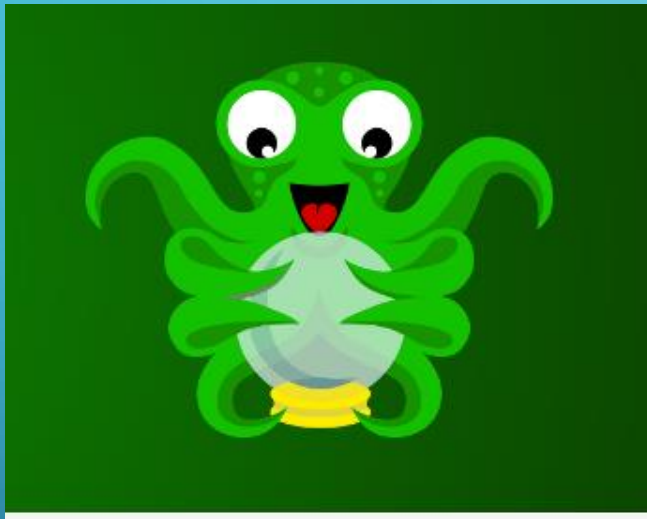


A decorative graphic on the left side of the slide, consisting of white lines and circles on a blue background, resembling a circuit board or data flow diagram.

# RASPBERRY PI


UTILIZING A RASPBERRY PI 3 TO CONTROL 3D PRINTER

## UTILIZING PI 3 OCTOPRINT IMAGE TO CONTROL ENDER 3 ROBOT



<https://octoprint.org/>

## ENDER 3 ROBOT



Roll over image to zoom in

by SainSmart  
**SainSmart x Creality Ender-3 3D Printer, Resume Printing V-Slot Prusa i3, for Home & School Use, Build Volume 8.7" x 8.7" x 9.8"**  
★★★★☆ 60 customer reviews  
| 49 answered questions

Price: **\$239.00** ✓prime  
Used & new (2) from \$190.07 ✓prime

**Specifications for this item**

Part Number	Ender-3
Brand Name	SainSmart
EAN	4897093840219
Model Number	Ender-3
Specification Met	

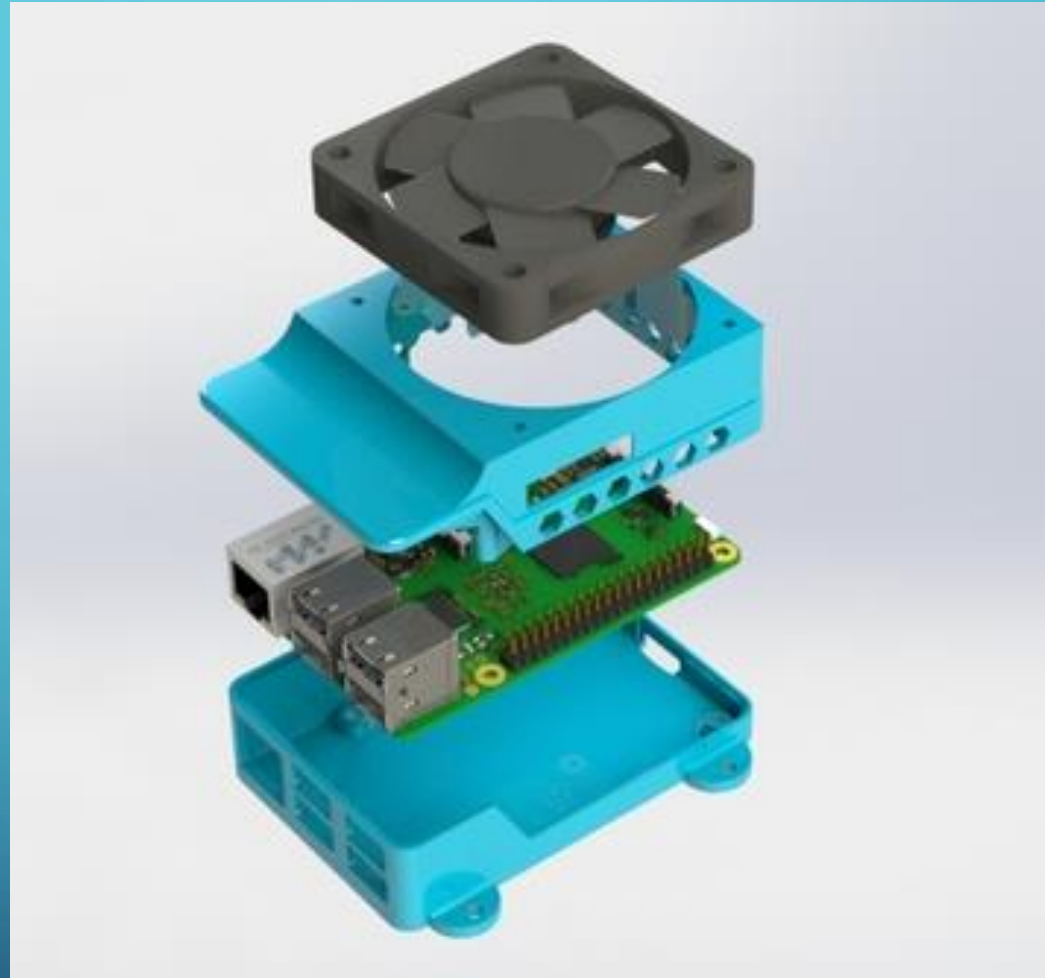
[See more product details](#)

