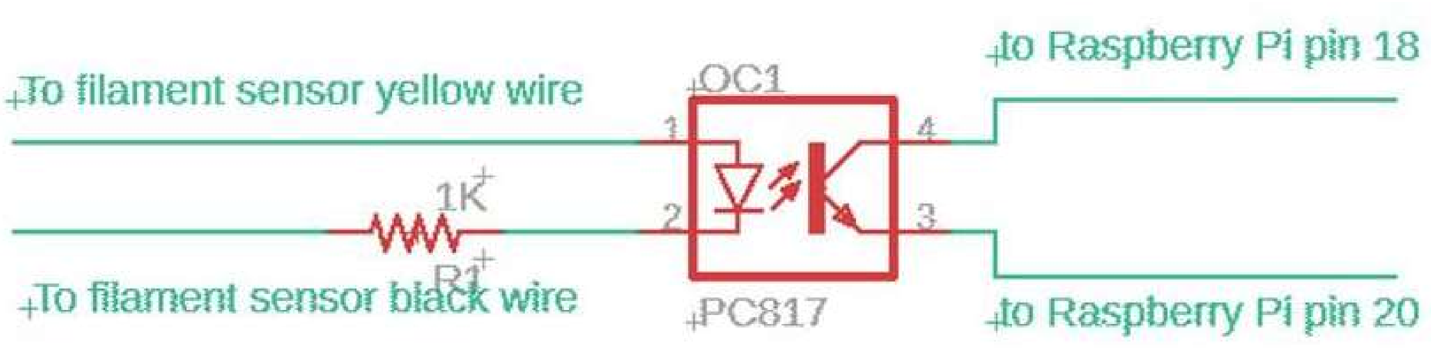
filament runout sensor octoprint

Just in case you ever need it, here is a schematic and the instructions for the circuit:

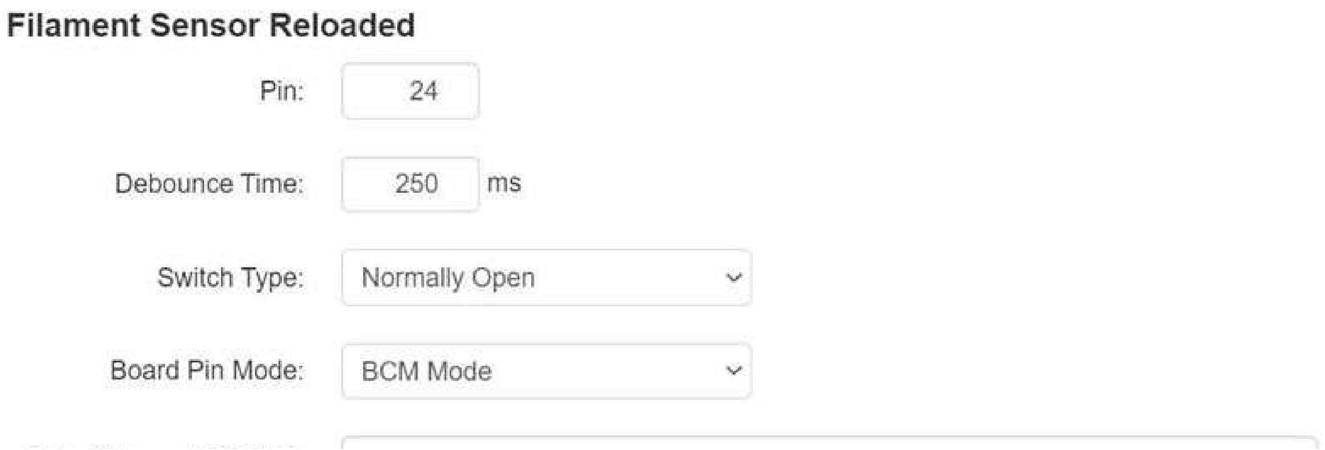


Using a PC817 optical isolator. The optical isolator minimizes any impact to the native signaling from the filament sensor to the controller while also isolating it from the Raspberry Pi.

The colors may be different for different printers but on my Creality Ender 5 Plus you need to splice in to the yellow (or signal) wire and connect it to pin 1 on the PC817 (the anode side of the LED) and splice the black (or ground) wire and connect it to a 1K resistor and then to pin 2 (the cathode side of the LED) and connect the transistor side to the Raspberry Pi as shown below. I used wired connectors to expose the signal from the filament sensor so I did not have to actually splice the original wiring.

Then connect the collector (Pin 4) to pin 18 (GPIO.5 / BCM 24) on the Raspberry Pi and the emitter (pin 3) to pin 20 (ground) on the Raspberry Pi. And then configure the filament sensor plugin using those pins.

