Project Proposal

1. Description of the overall concept
2. Goals - Base
3. Goals - Reach
4. Schematic drawing
5. Bill of materials including cost estimates.
6. Project schedule with checkpoints
7. Major Changes
8. Final Project
9. References

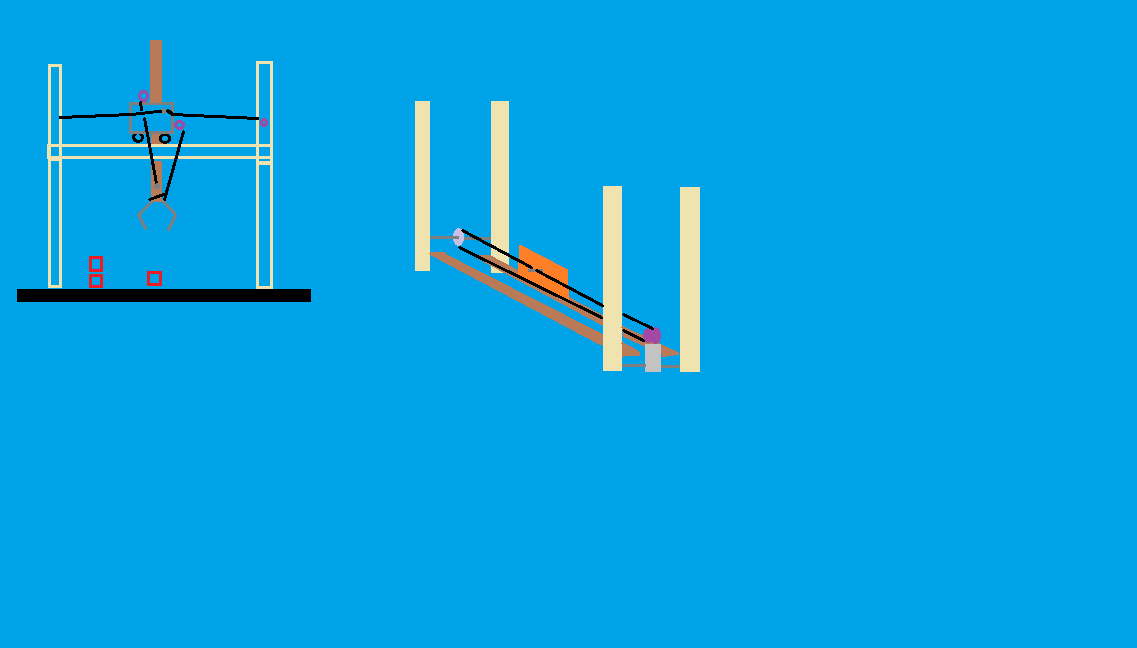
Project: Towers of Hanoi Robot Arm

Towers of Hanoi is a mathematical puzzle designed in 1883 by Édouard Lucas. It consists of three poles and a number of disks stacked from largest at the bottom, and smallest at the top. The goal of the puzzle is to move each disk from the first pole to the third pole, moving one disk at a time without placing a larger disk on top of a smaller one. Although there are multiple solutions to solve this puzzle, our hope is to program the Beaglebone Black and implement hardware to have a robot physically solve a Tower of Hanoi set. By building a rig and programming our board in Python, we hope to build a crane claw to solve the puzzle for us in the least amount of moves possible.

