Ready to Serve Trash Bin

Electrical & Computer Engineering

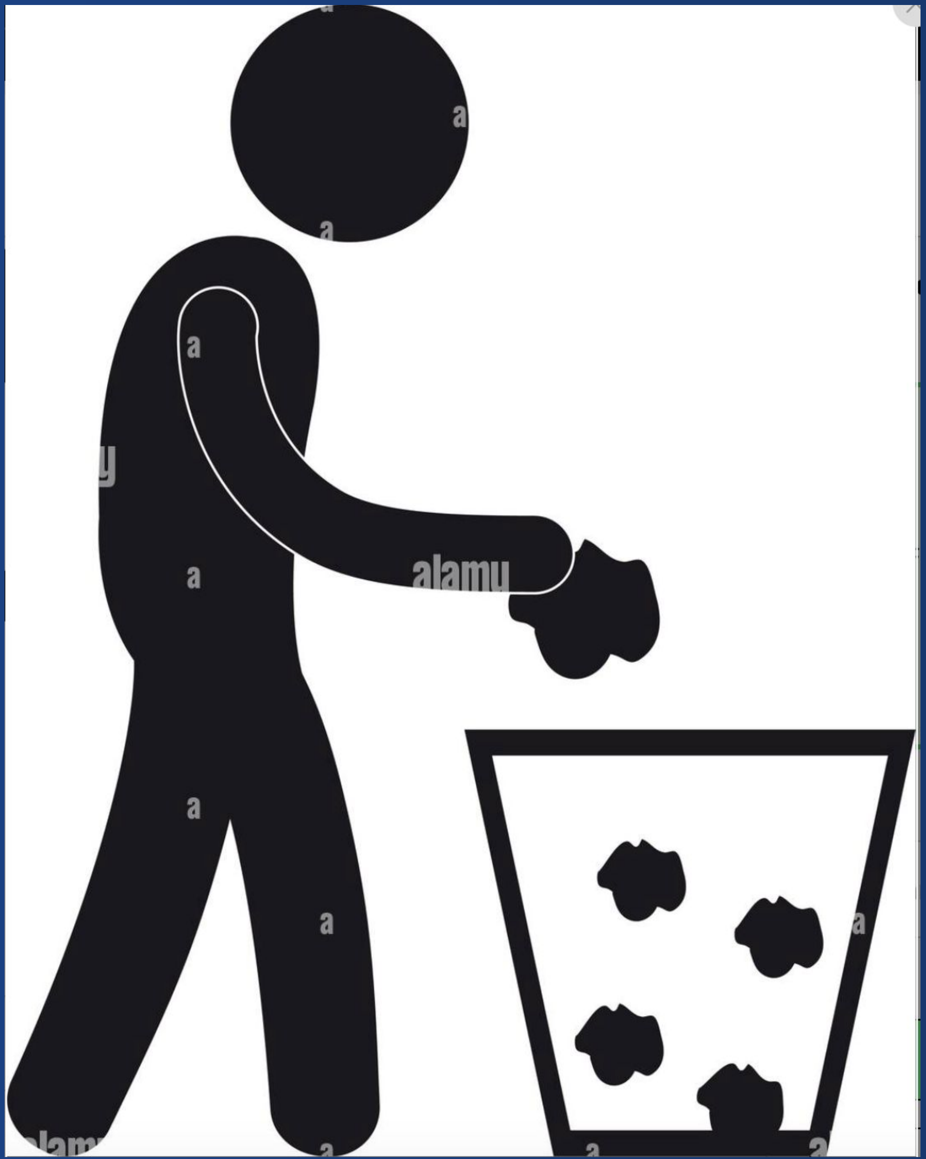
Team 19

Dongming Liu, Josh Litao, Owen Xu

04/29/2024

Agenda

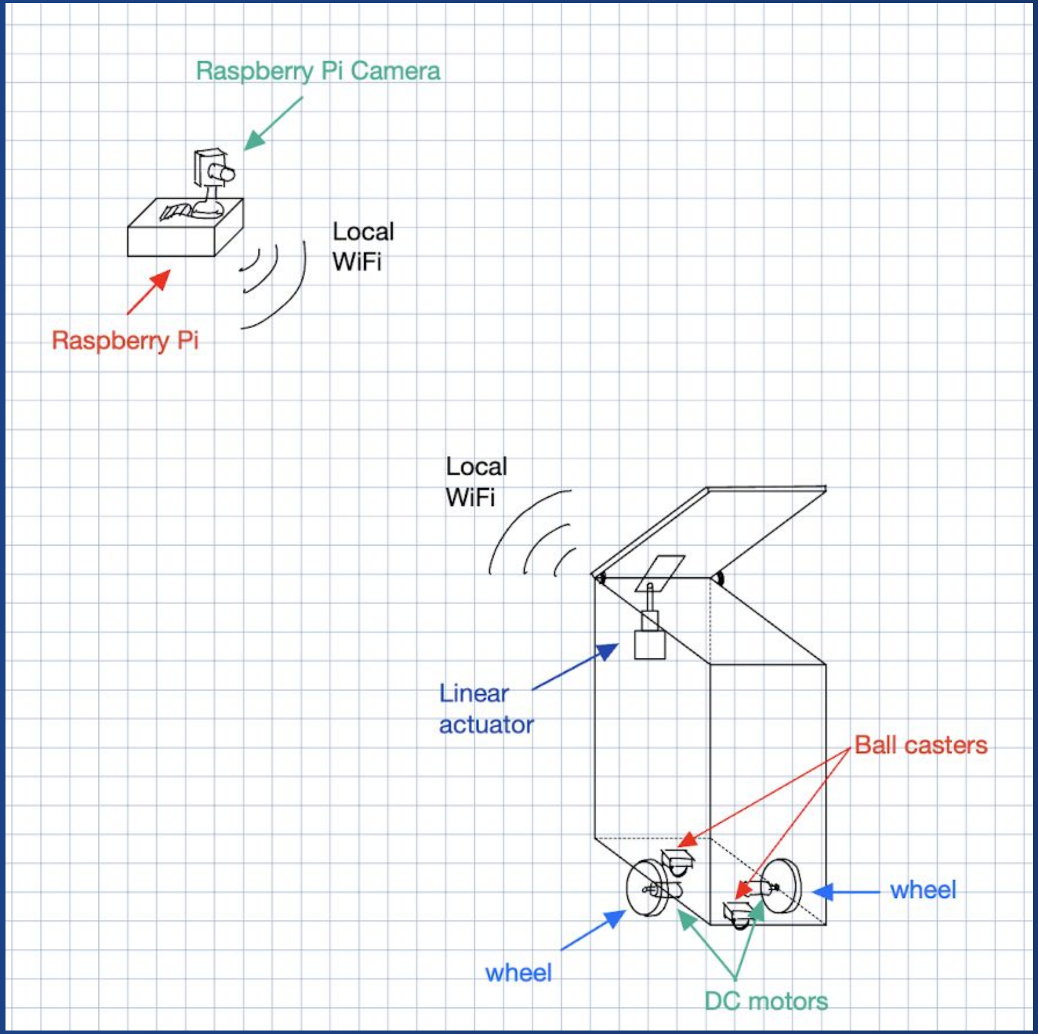
- **Introduction**
- **Objective**
- **Design**
- **Challenges**
- **Conclusion**
- **Future work**



https://www.alamy.com/stock-photo-man-throwing-trash-can-pictogram-133054853.html

The Problem

Disposing of trash is a routine task for many but poses significant challenges for those with mobility impairments, affecting around 12.1% of U.S. adults according to the CDC. These individuals often struggle to manage trash disposal independently, relying on assistance or allowing trash to accumulate, leading to sanitation concerns.

