CS7.1 Software Light Dimmer Mod

Overview

This mod replaces the hardware based light dimmer knob in the electronics box with a software-controlled system. Arduino supports PWM (pulse width modulation) signals which can be relayed through a pwm switch. In this mod, we are using the same MOSFET relay switch which is used for the AirDrop mod .

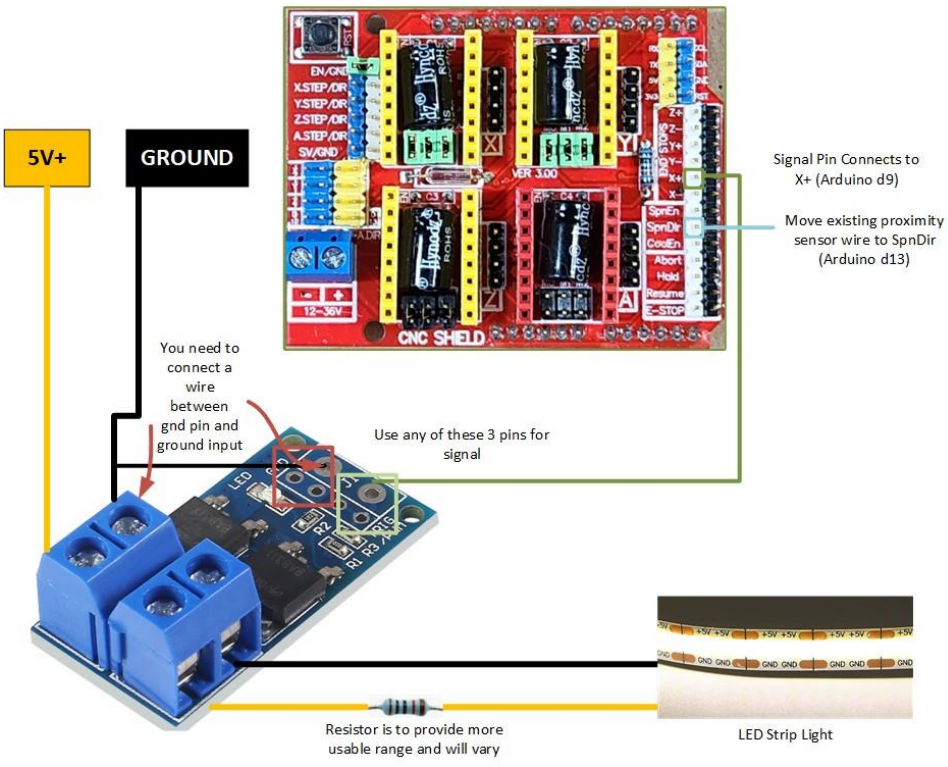
Hardware Required

The only additional hardware required for this mod is the MOSFET pwm switch/relay which is the same one we use for the Airdrop mod.

https://amzn.to/3Ac9f7Y

Installation

Unfortunately, not all pins are supported on the Arduino for PWM signals so we need to “trade” one of the currently used pins which supports PWM for one that doesn’t. On the Arduino Shield, pin D13 (labeled SpnDir) will become the new pin for proximity sensor (currently X+ or D9) and we will use the X+ pin for the PWM dimmer signal.



Inline Resistor

The inline resistor as seen in the screenshot above is not required but can help to provide a better range of light depending on your setup. For instance, if you are using clear PLA in your light ring, you may want to add a 1k resistor. For a white PLA light ring, a value closer to 200 ohm may be more appropriate. It is recommended to try it without a resistor first and see if you are able to get the right amount of light while still having range to adjust up or down. The range of the dimmer control is 0-255.

Configuring the Arduino

Finally, you need to make two changes in the Arduino code to enable and use this feature.

The first parameter you need to change is named UseArduinoPWMDimmer and the value needs to be set to true as seen in the screenshot below.

The second value you can change is the default light brightness value. Note that once you enable this mod, you will have an option to adjust this value in the software, so it is fine to leave the value as set.

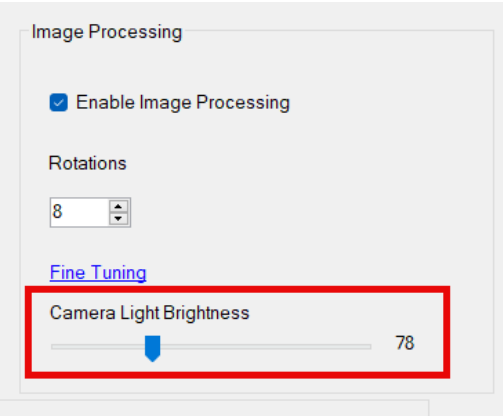
```
#define UseArduinoPWMDimmer true //if you have configured your hardware for to use arduino PW

#if UseArduinoPWMDimmer == false
  #define FEED_SENSOR 9 //the proximity sensor under the feed wheel
  #define CAMERA_LED_PWM 13 //NOT USED
#else
  #define FEED_SENSOR 13 //the proximity sensor under the feed wheel
  #define CAMERA_LED_PWM 9 //the output pin for the digital PWM
#endif

#define CAMERA_LED_LEVEL  78 //camera brightness if using digital PWM, otherwise ignored
```

Once you have updated the code, you need to upload it to your Arduino. When that is complete, close the Arduino IDE and launch the software (v1.1.45 or later)

You will now see an option in the software to adjust the light brightness in the Configuration panel. It is best to hit the “Preview” button to turn on the camera and then adjust the slider until you achieve the appropriate brightness.



Be sure to save your configuration changes once you have adjusted this setting.