1. Overall Concept:  
   Build and program a crane claw that can calculate a minimum move order to Tower of Hanoi, and physically play out a prespecified Tower of Hanoi puzzle set in front of it.
2. Base Goals:
   1. take a button signal to start operations
   2. use a Towers of Hanoi solving algorithm to calculate the optimal move order
   3. robot uses claw to physically play out the calculated solution on the physical puzzle

1. Stretch Goals:
   1. use a sensor to map out the initial state of the puzzle setup in front of it
      1. use mapping data to adjust claw movement distances appropriately
      2. use mapping data to count initial number, positions, and dimensions of discs
   2. introduce Z-axis and/or angular position dimensions to environment mapping and maneuvering
   3. interference/obstacle detection & reaction
   4. allow for interrupt signal from start button during operation and some move reversal

1. Schematic Drawing:

(Old Design)  


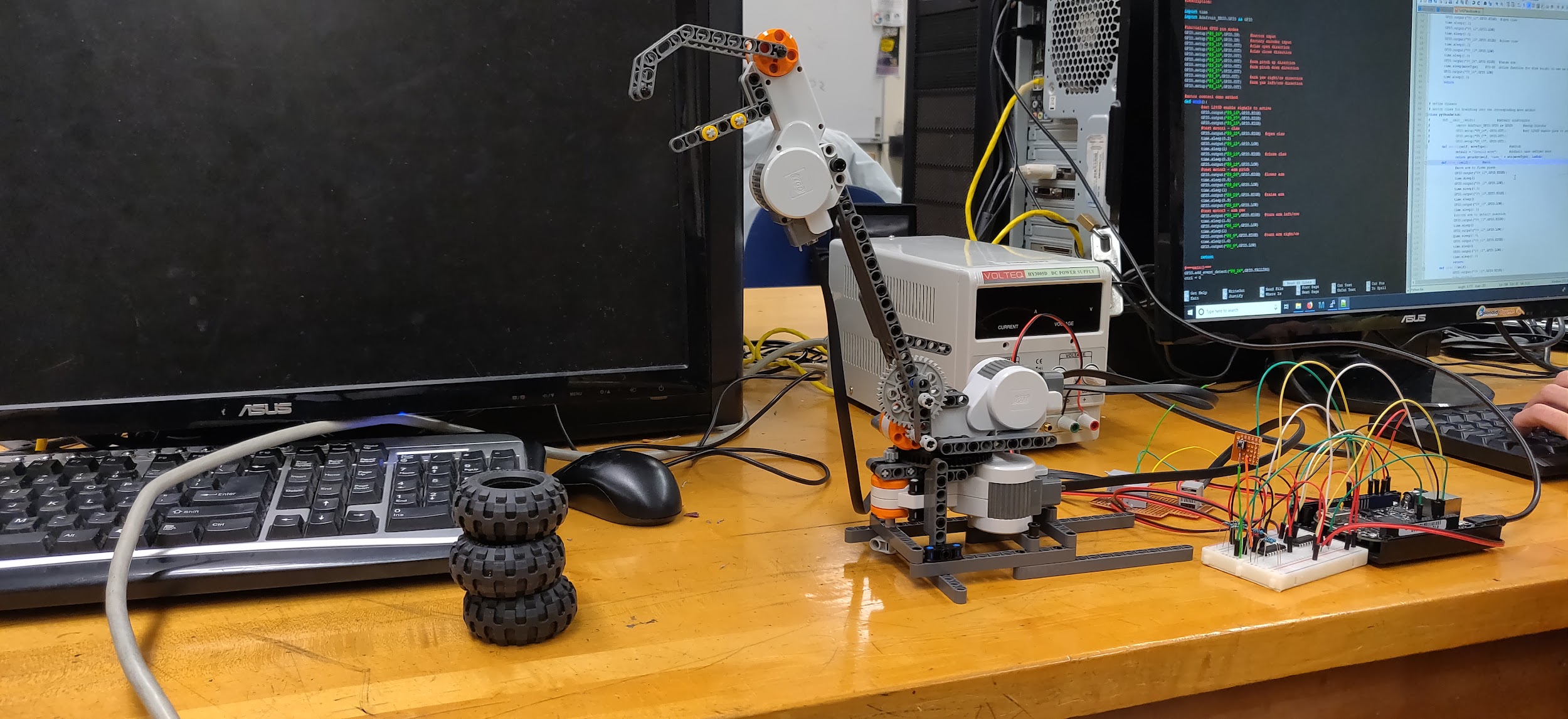
1. Materials and Costs:
   1. Tower of Hanoi puzzle <= ~$10
   2. Lego Mindstorm stepper motors, sensors, and connectors ~$25
   3. Wood 2x4 for structure ~$3
   4. Electrical wire ~$7
   5. Resistors ~$9
   6. SeeedStudios Motor Bridge Cape ~$20
   7. Prototype Breadboards ~$8
   8. Push Button Switch ~$1
   9. L293D H-Bridge ~$10
2. Project Schedule:
   1. week 1 (Mar. 11): obtain supplies
   2. week 2 (Mar. 18): test motors and sensors and program drivers
   3. week 5 (Apr. 8): assemble claw gantry; cut wood and wire to size, solder components together, nail/screw wood into position, etc.
   4. week 7 (Apr. 22): program Towers of Hanoi algorithm in Python
   5. week 8 (Apr. 27): prepare PowerPoint presentation and demo
   6. week 9 (May 6): project ready for presentation
   7. week 10 (May 13): presentation week
3. Major Changes:

