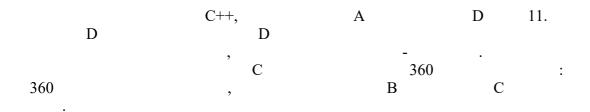


# XInput Tutorial Part 1: Adding gamepad support to your Windows game



,

•

•

•

**NOTE:** If you can't be bothered with the low-level details and just want to shoehorn gamepad support into a game really quickly, my <u>Simple2D library (version 1.11 and above)</u> includes gamepad support – see the <u>Tetris gamepad support</u> article for a quick example on how to use the library to add gamepad support in just a few minutes!

# Setting up your build environment

#### How to check whether a controller is connected

```
XInputGetState
           XINPUT STATE
XINPUT STATE state;
                            XINPUT STATE));
ZeroMemory(&state, si
if (XInputGetState(0, &state) == ERROR_SUCCESS)
// controller is connected on port 0
                                                         0.
int controllerId = -1;
                                 COUNT && controllerId == -1; i++)
for (DWORD i = 0; i <</pre>
{
XINPUT STATE state;
                           (XINPUT STATE));
ZeroMemory(&state, si
if (XInputGetState(i, &state) == ERROR_SUCCESS)
controllerId = i;
}
                  , controllerId
       for
How to test whether a particular button is pressed down
   XINPUT STATE
                               XINPUT GAMEPAD
```

```
wButtons
                                                                    1
       , 0
                                                                 wButtons
F
XINPUT GAMEPAD A
XINPUT GAMEPAD B
XINPUT GAMEPAD X
XINPUT GAMEPAD Y
D
XINPUT_GAMEPAD_DPAD_LEFT
XINPUT_GAMEPAD_DPAD_RIGHT
XINPUT_GAMEPAD_DPAD_UP
XINPUT_GAMEPAD_DPAD_DOWN
XINPUT GAMEPAD LEFT SHOULDER
XINPUT_GAMEPAD_RIGHT_SHOULDER
Α
```

```
XINPUT_GAMEPAD_LEFT_THUMB
XINPUT_GAMEPAD_RIGHT_THUMB
\mathbf{C}
XINPUT GAMEPAD BACK
XINPUT GAMEPAD START
                                                                      A D
                                   .F
   wButtons
                                                                         A,
     :
bool A button pressed = ((state.Gamepad.wButtons & XINPUT GAMEPAD A
How to test how much each trigger is depressed
                                                                    360
                                                                   ).
                                8-
                                                   0-255
255
                 ).
float leftTrigger = (float) state.Gamepad.bLeftTrigger / 255;
float rightTrigger = (float) state.Gamepad.bRightTrigger / 255;
                                  0 - 1
  state.Gamepad.bLeftTrigger
                              state.Gamepad.bRightTrigger
How to test the position of each analog thumb stick
                                                                     (0,0).
       , 0
float normLX = fmaxf(-1, (float) state.Gamepad.sThumbLX / 32767);
float normLY = fmaxf(-1, (float) state.Gamepad.sThumbLY / 32767);
(versions of Visual Studio prior to Visual Studio 2013 use a <math.h> header which isn't
C99-compliant. If fmaxf() doesn't work in your compiler, try using max() instead)
                                state. Gamepad.s ThumbLX,
state.Gamepad.sThumbLY(
                                    state.Gamepad.sThumbRX,
state.Gamepad.sThumbRY(
                                 ).
                                               1.
  32768,
                        32767.
                                                             32767
                                                       -1.
                                 -1
                                                                             -1
   +1.
```

A deadzone

### How to calibrate deadzones for the analog sticks

```
-1
                       +1
float deadzoneX = 0.05f;
float deadzoneY = 0.02f;
float leftStickX = (abs(normLX) < deadzoneX ? 0 : normLX);</pre>
float leftStickY = (abs(normLY) < deadzoneY ? 0 : normLY);</pre>
                               (5%
                                                  2%
                              360
                 ).
                                                            80%.
                         80%
      . B
                         80%-100%,
                                                               80%-100%,
    0-1. F
                                           80%
                                  80%.
                                                                    0,85%
                    0.25, 90%
                                                  0.5
leftStickX = (abs(normLX) < deadzoneX ? 0 : (abs(normLX) - deadzone</pre>
leftStickY = (abs(normLY) < deadzoneY ? 0 : (abs(normLY) - deadzone
if (deadzoneX > 0) leftStickX /= 1 - deadzoneX;
if (deadzoneY > 0) leftStickY /= 1 - deadzoneY;
E
80%,
                              0-0.2 (0%-20%).
                                     x / abs(x)
                                                                       0
            80%
                                            0 - 0.2
                                                                        1
   1.
0.8 = 0.2
                                      5. C
                                              , 0.2 (
```

```
80\% ) * 5 = 1,
```

,

## Rolling it all together into a Gamepad class

+ expand source Gamepad. Α ( Refresh(). Refresh(). GetPort(), GetState(), IsPressed(WORD button) Α Refresh(), leftStickX, leftStickY, rightStickX, rightStickY, leftTrigger rightTrigger Gamepad #include <iostream> #include <Windows.h> #include <Xinput.h> using std::cout; using std::endl; /\* Insert the Gamepad class code here \*/ int main() Gamepad gamepad; bool wasConnected = true; while (true) Sleep(100); if (!gamepad.Refresh()) if (wasConnected) wasConnected = false; cout << "Please connect an Xbox 360 controller." << endl;</pre> else if (!wasConnected) wasConnected = true;

cout << "Controller connected on port " << gamepad.GetPort() << enc</pre>

```
cout << "Left thumb stick: (" << gamepad.leftStickX << ", " << game
cout << "Left analog trigger: " << gamepad.leftTrigger << " Right
if (gamepad.IsPressed(XINPUT_GAMEPAD_A)) cout << "(A) button presse
}
}

Wait, is that it?!

, .A , ...

GetAsyncKeyState()

...

(WM_KEYDOWN WM_KEYUP)

- , ...

//</pre>
```

I'm a software developer with very limited work capacity due to having the debilitating illness M.E. – please read my article <a href="Dying with M.E. as a Software">Dying with M.E. as a Software</a>
<a href="Developer">Developer</a> and <a href="donate to the crowdfund">donate to the crowdfund</a> to help me with my bucket list if you found this article useful. Thank you so much!

!

**Useful Links** 

C A @ D