cube1.Position.Y:=StrToFloat(Sl[1]);
```

```
        Cube1.Position.Z:=StrToFloat(Sl[2]);
```

```
        Cube1.RotationAngle.X:=StrToFloat(Sl[3]);
```

```
        Cube1.RotationAngle.Y:=StrToFloat(Sl[4]);
```

```
        Cube1.RotationAngle.Z:=StrToFloat(Sl[5]);
```

```
    end;
```

```
    if Assigned(Sl) then
```

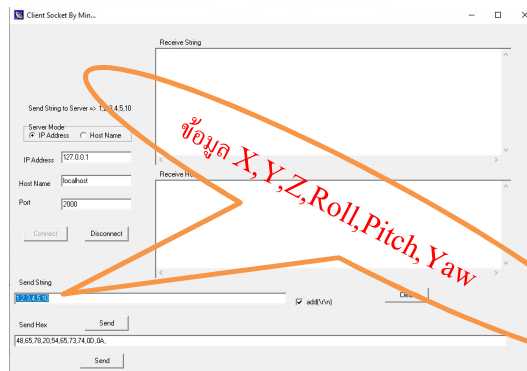
```
        FreeAndNil(Sl);
```

```
end;
```

```
end.
```

[Code\week11\ Ex_3D_App_2](#)

โปรแกรมสำหรับทดสอบส่งข้อมูล 6 DOF



030143361 การโปรแกรมคอมพิวเตอร์สำหรับงานควบคุม

Exercise

- เขียนโปรแกรม Plot graph $y = 10\sin(\omega t)$ ทุกๆ 0.1 mS. ตั้งแต่มุม 0-360 องศา โดยใช้ Canvas

Assignment

- เขียนโปรแกรม Plot graph $y=10\sin(\omega t)$ ทุกๆ 0.1 mS. ตั้งแต่มุม 0-360 องศา โดยใช้ TChart
- เขียนโปรแกรมควบคุม Cube 3 มิติ ให้เคลื่อนที่และหมุน 6 DOF ด้วย Joystick

Answer Sheet

[Code\week11\Ans_Canvas_1](#)

[Code\week11\Ans_TChart_1](#)

[Code\week11\Ans_3D_1](#)