Here are the settings for OctoPrint:



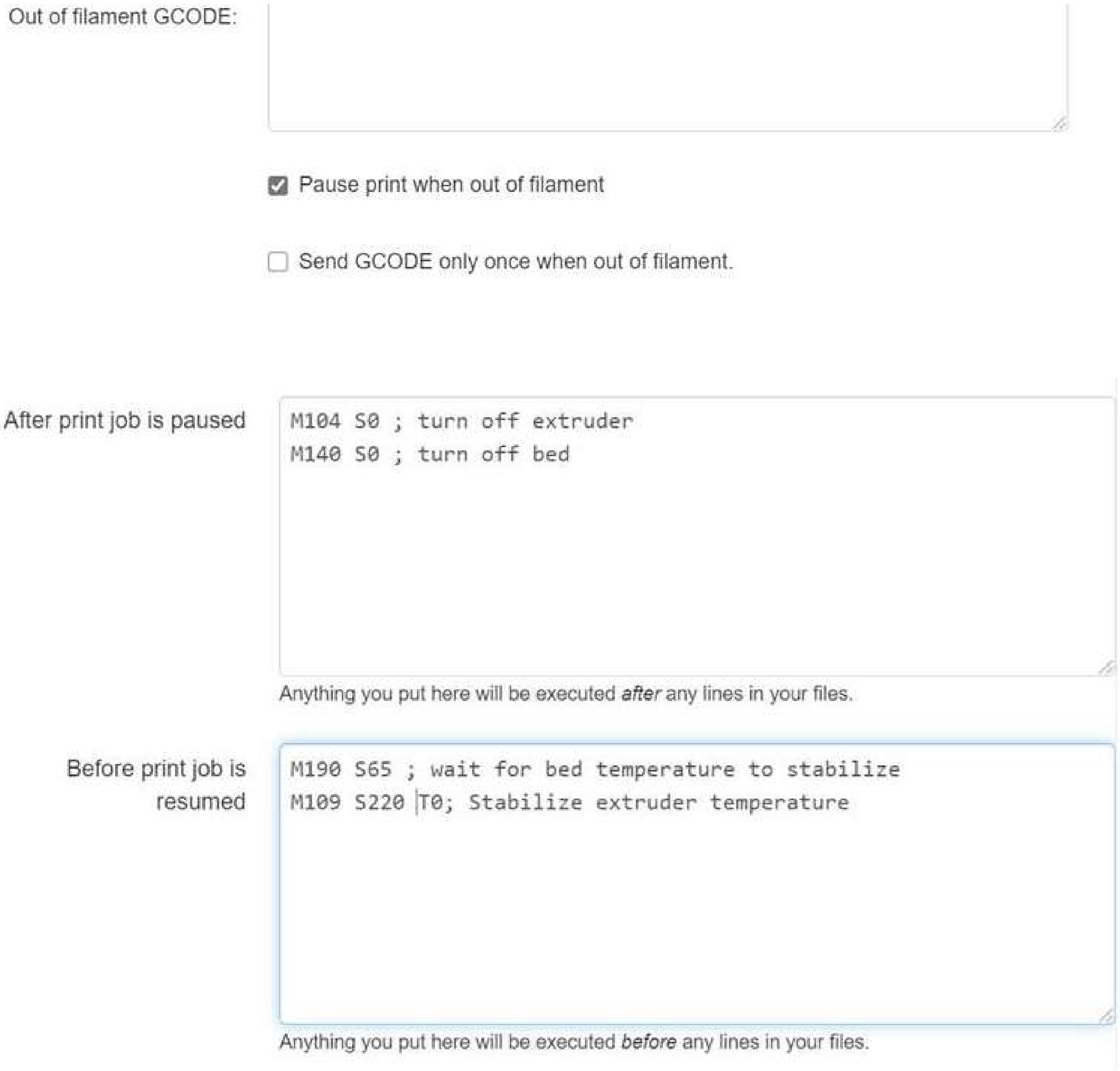
Filament Sensor Reloaded

Pin: 24

Debounce Time: 250 ms

Switch Type: Normally Open

Board Pin Mode: BCM Mode



Out of filament GCODE:

☑ Pause print when out of filament

☐ Send GCODE only once when out of filament.

After print job is paused

```
M104 S0 ; turn off extruder
M140 S0 ; turn off bed
```

Anything you put here will be executed *after* any lines in your files.

Before print job is resumed

```
M190 S65 ; wait for bed temperature to stabilize
M109 S220 T0; Stabilize extruder temperature
```

Anything you put here will be executed *before* any lines in your files.

In the before print job is resumed script it really needs to know what the previous bed temp (M190 S65) and the extruder temp (M109 S220) is but I have not found a way to get those values from OctoPrint.