

Sign in



AlanChatham / UnoJoy Public

UnoJoy! allows you to easily turn an Arduino Uno (or Mega or Leonardo) into a PS3-compatible USB game controller

GPL-3.0 license

459 stars 127 forks Branches Tags Activity

Star

Notifications

Code

Issues

40

Pull requests

Actions

Projects

Wiki



master



Go to file

Code



AlanChatham Update MegaJoy.h

3 months ago



DoubleJoy

Update DoubleJoy_usb_stuff.c to give a ...

3 months ago



Drivers

Updated MegaJoy! for OSX Yosemite

10 years ago



LeoJoy

Moving all my code from Google Code

10 years ago



MegaJoy

Update MegaJoy.h

3 months ago



UnoJoy

Merge pull request #34 from cmavc/ma...

3 months ago



UnoJoySteeringWheel

Add files via upload

4 years ago



LICENSE.txt

Create LICENSE.txt

3 months ago



OSX Deployment Collator.bat

Updating release collation scripts

10 years ago



README.md

Update README.md

3 months ago



Windows Deployment Collator.bat

Updates to README, Windows Deploym...

5 years ago




README



GPL-3.0 license



UnoJoy

UnoJoy! allows you to easily turn an Arduino Uno (or Mega or Leonardo) into 

Getting Started

Hi there! Welcome to using UnoJoy! 

UnoJoy lets you use a plain, unmodified Arduino Uno to create native USB joysticks. It is a three-part system:

Drivers - Needed to re-flash the Arduino's USB communication chip

Software - The UnoJoy library for Arduino

Firmware - Code to load onto Arduino's USB communication chip

In order to make UnoJoy work, you'll need to take care of all three parts. We're here to make that process as easy as possible.

Drivers

=====

In the UnoJoy directory, there are installer files for the drivers you'll need for the DFU bootloader. Choose the correct one for your OS:

WindowsUnoJoyDriverInstaller.exe

LionUnoJoyDrivers.pkg

SnowLeopardUnoJoyDrivers.pkg

On Windows, you'll also need to download and install Atmel's FLIP tool:

<https://www.microchip.com/developmenttools/ProductDetails/flip>

On Linux you'll need to install dfu-programmer. you can get it by typing to

```
sudo apt-get install dfu-programmer
```

or

```
sudo aptitude install dfu-programmer
```

depending on your distribution.

You can also build it from source: <https://github.com/dfu-programmer/dfu-pr>

You also have to make the flashing script runnable by typing:

```
chmod +x TurnIntoAJoystick.sh
```

into your terminal when in UnoJoy directory.

Software

=====

To get started, first, go to the UnoJoyArduinoSample

folder. Open up UnoJoyArduinoSample and upload that code to your Arduino.

Next, test to make sure that it's working, we have a Processing sketch to test your controller without having to cycle through the process of going reflashing the firmware back and forth. Go to <https://processing.org/> to download and install Processing. Then you can run the UnoJoyProcessingVisualizer sketch.

Note You'll need to install the ControlP5 library in order for the UnoJoyProcessingVisualizer to work. To install that, in Processing, go to Sketch -> Import Library -> Add Library... search for ControlP5, install it, then possibly restart Processing.

Second Note The compiled stand-alone processing applications seem to have stopped working, and with Java being a pain, they're not currently being supported.

You should see a representation of the controller, and if you ground any of the pins between 2 and 12, you should see buttons on the controller light up. Now, we move onto the hardware!

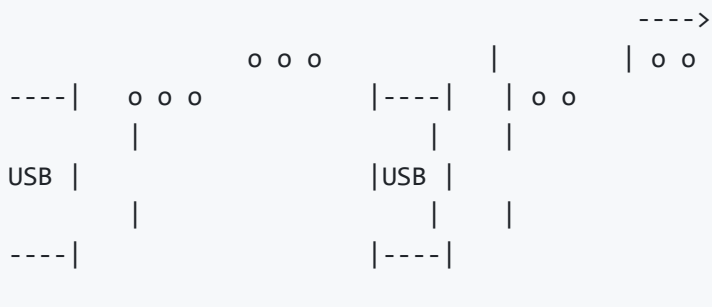
Hardware
=====

Now that we have the proper code on the Arduino, we need to reprogram the communications chip on the Arduino. In order to do this, you need to first put the Arduino into 'Arduino UNO DFU' mode. The official documentation for this is here

<http://arduino.cc/en/Hacking/DFUProgramming8U2>

----HOW TO PUT YOUR ARDUINO INTO DFU MODE----

You do that by shorting two of the pins on the block of 6 pins between the USB connector. Using a piece of wire or other small metal object, connect the 2 pins closes to the USB connector together. (the ones that turn from o to | in the diagram)



It should disconnect (be-dun.) and reconnect (buh-din!) and now show up to your system as 'Arduino UNO DFU'. In OSX, you will get no feedback from your computer, but

the lights on the Arduino will stop flashing.