Item arrives in packaging that reveals what's inside. To hide it, choose **Ship in Amazon packaging** at checkout.

[Link to Amazon website](#)

<https://www.sainsmart.com/collections/3d-printing/products/sainsmart-x-creality3d-ender-3-3d-printer>

PI CASE WITH FAN



<https://www.thingiverse.com/thing:3183658>



## 24MM BUTTON MOUNT & SWITCHLAMPS

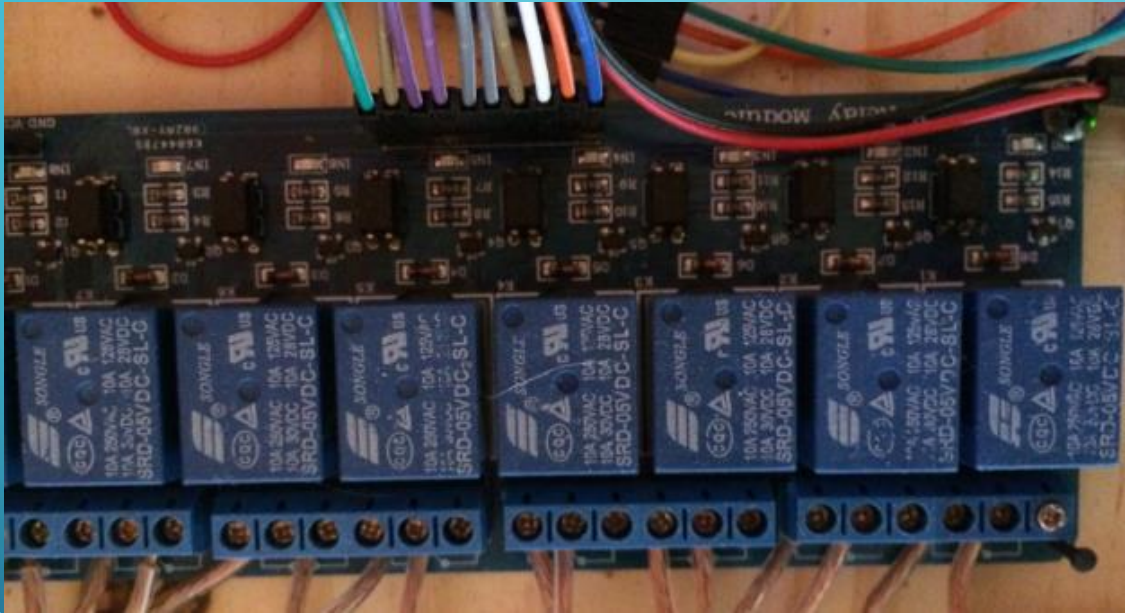


<https://www.thingiverse.com/thing:3172566>



Link to Amazon for Switchlamps

## SAINSMART 8 CHANNEL RELAY



### SainSmart 8-Channel Relay Module

by [SainSmart](#)

★★★★★ ▾ 85 customer reviews

| 14 answered questions

List Price: \$48.40

Price: **\$13.18** & **FREE Shipping** on orders over \$35.