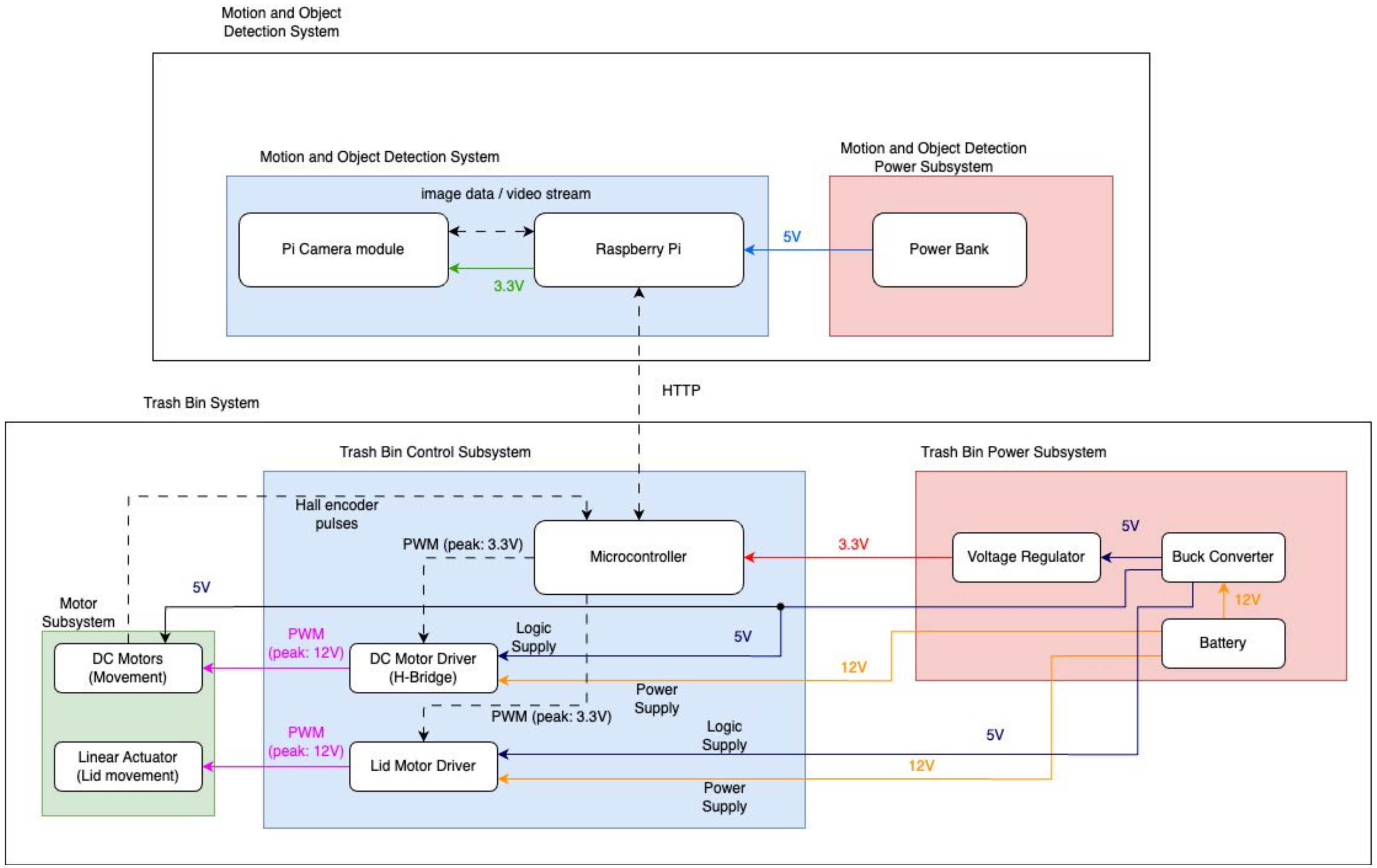


The Solution

To address these challenges, we propose a revolutionary trash bin equipped with Computer Vision technology. This bin responds to specific hand gestures, initiating its movement towards the user. A strategically positioned camera detects the user's intention to dispose of trash. Additionally, the lid is automated, opening upon arrival to facilitate easy trash disposal. After use, the lid closes, and the bin autonomously returns to its original position, streamlining the disposal process.