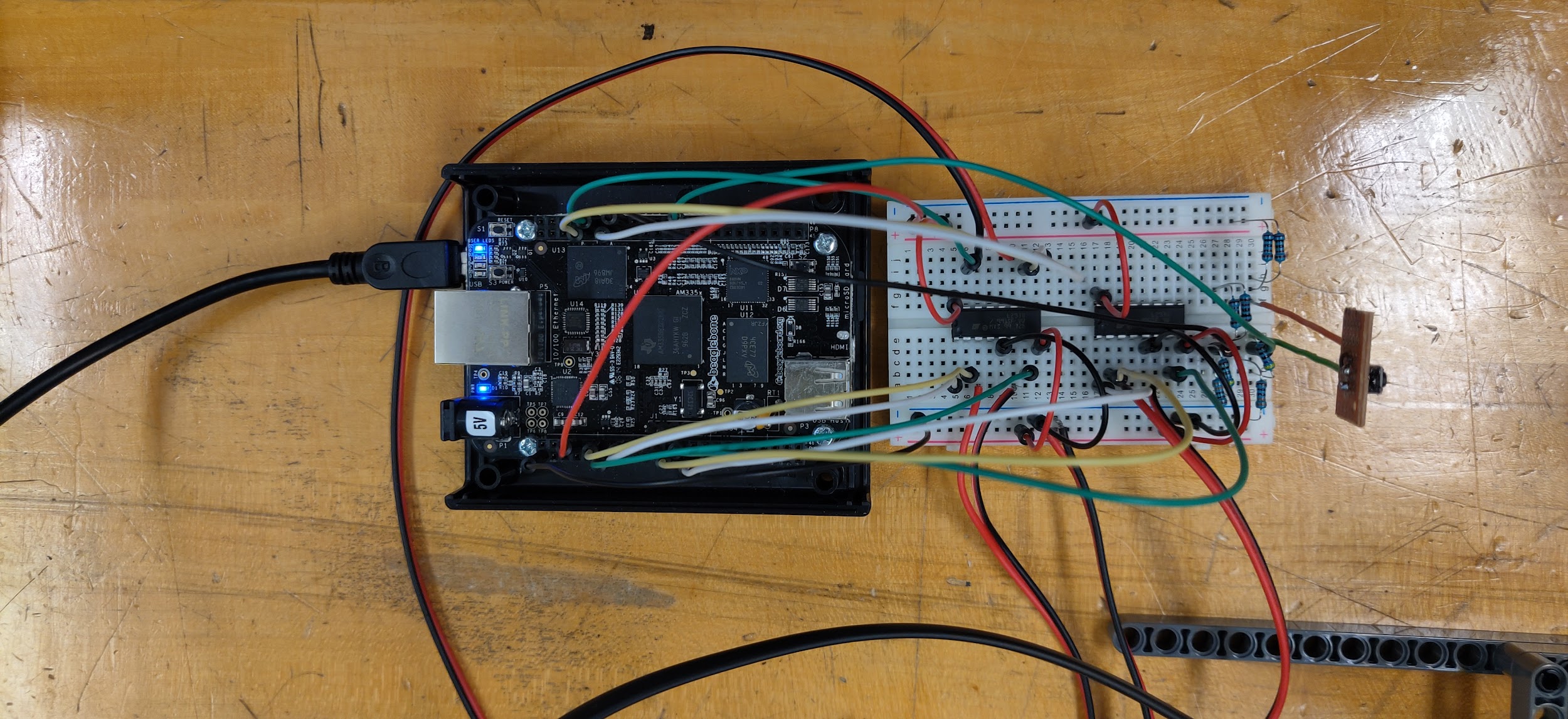
Instead of opting to create a structure that would move the claw on a cart, we ended up creating a robot arm that would play the towers of hanoi puzzle. We used tires from the Lego NXT set to act as our rings and had the robot place the tires in three columns in front of it, acting as if there were three imaginary poles to simulate our own puzzle. To control the arm, we built a new setup involving a breadboard, wires, resistors, and two L293D H-bridges, all hooked up to the Lego NXT motors and our BeagleBone Black.