[Details](#)

You Save: **\$4.92 (27%)**

**In Stock.**

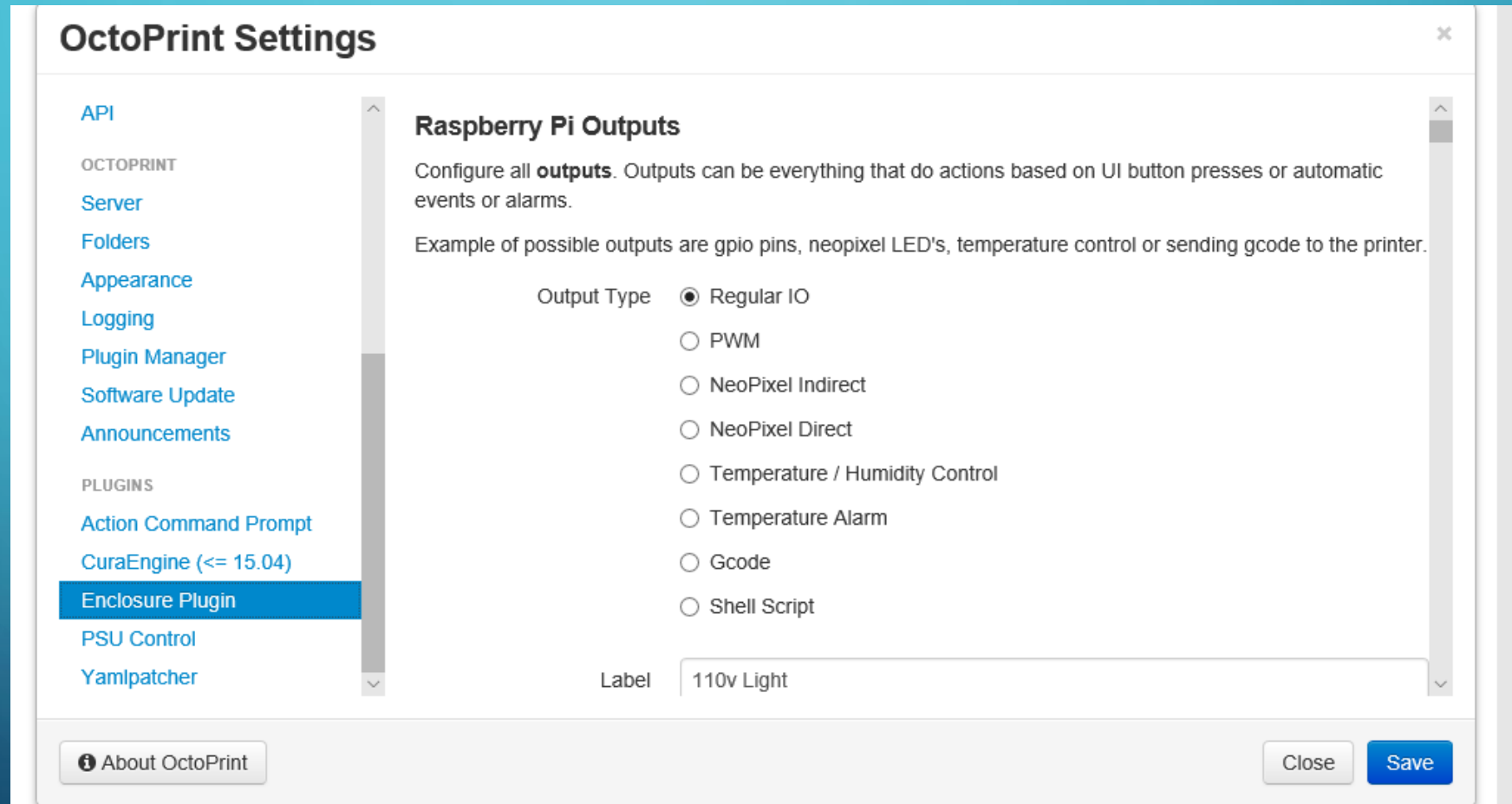
Ships from and sold by Amazon.com. Gift-wrap available.

