**PitCam Readme and How-to Guide**

**Peeping Patty**

Written by Lee Trawick on 10/31/19

Coded by Jordan Ford, Neil Khera, Lee Trawick

Project contributors: Dr. Heather Jones, Chuck Whittaker, Red, Dr. Uland wong

**How do I login to the laptop?**

* The laptop user is pipedream and the password is pipedream

**How do I run the Pitcam?**

Open a terminal, cd into the /Pitcollector directory and run

* roslaunch pitcollector.launch

**How do I change the Pitcam sequence of positions/pans/tilts?**

* The pitcam movement sequence are saved in a json file in /Pitcollector/json\_sequences/

The JSON file looks like this:

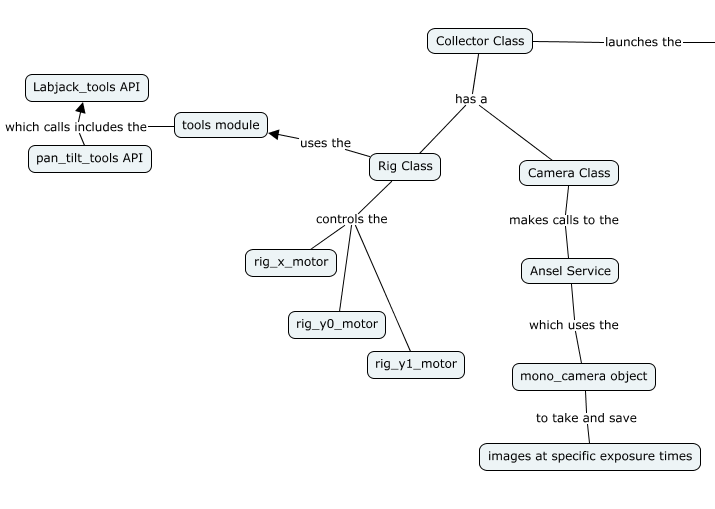
{  
 "number\_of\_bracketed\_images": "21",  
 "baseline\_exposure\_microseconds": "50000",  
 "bracketing\_factor": "1.2",  
 "position\_pan\_tilt":  
 [  
 [1, 0, -30]  
 ]  
}

You can change any value in the json file to change the Pitcam sequence

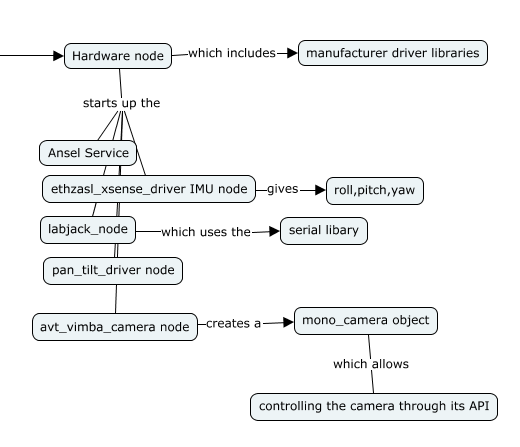
* Number\_of\_bracketed\_images sets how many images you take for each position-pan-tilt position.
* Baseline\_exposure\_microseconds sets the middle exposure where the bracketing will go above and below it
* Bracketing\_factor Dr. Uland Wong recommends we set this to 2 in the field to capture a wide range of light values.
  + Doubling the exposure time effectively equals 2 F-stops above or below the baseline exposure
* Position\_pan\_tilt is an array that reads [position, pan, tilt]
  + Position can be 1,2,3,4,5,6,7,8,9
  + Pan can be between [90, -90] degrees
    - The pan-tilt unit has higher limits but for this machine we set tighter constraints so the camera doesn’t hit the rig
  + Tilt can be between [30,-90] degrees
  + You can add as many [position,pan,tilt] arrays as you want.
    - For every array, the rig will go to that pose and take n bracketed photos

How is the ROS program structured?

The main controlling node is called the Collector. See below. It launches the Hardware node.