Design

Block Diagram



Ready-to-serve Trash Bin Block Diagram

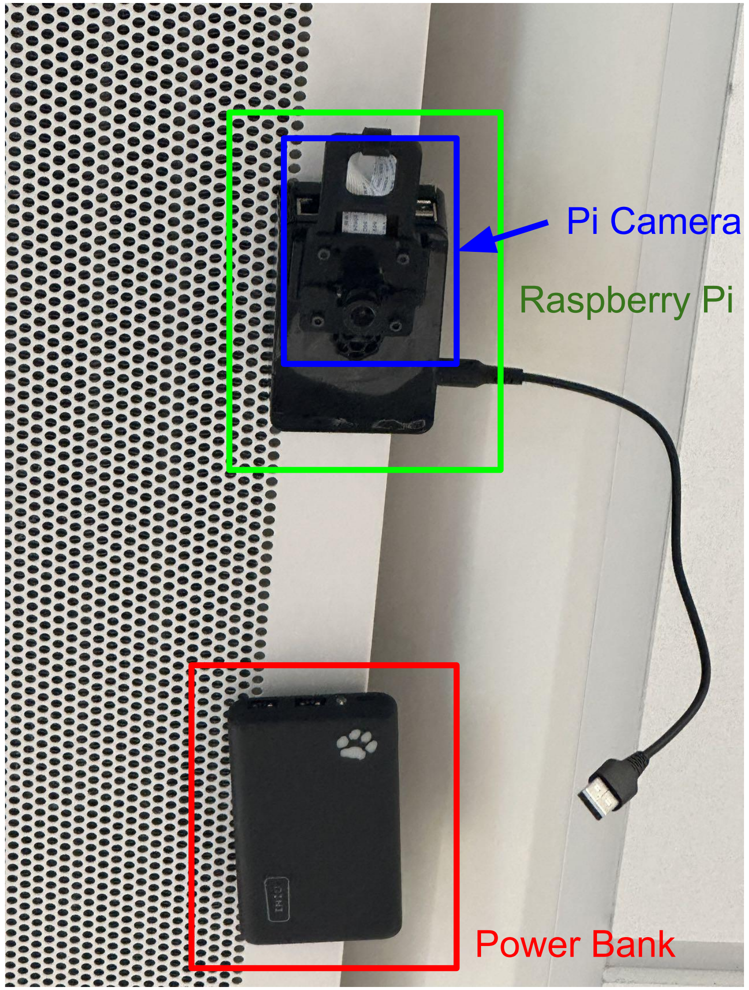
Color code
Red --- Controller Power (3.3V)
Orange --- Motor Driver Power (12 V)
Blue --- Pi Power (5 V)
Green --- Camera Power (3.3 V)
Pink--- Motor Power (6V ~ 12V)

Line Coding
Solid --- Power Line
Dash --- Data Line

System

Motion and Object Detection System

Motion and Object Detection System - Physical View



Physical View of the Motion and Object Detection System

Power Subsystem

Requirement	Verification
• The power bank should continuously provide at least 5V for at least 3 hours.	• Connect the USB-C output to a USB-C power tester and connect the USB-C tester to the Raspberry Pi. Then, verify the voltage reading and power reading. • Then, leave the power bank for at least 3 hour and verify the reading afterward.
• The power adapter should continuously supply at least 15W.	• Connect the USB-C output to a USB-C power tester and connect the USB-C tester to the Raspberry Pi. Then, verify the voltage reading and power reading.

