opentrons

You are currently viewing the documentation for the Opentrons OT-1. To view documentation for the OT-2, click here.

Hardware Modules

Heat Deck

The heat deck runs off the opensource platform Arduino, which is how you can control it's temperature. Our heat decks come automatically set to reach a temperature of 55 deg Celsius, but you can edit this value by editing the Arduino file.

Find our Heat Deck source code on GitHub here, and download.

Also download and install the Arduino IDE.

Open the file, and you will see detailed instruction for how to update the temperature. The overiew is that you simply set the number for what temperature you want, then upload that code to the Heat Deck.

Magbead

Setting up Hardware

Use your included DIY Mag Bead kit to configure the motor control board (see kit instructions).

Initializing Module in API

Just like a pipette, you need to set up and name your module.

instruments.Magbead (*mofset, name*)

mosfet - integer 0-5 (defaults to 0)
name - the name you want to call your module

```
mag_deck = instruments.Magbead(name='mag_deck')
```

Activate and Deactivate Magnets

To activate the magnets and raise the module's platform, run `.engage()`:

module.engage ()

```
mag_deck.engage()
```

To deactivate the magnets and lower the module's platform, run `.disengage()`:

opentrons

Chain Other Commands

Just like `aspirate()` and `dispense()` can be chained, you can chain `engage()` and `disengage()`, as well as the `delay()` if you don't want to do anything between engaging and disengaging the magnets.

```
mag_deck.engage()
mag_deck.delay(60)
mag_deck.disengage()

mag_deck.engage().delay(60).disengage()
```

You can call `delay()` with a `Pipette` or a `Magbead` module.

```
p200.delay(10)
mag_deck.delay(10)
```