

Excel anaglyph stereoscopy #4 – a joystick drive

by George Lungu

– In the previous section, a stereoscopic rotating cube was created. In this section, two degrees of rotation, yaw and pitch are being linked to a virtual joystick.

A brief description of the virtual joystick:

- This macro will be assigned to a small chart. When the chart is clicked the macro returns two numbers which are the relative coordinates of the mouse from the initial click point. Clicking the chart again will stop the macro. The chart will display a stick centered in the center of the chart and whose end will follow the mouse pointer on the screen. The movement will not exceed 100 pixels in any direction.
- The joystick and its operation are very important for future models. Though previously explained before, a brief description of this device and the macro behind it will be given here.
- The code for this macro has to be placed in a module otherwise it won't work (the feature retrieving the mouse coordinate will not work in any sheet section of the VBA project).
- This part of the VBA code like all declarations needs to be placed on the top of the page!!!

```
Public Declare Function GetCursorPos Lib "user32" _  
(Some_String As POINTAPI) As Long
```

} Declaration of a special API (Application Programming Interface) function which retrieves the cursor position

```
Type POINTAPI  
    X As Long  
    Y As Long  
End Type
```

} declaration of a special structure (Point API) used as the output type of the previous API function. It is essentially the pair of coordinates (as long integers) of the cursor on the screen started to be measured from the upper left corner of the screen.

```
Dim RunPause As Boolean
```

declaration of a Boolean variable which has the role of a "switch", keeping track if the macro is running or is stopped. This will allow the macro to be started or paused by clicking the same object (the joystick chart).

The JoyStick macro - continuation:

- Copy last worksheet and rename the new worksheet "3D_Stereoscopy_Joystick"
- The new JoyStick macro, shown below, is similar to the old JoyStick macro used before on this blog either in the "Joystick" tutorial series or in the "Flight Simulator" tutorial series.
- The macro will first reset the yaw and pitch then it will record the cursor coordinates at the click point that started the macro. During a continuous conditional loop the macro will constantly change the yaw and pitch parameters by a value proportional to the relative mouse coordinates. As a consequence the joystick will dictate the rotation speed not the (angular rates) for the cube positional angles.

Declaration two Point API type structures,
one as initial click coordinates and the
second as the current cursor coordinates

Yaw and pitch value are reset at the start of the macro

This flips the "RunPause" Boolean variable and allows the macro
to be started and paused clicking the same button (chart)

Assigns "Pt0" the initial click coordinates

Conditional "Do" loop declaration (start)

Always add this statement if you ever need to stop the
loop manually or update a chart while the loop is running

Every loop cycle assigns "Pt1" the current cursor coordinates

Every loop cycle calculate the relative cursor
coordinates and display them in the range "N2:O2"

Every loop cycle the yaw is changed by an amount proportional
to the x joystick deviation (this is a numerical integration)

Every loop cycle the pitch is changed by an amount
proportional to the y joystick deviation (numerical integration)

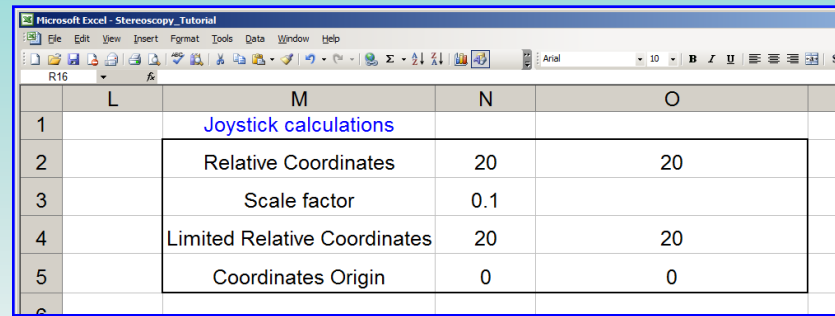
End of "Do" loop declaration

End of macro declaration

```
Sub JoyStick()  
  Dim Pt0 As POINTAPI  
  Dim Pt1 As POINTAPI  
  [B2] = 0  
  [B4] = 0  
  RunPause = Not RunPause  
  GetCursorPos Pt0  
  Do While RunPause = True  
    DoEvents  
    GetCursorPos Pt1  
    [N2] = Pt1.X - Pt0.X  
    DoEvents  
    [O2] = -Pt1.Y + Pt0.Y  
    DoEvents  
    [B2] = [B2] + [N3] * [N4]  
    DoEvents  
    [B4] = [B4] + [N3] * [O4]  
    DoEvents  
  Loop  
End Sub
```


Worksheet update:

- Copy last worksheet and rename the new worksheet "3D_Stereoscopy_Joystick".
- Create a joystick calculation area:
- Range M1:M5 will contain the labels to the left.
- Range N2:O2 will be updated by the "Joystick" macro.
- Cell N3 will contain the rotation speed scaling factor. You can adjust it later to a value you like depending on how fast you want the cube to rotate for a certain joystick deviation.
- Range N5:O5 will be the data for the origin of the joystick. Type a zero in both cells.
- Range N4:O4 will be the data for the joystick head, which is the relative coordinate data limited to the range [-100, 100] and scaled by the rotation speed scaling factor:
 - => N4: "=\$N3*IF(N2<0,MAX(-100,N2),MIN(N2,100))"
 - => Copy N4 to O4 and you get O4: "=\$N3*IF(O2<0,MAX(-100,O2),MIN(O2,100))"



	L	M	N	O
1		Joystick calculations		
2		Relative Coordinates	20	20
3		Scale factor	0.1	
4		Limited Relative Coordinates	20	20
5		Coordinates Origin	0	0

Create and format the joystick chart:

- Create a 2D scatter chart with [N4:N5] for X axis values and [O4:O5] for Y axis.
- Delete the chart legend, the titles and change the color of the plot area to your preference. Minimize the font of both axes to the value of 1. Type number 20 in both cells N2 and O2.
- Size the chart as a square and also adjust the range of both axes to [-100, 100]
- Click the center point once then wait 2 second and click it again. The cursor turns into a cross. Choose a red marker for the center of size 8 and repeat the operation for the other end of the joystick (the handle) choosing a blue circle of size 20. You can of course choose the colors and sizes you like but a small red pivot point for the joystick is useful in referencing (zeroing) the joystick neutral at the start.
- Right click the chart -> "Assign Macro" -> select "JoyStick()"
- The model is ready to use by clicking the center of the joystick chart. Observe that the rotation speed is proportional to the joystick deflection. To stop macro and the rotation just click the joystick chart again.