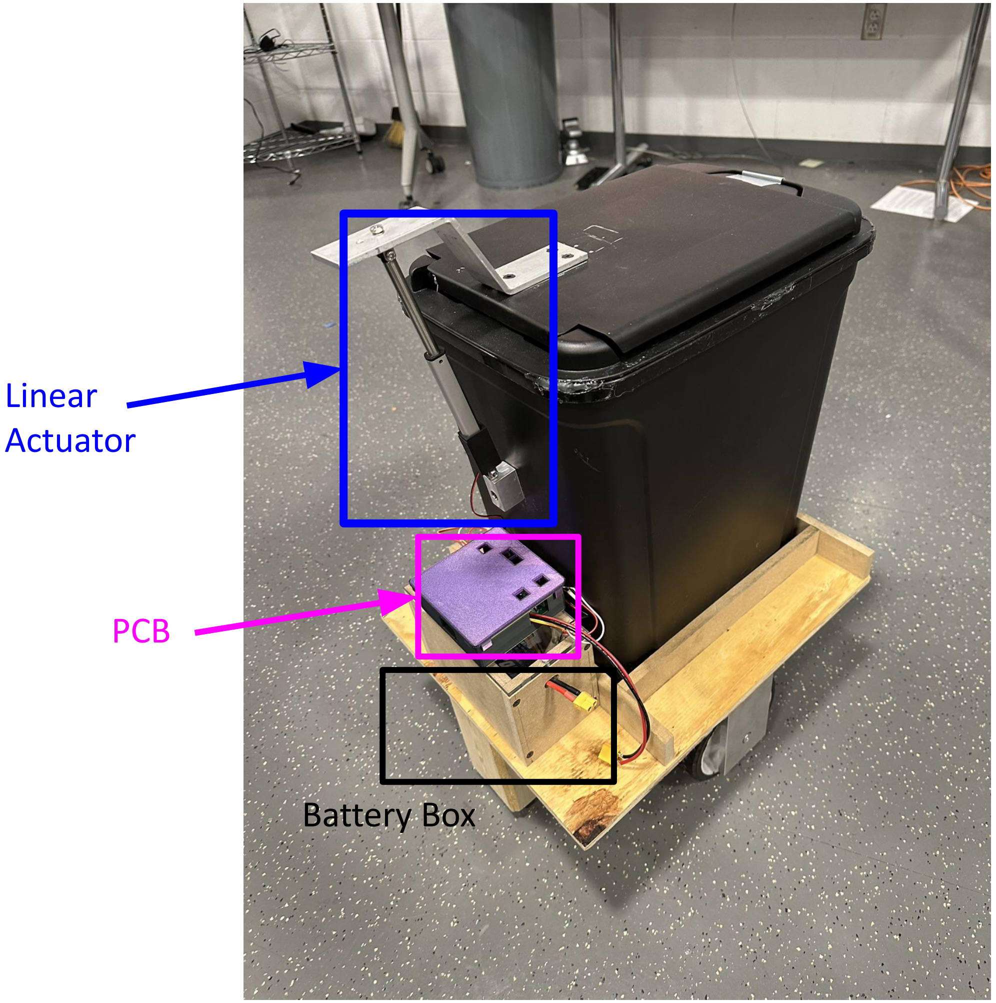
Control Subsystem

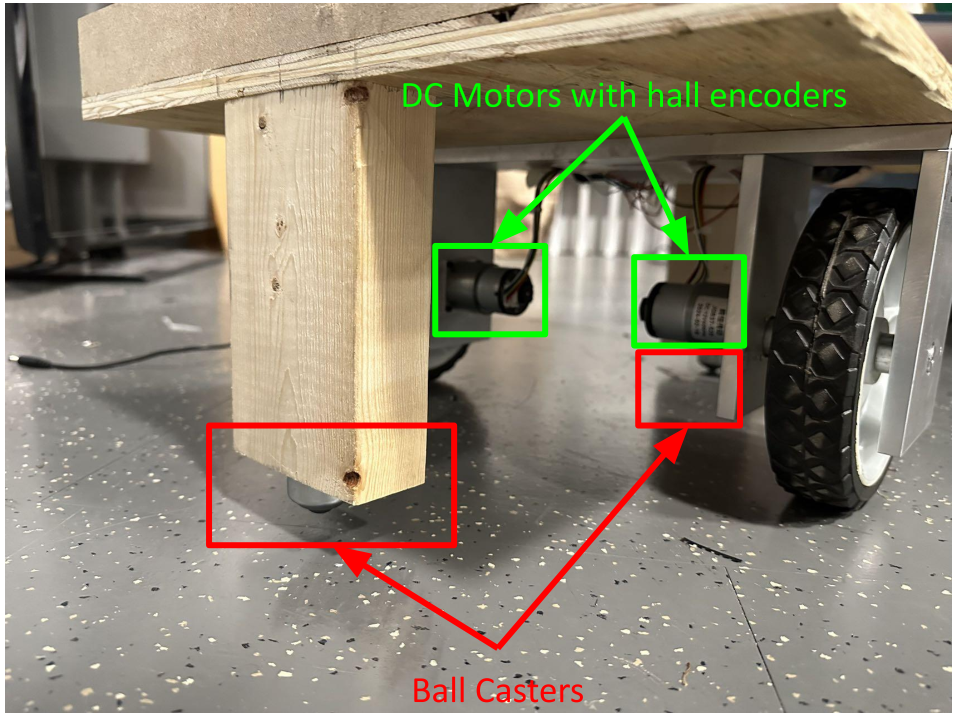
Requirement	Verification
• The Raspberry Pi needs to continuously power the Raspberry Pi camera module with a voltage of 3.3V ± 0.2V.	• Power on the Raspberry Pi. Look for the Pi camera connector (a rectangular 15-pin connector near the 3.5mm audio jack with the label “camera”). • Turn on a multimeter. Put the negative probe on PIN 1 of the connector and the positive probe on PIN 15. Then, check the voltage readings from the multimeter and see whether it is within the range of 3.3V ± 0.2V. Note: PIN 1 is the left furthest pin from the label “camera” and PIN 15 is the right furthest pin from the label “camera”.
• The RTT (Round Trip Time) of the communication should be no more than 700ms.	• Power on the MCU and flash the code to the MCU. The code will output the IP address of the MCU on the serial monitor on the Arduino IDE. • Power on the Raspberry Pi and open the terminal on Pi. Type **ping ip_address_of_mcu -c <n>** in the terminal, where n is specified as the number of packets. By default, **n = 20**. The **time** section of the ping output is the RTT of the communication between Pi and MCU. Compare the output and see whether it is within the limitation.
• The CV system should send commands to the trash bin only when the V sign is detected. The CV system should also be able to localize the position of the hand.	• Test with multiple hand gestures at different positions. The CV unit should correctly distinguish the V sign. Trash bin should only move when showing a V sign to the camera. The stop position of the bin should be in a 25 cm ± 5 cm range from the hand.

