

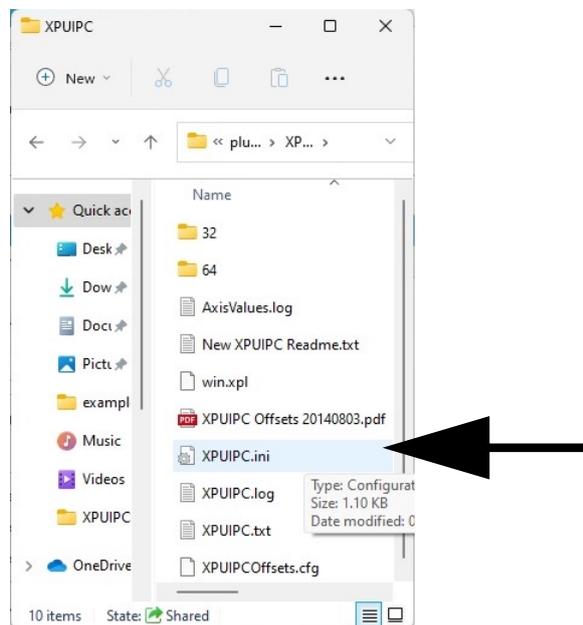
Setting up a simple home-made flight sim yoke with Arduino, X-Plane 11, Mobiflight and Arduino Mega.

Intro:

This prototype yoke was made with a Creality Ender 3 3D printer, using a 10k linear rotary potentiometer for the aileron control, and a 10k linear slide potentiometer for the elevator control. These two pots were connected to an Arduino Mega 2560 R3 clone. A usb cable connects the Arduino to a Win11 box running Mobiflight 9.2.0 and X-Plane 11, with XPUIPC installed. Because of the requirement to run Mobiflight (a Windows-only application) it is not possible to use the yoke on Mac OS X. Thanks to Pizman82, moderator of the [Mobiflight forum](#), for the help in configuring and calibrating the prototype.

Setting up the computer:

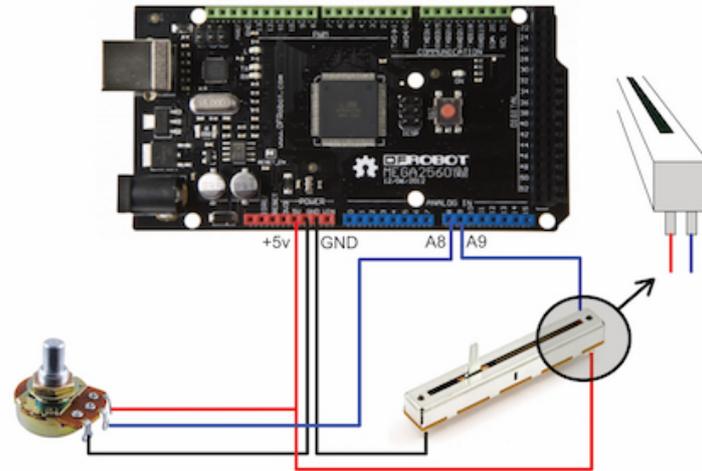
1. Install X-Plane 11 - I installed it on the desktop rather than in the Program Files directory.
2. Download, unzip and install XPUIPC from <http://www.schiratti.com/xpuipc.html>. The install is simply a matter of copying the unzipped XPUIPC folder into the X-Plane 11/Resources/Plugins folder in X-Plane 11.
3. You may need to edit one of the XPUIPC files as follows - open the XPUIPC folder and find the file called XPUIPC.ini. Open in a text editor and change the line which reads Server Address = 10.0.0.2 to your own computers IP address (which is likely to be something like 192.168.1.xxx).



4. Download and install Mobiflight from www.mobiflight.de. Once again I installed this on the desktop.

Building the yoke:

I downloaded a yoke design from Thingiverse - <https://www.thingiverse.com/thing:4855469>. There are several different designs, but this seemed suitable for a prototype. I made some changes, mainly to use a linear slide potentiometer for the elevator control rather than the rotary pot in the design, and to suit the size of the other components I had to hand at the time. The 3D printing files for my own design are in a public Dropbox folder - [link](#)

