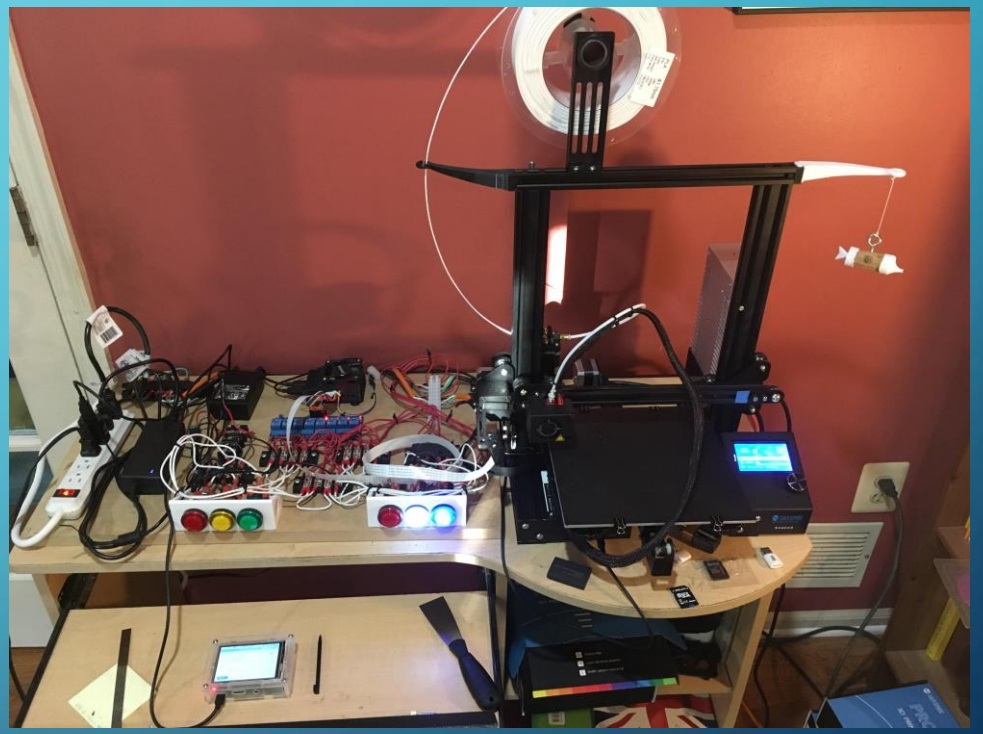
RASPBERRY PI

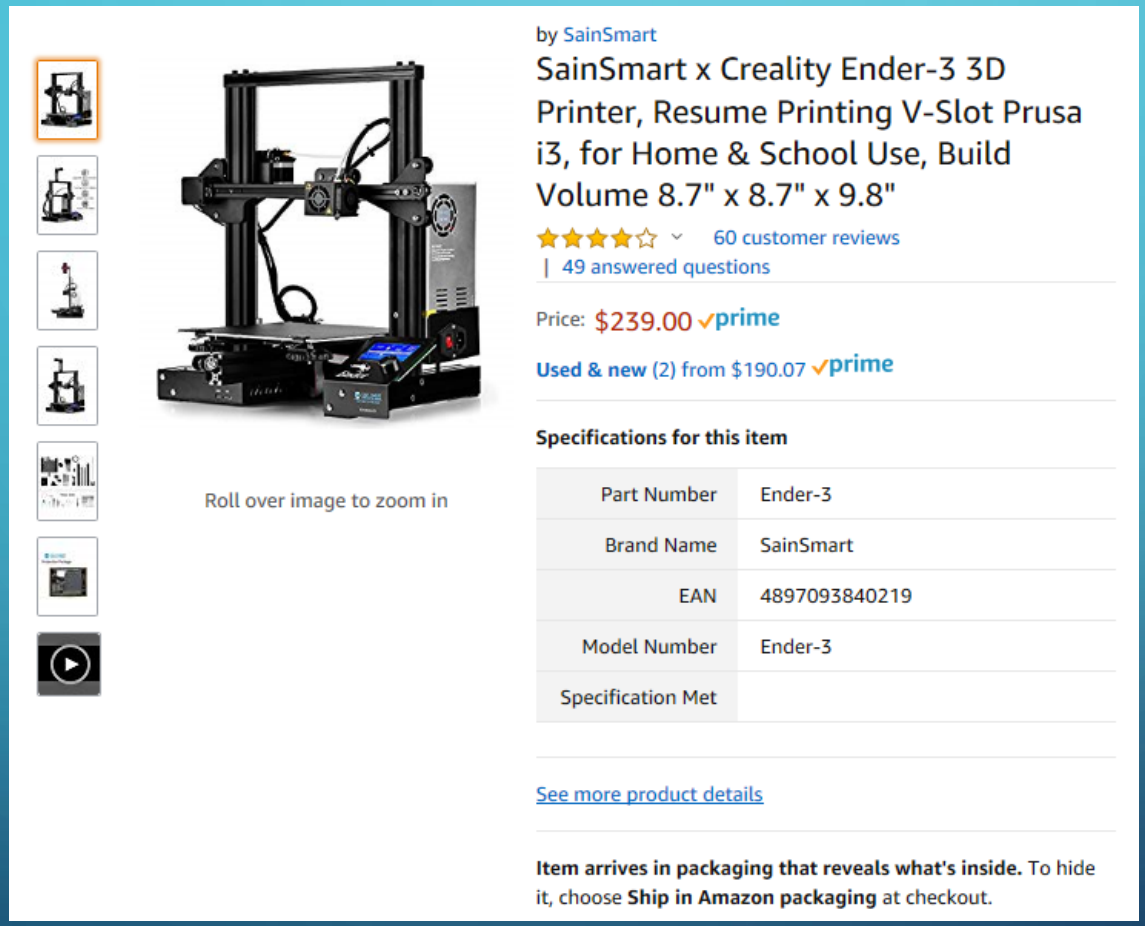
UTILIZING A RASPBERRY PI 3 TO CONTROL 3D PRINTER

UTILIZING PI 3 OCTOPRINT IMAGE TO CONTROL ENDER 3 ROBOT



https://octoprint.org/

ENDER 3 ROBOT