System

Trash Bin System

Trash Bin System - Physical View





Physical View of the Trash Bin Subsystem

Power Subsystem

Requirement	Verification
• The power system needs to provide a voltage with a range from 6V to 12V to the DC motor and a voltage with a range from 6V to 12V to the linear actuator.	• Using a digital multimeter, measure the output voltage of the battery.
• The buck converter of the power subsystem should supply a voltage of 5 V ± 0.5 V.	• Using a digital multimeter, measure the output voltage of the converter.

Motor Subsystem

Requirement	Verification
• The DC motor can work in the range of 6V and 12V.	• Connect the positive and the negative ends of the motor to the voltage source and multimeter. • Set the voltage source to 6V, check the output from the multimeter, and see if the motor is turning • Repeat this process for 7 to 12V voltage source
• The linear motor can work in the range of 6V to 12V.	• Connect the positive and the negative ends of the motor to the voltage source and multimeter. • Set the voltage source to 6V, check the output from the multimeter, and see if the motor is turning • Repeat this process for 7 to 12V voltage source
• The hall encoders on the DC motors can output square wave.	• Connect encoder phase A output to channel 1 of an oscilloscope and encoder phase B to channel 2. • Supply 5V to the hall encoder logic • Spin the magnetic wheels at the end of the DC motors and check the waveform on the oscilloscope.

Control Subsystem - Part 1

Requirement	Verification
• The MCU needs to ensure a persistent WiFi connection.	• Connect the USB of the MCU board to a laptop. • Write and upload a program that continuously verifies the WiFi connection every 1 to 2 seconds using Arduino IDE. The program writes messages of successful connection and connection failure to the Arduino serial monitor. • Use a COM tool that records the output of the serial port that connects to the MCU and outputs a log file once execution finishes. • Run the MCU program simultaneously for at least 30 minutes. Once the MCU program finishes running, check the log and see if the log contains any message regarding connection failure. • To test the stationary and moving condition, we will verify to see if the connection is stable under the following scenarios: ○ Some obstacles or people are around the bin.(stationary) ○ The bin is moving towards or away from the router. (moving)

Control Subsystem - Part 2

Requirement	Verification
● The MCU needs to receive at least 90% of the requests (commands) coming from the Motion and Object Detection System.	● Connect the USB of the MCU board to a laptop. ● Write a Python script that periodically sends different requests (commands) to the MCU. By default the testing period would be every 2 ± seconds. Put a serial print statement in the command handler code, and upload the code to the MCU ● Use a COM tool that records the output of the serial port that connects to the MCU and outputs a log file once execution finishes. ● Let the MCU and Python script run for at least 30 minutes. Once the Python script finishes running, check the log and see if total commands received is at least 90% of total commands sent.
● The MCU needs to distinguish different commands, set the correct duty cycles of the PWM signals, and output the PWM signals to the motor drivers.	● Connect the PWM output pins of the MCU to an oscilloscope and the USB port of the MCU to the laptop. ● Writing a testing function in Arduino IDE. The function takes in inputs from the serial port and change commands (to simplified the testing, each command has an char ID) ● Check the waveform on the oscilloscope and see if the PWM cycle is adjust accordingly