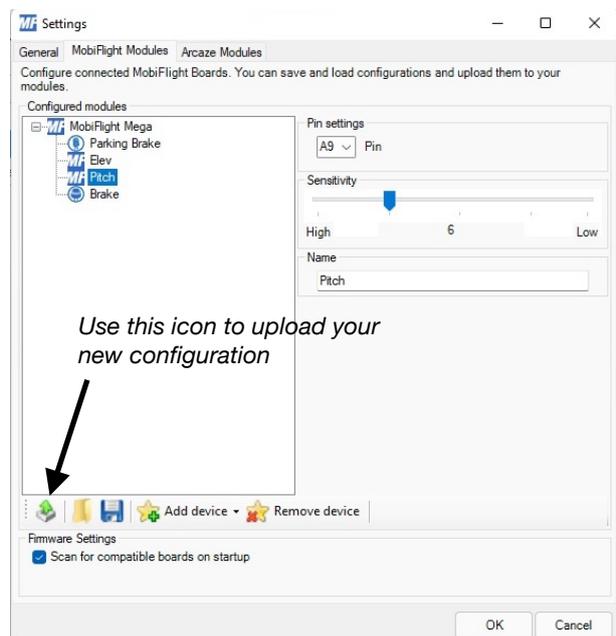
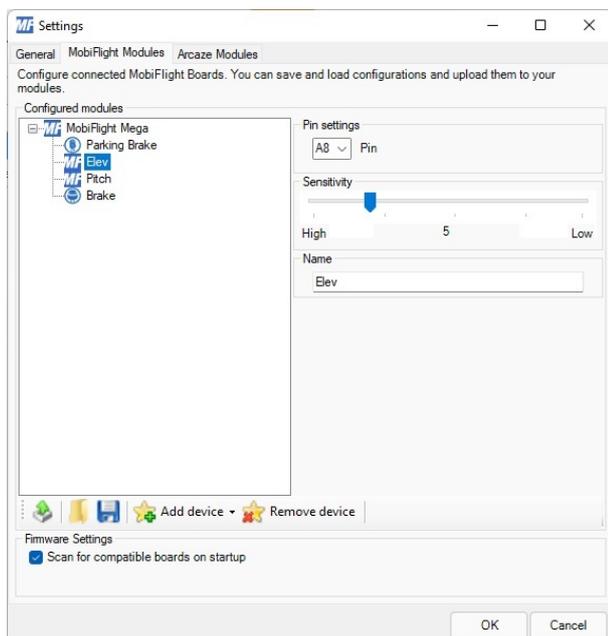


Connecting up the Arduino:

1. For the rotary potentiometer, connect the centre terminal to one of the analog inputs - I used A9 for the aileron control. The two outer terminals are connected to 5v and GND - it doesn't matter which is which, but if you find later that the yoke works backwards you can swap these over.
2. For the slide potentiometer there are again three terminals, and you will have to use a multimeter to find which terminal is the equivalent of the centre terminal of a rotary pot. Once you work out which it is, connect it to one of the analog pins (I used A8). The other two go to 5v and GND as above.
3. Connect the Arduino to your pc with a USB cable. When you run Mobiflight it will recognise the Arduino and set it up as a Mobiflight Module.

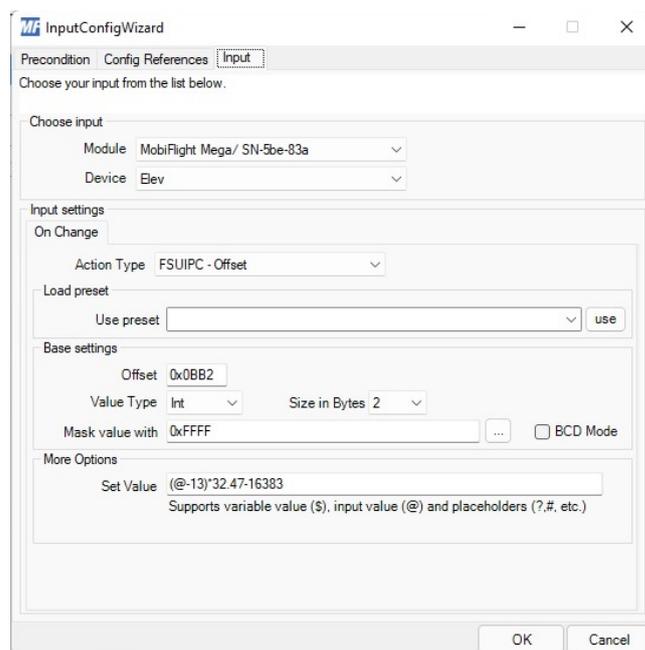
Setting up Mobiflight:

1. Go to Settings > Extras and choose the MobiFlight Modules tab. You need to set up your two potentiometers as devices, so go to Add Device near the bottom of the dialog box, click it and choose Analog Input. Give it a suitable name (eg Ailerons) by filling in the field on the right hand side. Repeat this for the other potentiometer, and then upload your new configuration by clicking the Upload icon at the bottom right.



2. Go to Extras > Settings in Mobiflight and tick the check box 'Logging'. Select 'Debug' from the drop-down list just to the right. This will log the inputs and display them in the split screen at the bottom of the MobiFlight window. You can now check the action of the two potentiometers - it will say something like "Config found for AnalogInput: Elevator (CHANGE=>512" when the yoke is moved. The actual number displayed will vary between 0 and 1023 in an ideal world, but in practice (and depending on whether your yoke actually physically limits the full travel of the pots) it may be slightly less, say 13 to 1022. Make a note of the numbers as they are needed to calibrate the yoke correctly.

3. This is the trickiest part. While the potentiometers have now been recognised by Mobiflight they have yet to be configured correctly for the flight sim software as inputs. In the main MobiFlight window choose the Inputs tab. Start with the Elevator input, click the Edit.. field on the right hand side to display the configuration wizard, and select the input tab on this new dialog box.