ONCE YOU ARE IN DFU MODE
=====

Once the Arduino is in DFU mode, to update the firmware, simply click:

Windows: TurnIntoAJoystick.bat

OSX: TurnIntoAJoystick.command

Linux: ./TurnIntoAJostic.sh

IMPORTANT: Once you update the firmware, you'll need to unplug and plug the Arduino back in for it to show up with the new firmware - it'll just hang out in DFU mode until you do.

When you plug the Arduino in again now, it will show up to your computer as an 'UnoJoy Joystick'. You can check this by doing the steps in the next section.

HOW TO CHECK WHICH MODE YOU ARE IN
=====

On Windows 7, you can check it out by going to

Start->Devices and Printers

and you should see it there under 'Unspecified'

In Arduino mode, it will appear as 'Arduino UNO (COM 23)'

In DFU mode, it will appear as 'Arduino UNO DFU'

In UnoJoy mode, it will appear at the top as 'UnoJoy Joystick'

On OSX, you should see it:

Snow Leopard: Apple->About This Mac->More Info...->USB

Lion: Apple->About This Mac->More Info...->System Report->USB

You may need to refresh (command-R) to see it update.

In Arduino mode, it will appear as 'Arduino UNO'

In DFU mode, it will appear as 'Arduino UNO DFU'

In UnoJoy mode, it will appear at the top as 'UnoJoy Joystick'

On Linux, you can type lsusb to your terminal.

In response you'll get list of all connected usb devices.

From there you should find:

In Arduino mode, you should see a device named Arduino Uno etc.


In DFU mode, you should see a device named Atmel corp. etc.

In UnoJoy mode, you should see a device named Cygnal Integrated Pro


****Update****

As of 2021, there has been success using the [jstest-gtk](https://m. tool to test the joystick once it has been turned into a joystick

Using the Deployment Collators

There are a couple of programs for creating a quick release zip file. 
The OSX one may or may not work; I haven't had a Mac in years so I haven't
To use the Windows one, you'll need to install 7zip (<https://www.7-zip.org/>
then add it to your system or user path (search for Environment Variables,
the Path variable and add the 7zip folder). Then you should just be able to
Windows Deployment Collator.bat.

Steering Wheel Update, courtesy of Mustavfa Cem Avci

With this update you can now make your own steering wheel and pedal set for 
you always wanted to play with controller!


You need an MPU6050 IMU sensor for this project, simply connect SDA and SCL
VCC -> 5V GND->GND

or you can use the SDA SCL pins on Arduino close to the GND pin if you need
You can find the program for the gaming 900 degree wheel set on the Uno
Upload the program and follow the instructions!

Feel free to contact me for any problems on the program : cem.avci@gazi.edu
Mustafa Cem Avci, 17/06/2020
LONG LIVE OPEN SOURCE!

YouTube: <https://www.youtube.com/watch?v=Rq2QivBzshs>

Releases 4

 **UnoJoy! Release for Windows** Latest
on Mar 27, 2019

[+ 3 releases](#)

Packages

No packages published

Contributors 3



AlanChatham Alan Chatham

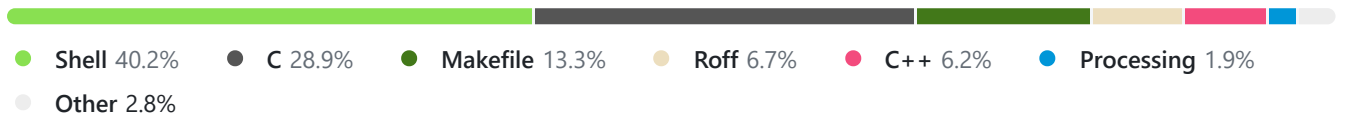


Erbonator3000 Eero Prittinen



cmavc Cem Avci

Languages



[Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact](#) [Manage cookies](#) [Do not share my personal information](#)



© 2024 GitHub, Inc.