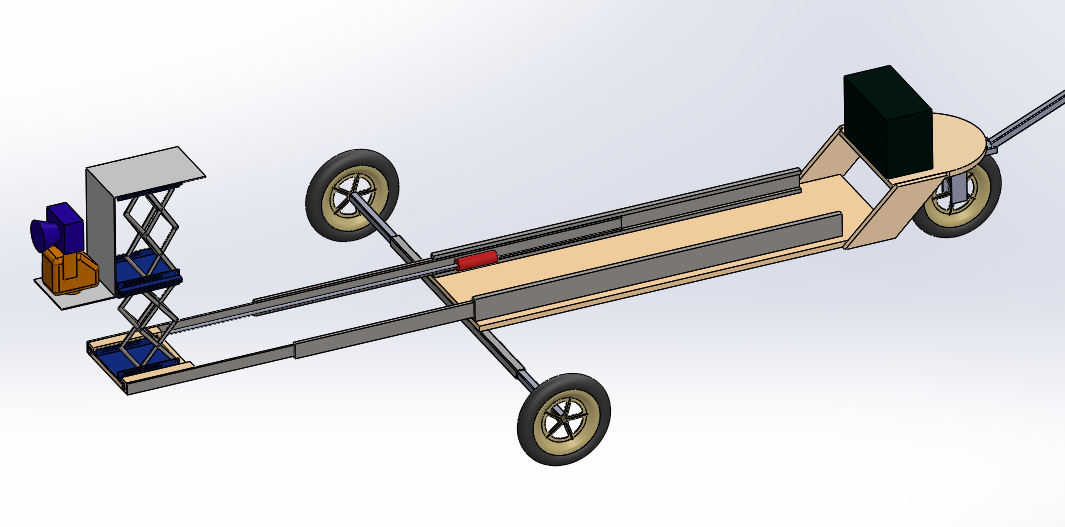


The Hardware node is mapped below



**What is the PitCam?**

* The PitCam was designed to safely take pictures of the inside of a sinkhole while sitting on its rim.

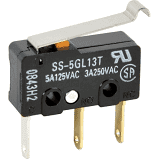


**How is the Pitcam built?**

* The pitcam moves horizontally and vertically with 3 independent motors
* The pitcam has 3 motors and 7 limit switches
  + One motor for horizontal movement
    - Horizontal axis has 3 limit switches at 0cm, 25cm, 50cm
  + One motor for the lower lift
    - The lower lift has a bottom switch and a top switch to tell the motor to not crush the scissor lift
  + One motor for the upper life
    - The upper life has a bottom and top limit switch

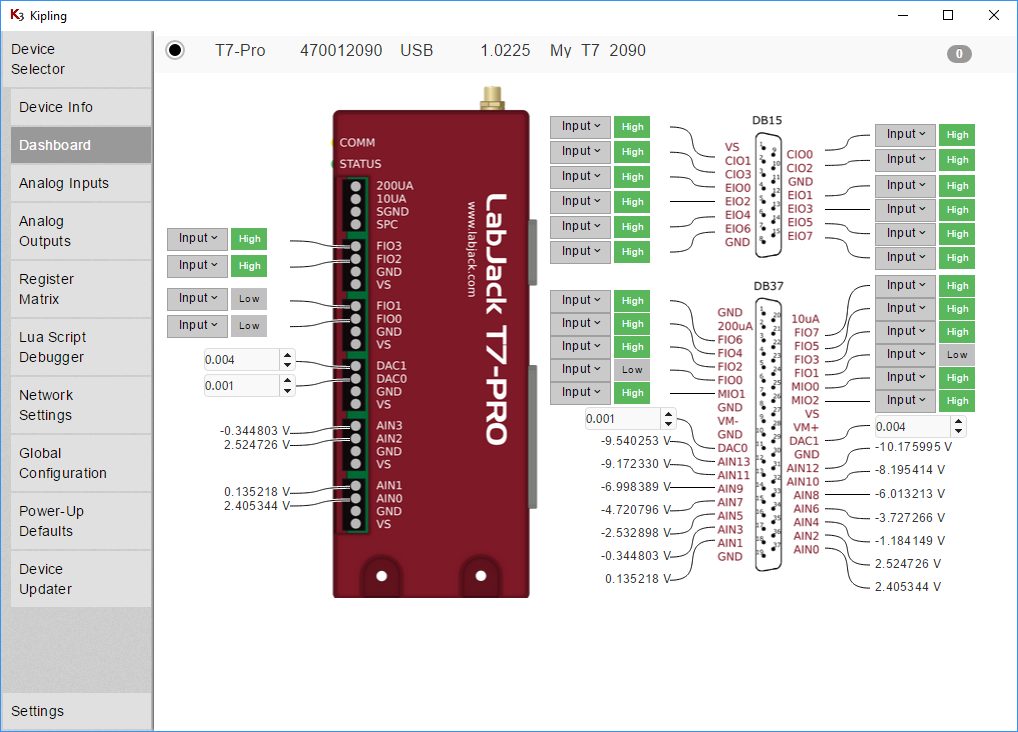
**What is a limit switch?**

* It’s a swich that gets triggered by physically pressing the switch into the body