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"

4.1 out of 5 stars 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

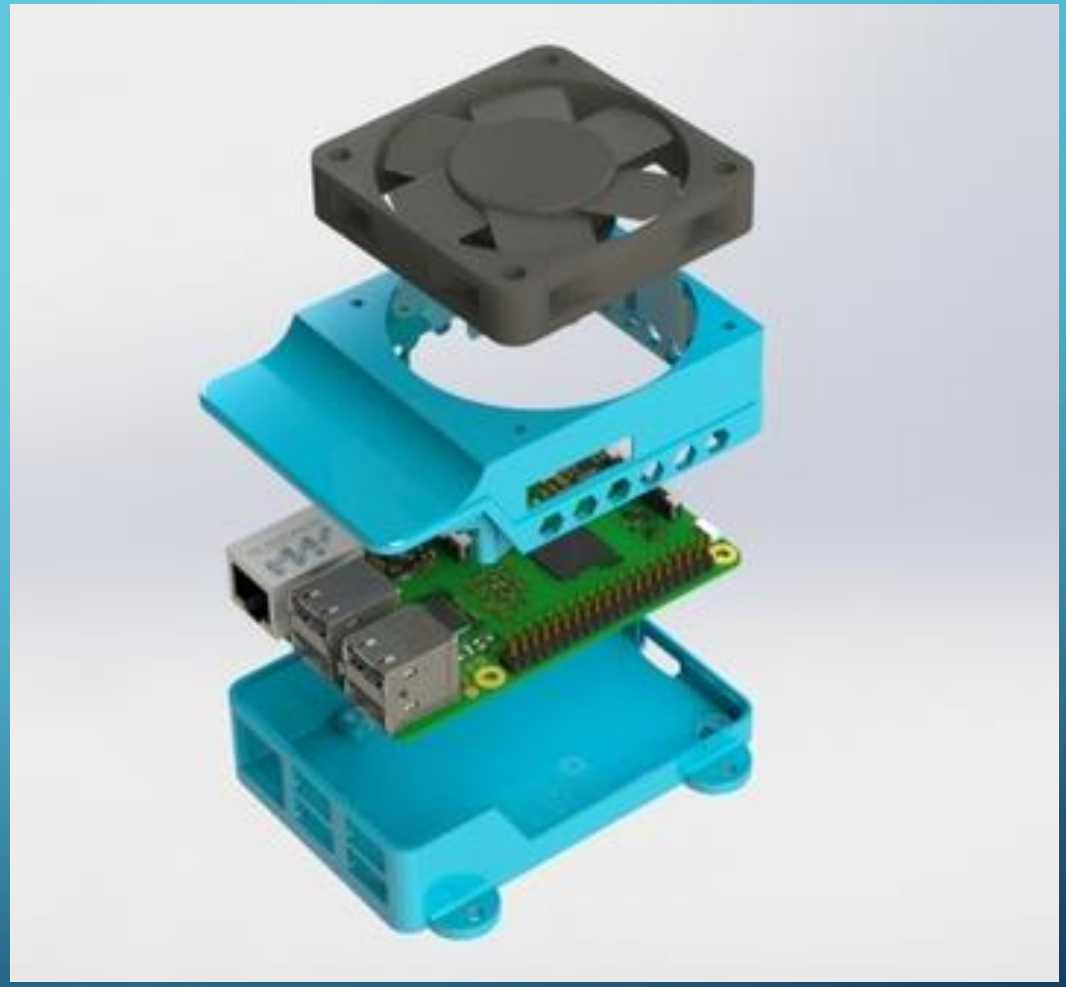
Item arrives in packaging that reveals what's inside. To hide it, choose **Ship in Amazon packaging** at checkout.