**Want it tomorrow, April 5?** Order within **5 hrs 19 mins** and choose **One-Day Shipping** at checkout. [Details](#)

[Click to open expanded view](#)

- 5V 8-Channel Relay interface board, and each one needs 15-20mA Driver Current
- Equipped with high-current relay, AC250V 10A ; DC30V 11A
- Standard interface that can be controlled directly by microcontroller (Arduino , 8051, AVR, PIC, DSP, ARM, ARM, MSP432, TTL logic)
- Indication LED's for Relay output status

## OCTOPRINT ENCLOSURE PLUGIN – TO CONTROL SOME OUTPUTS



The screenshot shows the OctoPrint Settings window with the 'Enclosure Plugin' selected in the left sidebar. The main content area is titled 'Raspberry Pi Outputs' and contains the following text and options:

**Raspberry Pi Outputs**

Configure all **outputs**. Outputs can be everything that do actions based on UI button presses or automatic events or alarms.

Example of possible outputs are gpio pins, neopixel LED's, temperature control or sending gcode to the printer.

Output Type

- Regular IO
- PWM
- NeoPixel Indirect
- NeoPixel Direct
- Temperature / Humidity Control
- Temperature Alarm
- Gcode
- Shell Script

Label

At the bottom of the window, there is an 'About OctoPrint' button on the left, and 'Close' and 'Save' buttons on the right.

<https://plugins.octoprint.org/plugins/enclosure/>

## OCTOPRINT CONFIG.YAML

[http://docs.octoprint.org/en/master/configuration/config\\_yaml.html](http://docs.octoprint.org/en/master/configuration/config_yaml.html)

### config.yaml

If not specified via the command line, the main configuration file `config.yaml` for OctoPrint is expected in its settings folder, which unless defined differently via the command line is located at `~/.octoprint` on Linux, at `%APPDATA%/OctoPrint` on Windows and at `~/Library/Application Support/OctoPrint` on MacOS. If the file is not there, you can just create it - it will only get created by OctoPrint once you save settings that deviate from the default settings.

Note that many of these settings are available from the "Settings" menu in OctoPrint itself.

#### Contents

- [config.yaml](#)
  - [Access Control](#)
  - [API](#)
  - [Appearance](#)
  - [Controls](#)
  - [Development settings](#)
  - [Estimation](#)
  - [Events](#)
  - [Feature](#)
  - [Folder](#)
  - [GCODE Analysis](#)
  - [GCODE Viewer](#)



## OCTOPRINT CONFIG.YAML – USING EVENTS TO CONTROL STATUS

<http://docs.octoprint.org/en/master/events/index.html#sec-events>

### Example

```
events:
  enabled: True
  subscriptions:
    - event: Disconnected
      command: python ~/growl.py -t mygrowlserver -d "Lost connection to printer" -a OctoPr:
      type: system
      enabled: false
    - event: PrintStarted
      command: python ~/growl.py -t mygrowlserver -d "Starting {file}" -a OctoPrint -i http:
      type: system
    - event: PrintDone
      command: python ~/growl.py -t mygrowlserver -d "Completed {file}" -a OctoPrint -i http:
      type: system
    - event: Connected
      command:
        - M115
        - M117 printer connected!
        - G28
      type: gcode
```

## OCTOPRINT CONFIG.YAML – ACTUAL CODE

In this case based on specific defined Octoprint events the Wiring Pi command to switch on status lamp is used

```
events:  
  enabled: true  
  subscriptions:  
    - command: gpio write 2 0  
      event: Connected  
      type: system  
    - command: gpio write 2 1  
      event: Disconnected  
      type: system  
    - command: gpio write 3 0  
      event: PowerOn  
      type: system  
    - command: gpio write 3 1  
      event: PowerOff  
      type: system  
    - command: gpio write 4 0  
      event: Error  
      type: system  
    - command: gpio write 4 1  
      event: Shutdown  
      type: system  
    - command: gpio write 6 0  
      event: FileAdded  
      type: system  
    - command: gpio write 6 1  
      event: PrintStarted  
      type: system  
    - command: gpio write 7 0  
      event: PrintCancelled  
      type: system  
    - command: gpio write 7 1  
      event: PrintStarted  
      type: system  
plugins:
```