Throughout the process of creating the motor arm, most of our available time was spent trying to operate our Motor Bridge Cape. Out of the box, the bridge cape required a lot of software fixing on arrival. The firmware shipped on the unit was non-functional, giving us multiple errors whenever we tried to run code. Additionally, there was little to no support or resources to be found online.

Additionally, having no prior experience with Python or the Adafruit\_BBIO Library was a challenge. Through the duration of the entire class, we had to learn how to work with the different pins on the BeagleBone Black. This meant figuring out what different libraries such as GPIO, PWM, ADC, etc. meant. We both only had prior experience coding in C++, making the transition to Python fairly daunting. Several class days were dedicated to learning the basic conventions and notations of Python.

1. Final Project:





Our final project was the robot arm in the picture above. Although our arm was not able to successfully complete the towers of hanoi puzzle, we believe that if given more time we could have made this happen. Using the Lego Mindstorms NXT kit provided by the classroom, we constructed the robot arm using the Lego Technic pieces and three of the motors. In order for the BeagleBone to communicate with these motors, we ended up having to purchase the NXT connection ports and solder them to a prototype board. We soldered additional wires connected to the ports so that we could connect it to our BeagleBone. Using our breadboard, we finished wiring up the necessary wires, resistors and L293D H-bridges to the BeagleBone. After initializing the program through Python, all we had to do was press the push button, giving a “proof of concept” demonstrating the motors working to operate the robot arm.

The Towers of Hanoi algorithm we implemented used a recursive function accepting the labels of the three mounting rods, the number of disks in the starting stack, and a move order list to save the order of execution for the moves as arguments. The first recursive call in the algorithm moves any disks from on top of the lower ones to the non-target rod to make way for the base disk, the second recursive call moves the base disk to the target rod, and the final recursive call retrieves the disks that were formerly on top of it to the target rod on top of the base disk.

1. References:
   1. <https://cdn-learn.adafruit.com/downloads/pdf/setting-up-io-python-library-on-beaglebone-black.pdf>
   2. <http://wiki.seeedstudio.com/Motor_Bridge_Cape_v1.0/>
   3. <http://www.ti.com/lit/ds/symlink/l293d.pdf>
   4. <http://trivox.tripod.com/lego-nxt-motor-input-output.html>
   5. <https://learn.adafruit.com/connecting-a-push-button-to-beaglebone-black/overview>
   6. <https://learn.adafruit.com/setting-up-io-python-library-on-beaglebone-black/gpio>