Roll over image to zoom in

Link to Amazon website

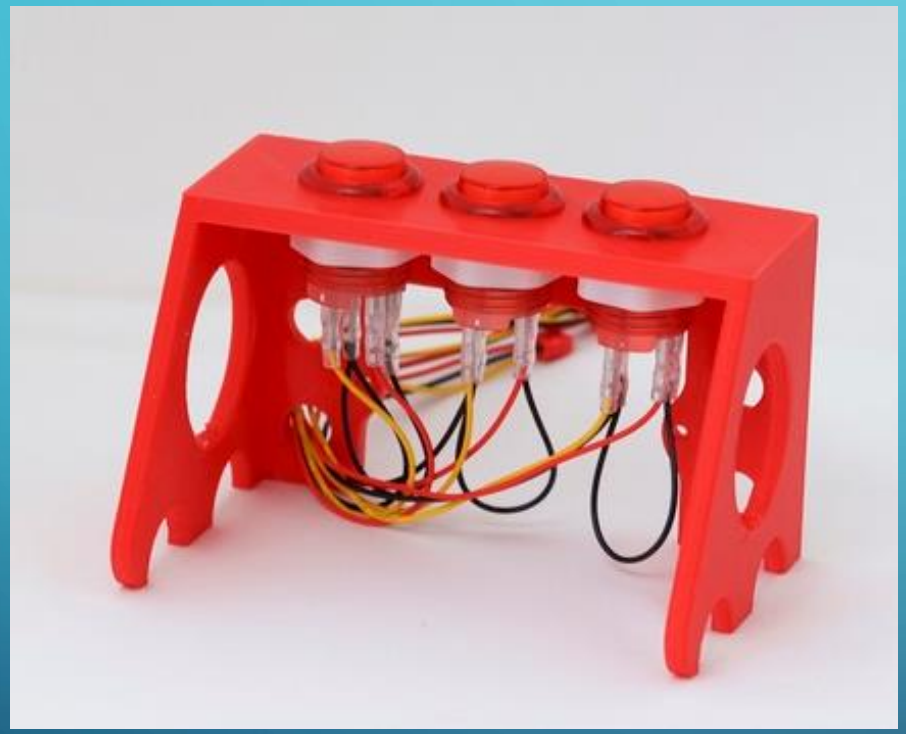
https://www.sainsmart.com/collections/3d-printing/products/sainsmart-x-creality3d-ender-3-3d-printer