**How do I control the Pitcam Motors and read the Limit switches?**

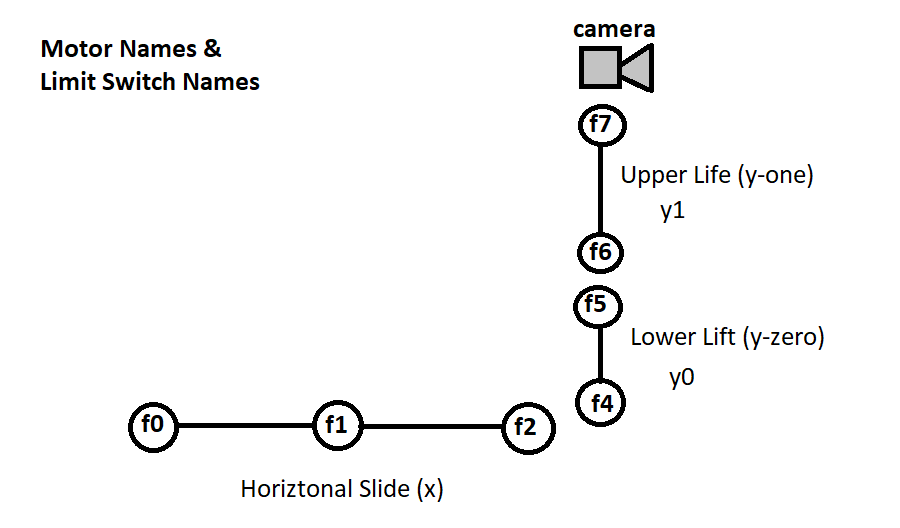
* The PitCam motors and limit switches are operated by a labjack. A The labjack API can be accessed by running “labjack\_kipling”



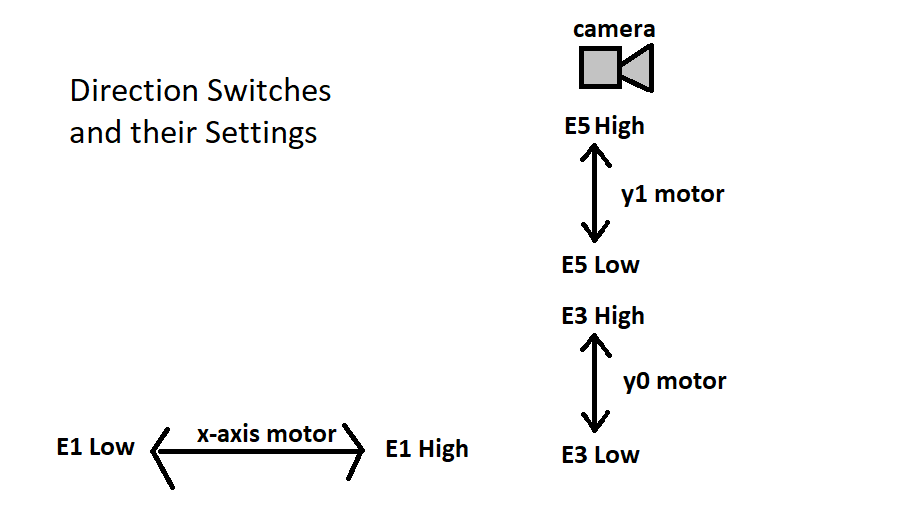
From the API, you can control all of the input output pins on the Labjack. The pitcam is only using the following pins

* Limit switches: F0,F1,F2 + F4,F5 + F6,F7
* Motor engage: E0,E2,E4
* Motor Direction: E1, E3, E5

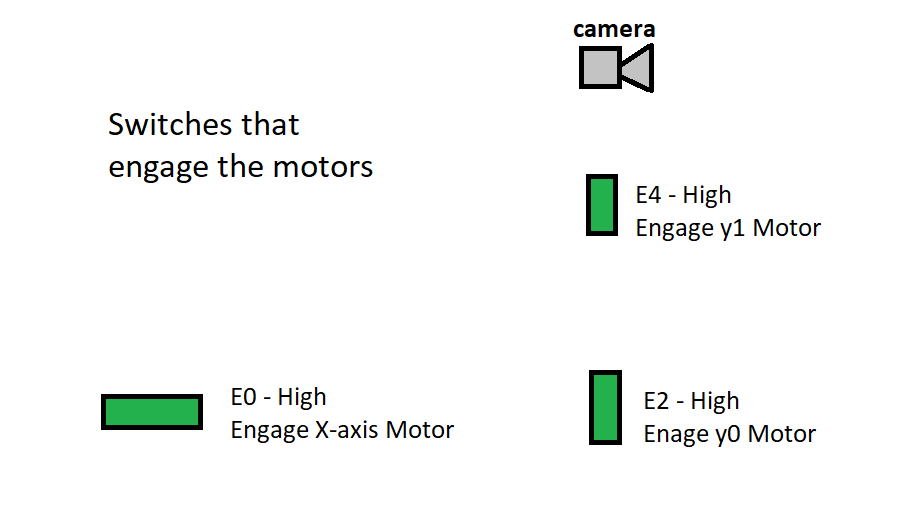
The limit switches are mapped below:



The motor directions are mapped below



The motor Engage Switches are mapped below:



The Pitcam can achieve 9 discrete positions. They are named below. (1 is home position)