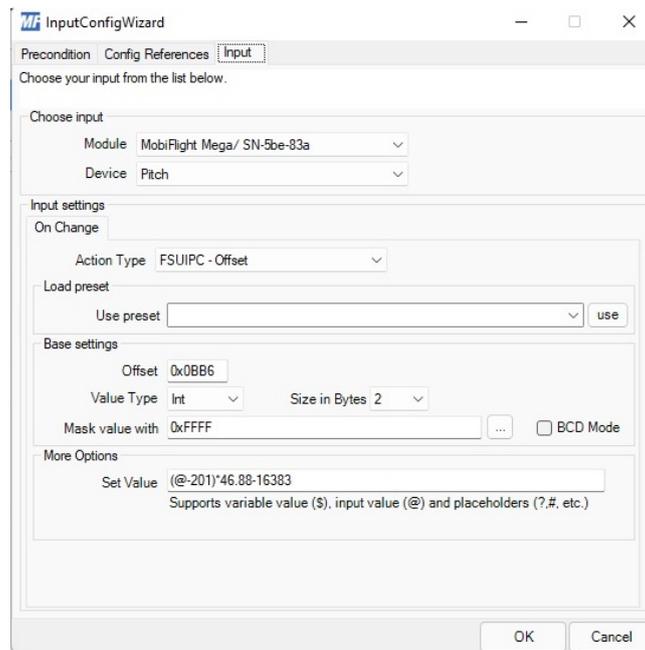


Select the Arduino (MobiFlight Mega is shown above) followed by the input module (Elev). Then select the Action Type from the drop-down list - in this case FSUIPC - Offset. This is where the XPUIPC file comes into play - it allows us to specify settings that will address particular commands such as elevators and ailerons. The offset for elevators in X-Plane is denoted by the hex value 0BB2, which is what needs to be entered in the Offset field, followed by 2 in the Size field.

Next comes the maths. X-Plane expects values for 0BB2 to range from -16383 to +16383, but the potentiometer, in theory at least, passes values of 0 to 1023 to Mobiflight. In practice, my potentiometer actually only gives a range of 13 to 1022, so:

1. Actual pot range = $1022 - 13 = 1009$
2. Ratio of X-Plane values to Pot values = $16383 / 1009 = 16.237$
3. Because we want the joystick to work in both the up and down directions we actually need to go from -16383 to +16383, so the ratio needs to be doubled - $16.237 \times 2 = 32.47$
4. The Set Value field therefore needs to show $@(-13) * 32.47 - 16383$. This actually says "Take the value sent by the pot (@), subtract 13 to move it slightly back to the zero point, divide it by the total ratio (32.47) and then subtract 16383 to go to the middle of the range."

5. Repeat this process for the aileron potentiometer (note that in the screenshot below this has incorrectly been given the name 'Pitch' and it should read 'Ailerons').



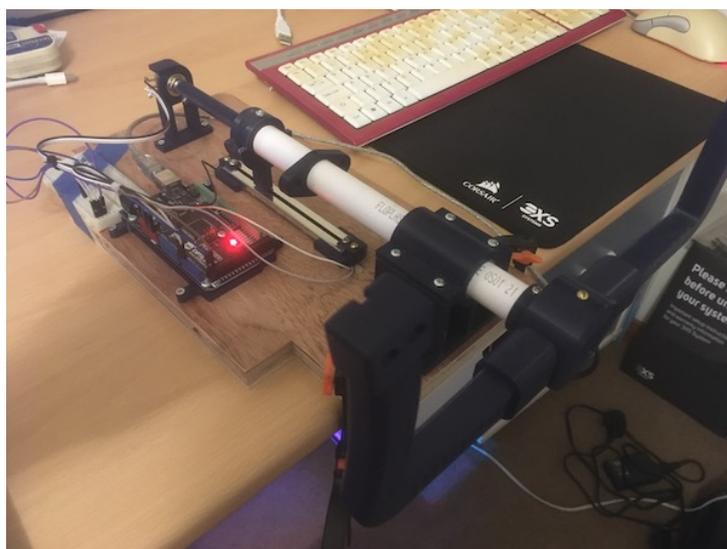
This is slightly different in that the prototype yoke is limited to about 180°, which gives logged values of 210 to 909 rather than 0 to 1023. So:

1. Range = 909 - 210 = 699
2. Ratio = 16383/699 = 23.44
3. Double the ratio = 46.88

Therefore the Set Value reads $(@-201)*46.88-16383$.

Testing:

Save the Mobiflight config file, run X_Plane and check the Mobiflight is connected (green ticks at bottom of window). Check the response of the yoke. If it works backwards you need to reverse the connections of the 5V and GND wires to the potentiometers. If the movement of the yoke is either very limited, or very exaggerated, then the Set Value expression has been entered incorrectly. If it doesn't work at all, check that Mobiflight is running, and that the Arduino connections including the usb lead are still connected.



Note: The Arduino (SKU: DFR0191) and rotary pot (SKU: 102841) came from thepihut.co.uk, and the slide pot from cpc.farnell.com, order code RE06841.