PI CASE WITH FAN



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

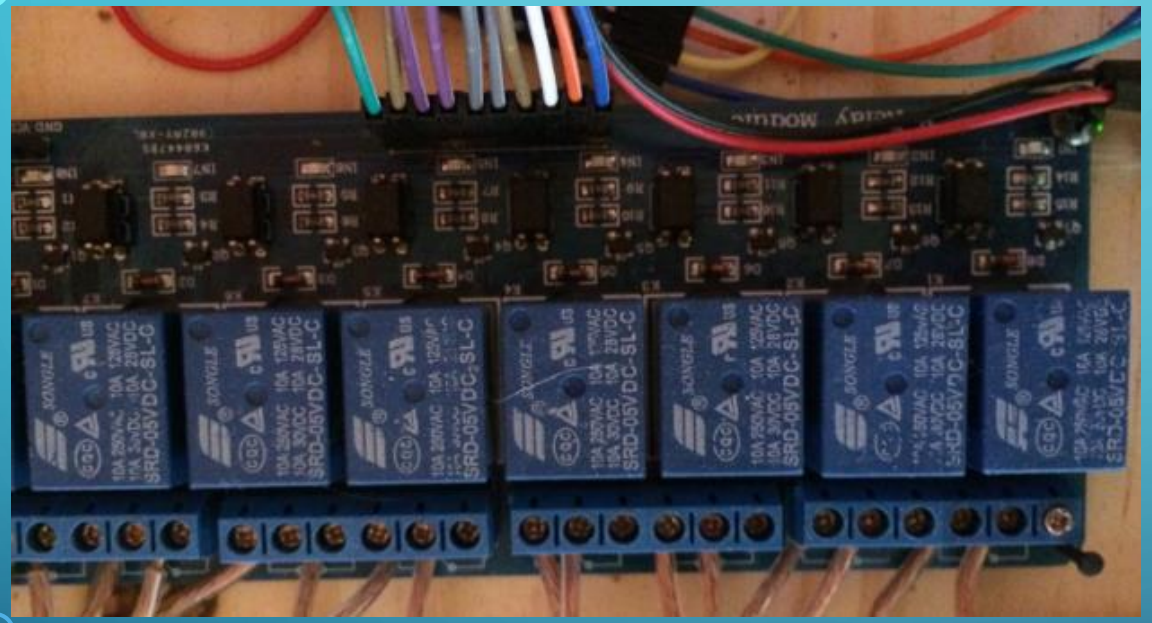
24MM BUTTON MOUNT & SWITCHLAMPS

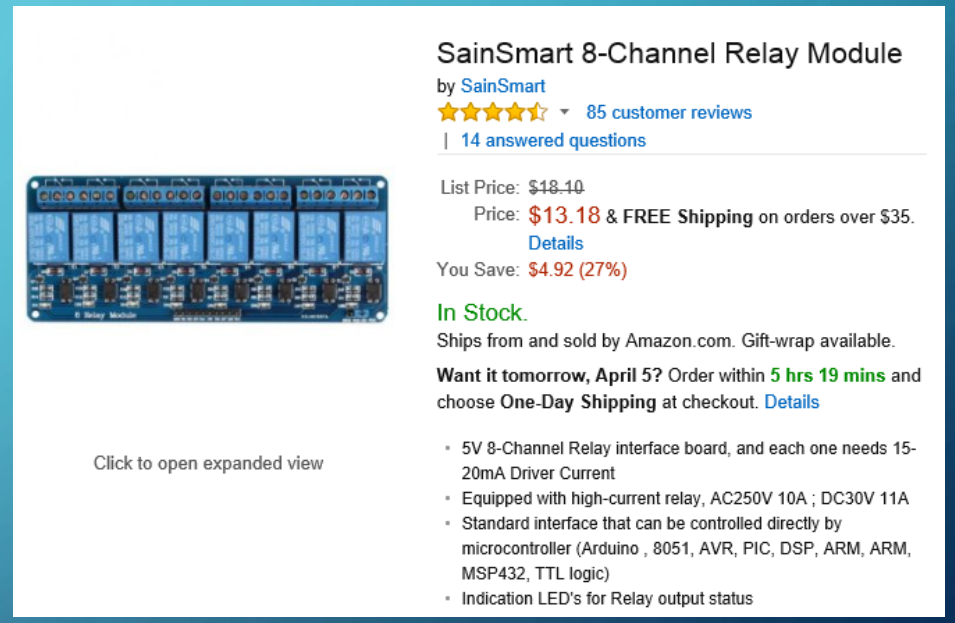


https://www.thingiverse.com/thing:3172566

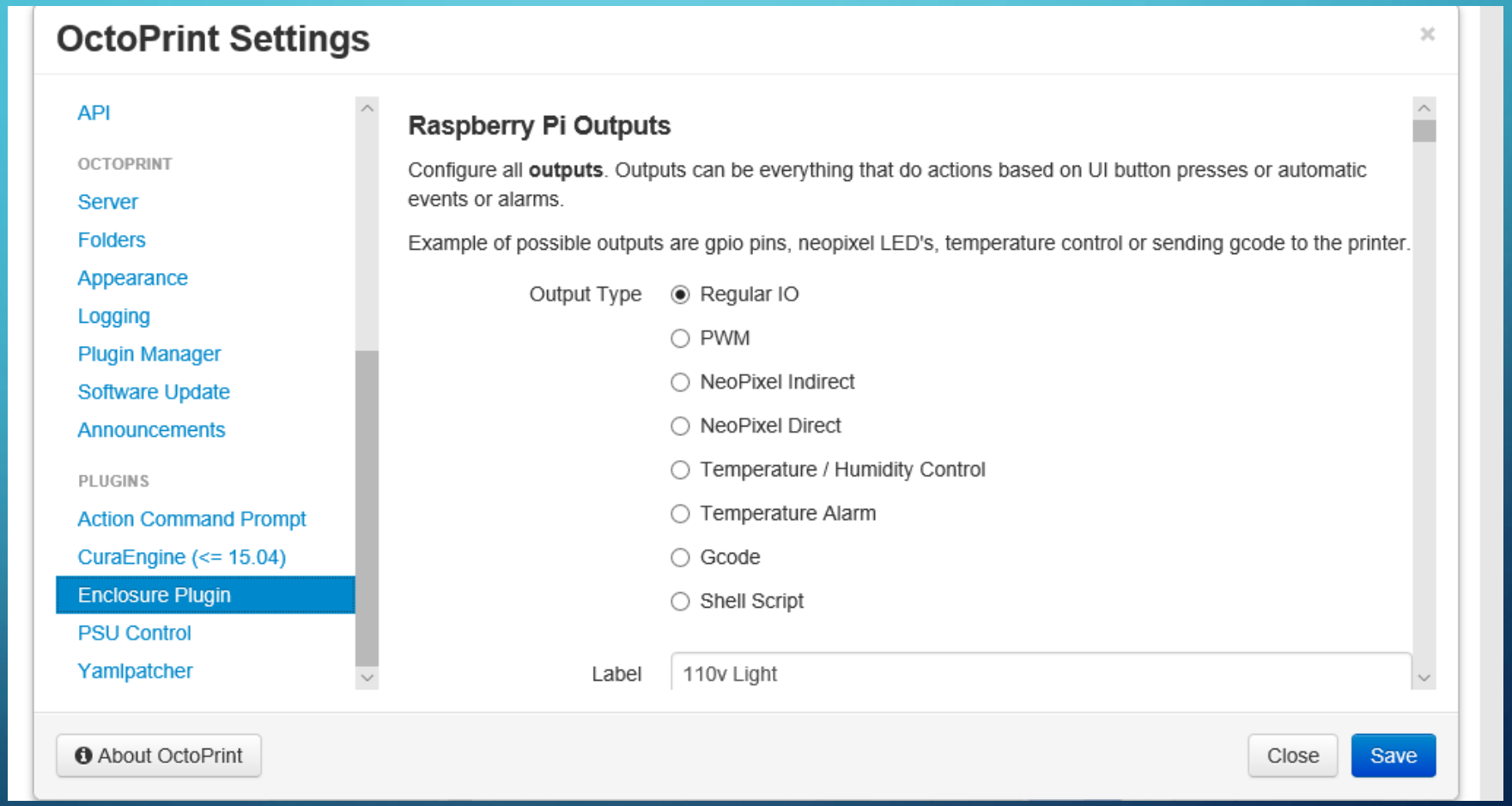
Link to Amazon for Switchlamps

SAINSMART 8 CHANNEL RELAY





OCTOPRINT ENCLOSURE PLUGIN – TO CONTROL SOME OUTPUTS



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

OCTOPRINT CONFIG.YAML

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

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 OctoPri
    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 htt
    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:
```