Control Subsystem - Part 3

Requirement	Verification
• L298 chip can switch the turning direction of DC motors and the linear actuator.	• Supply 12 V to the VS (supply voltage) port and 5 V to the VSS (logic supply voltage) port. • Ground all the CURRENT SENSING ports. Connect the OUTPUT1 and OUTPUT2 pins to the positive and negative ends of a DC motor / linear actuator. Connect ENABLE A to 3.3V. • Power IN1 with a logic high (3.3V) and logic low (eg. GND). Observe the turning direction of the motor / linear actuator for 5 seconds. The motor should turn in one direction (clockwise / counterclockwise). The linear actuator should either push or pull back. • Then, flip the inputs to IN1 and IN2. Observe the turning direction of the motor / linear actuator for 5 seconds. The motor / linear actuator should turn in the opposite direction. • Repeat the process for OUTPUT3, OUTPUT4, IN3, IN4, ENABLE B. Note: ENABLE B, IN 3 (h-bridge control 1), IN 4 (h-bridge control 2), OUTPUT 3 (motor positive end), and OUTPUT 4 (motor negative end)

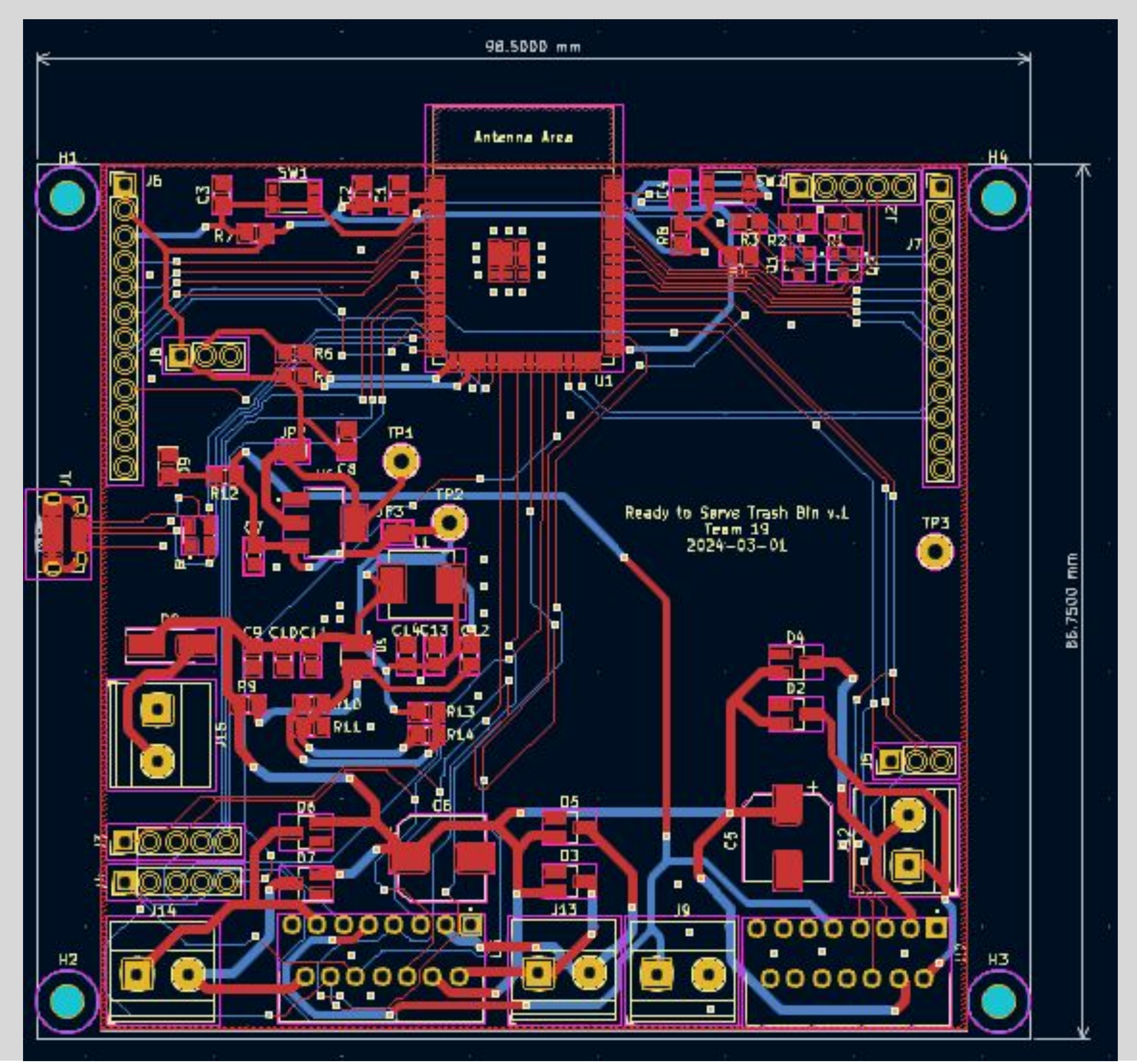
Hardware Design

PCB

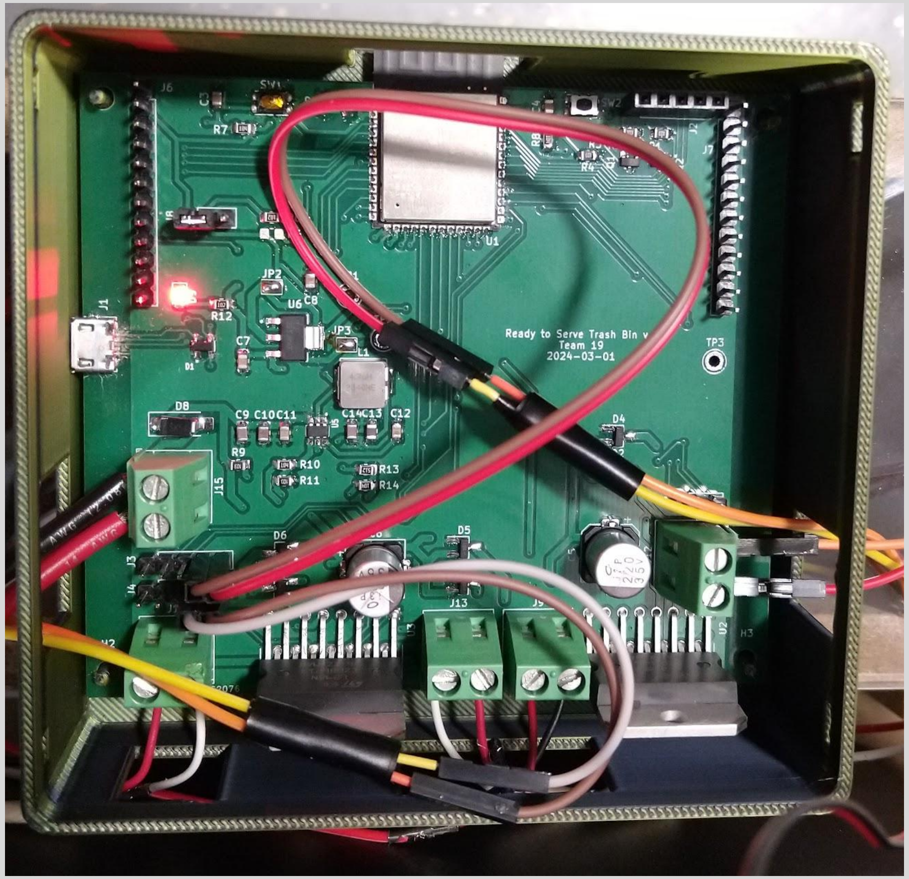
PCB Layout

PCB Purpose

- Deliver power from battery to microcontroller and motors.
    - Buck converter from ~12 V to 5 V.
    - LDO linear regulator from 5 V to 3.3 V.
- Control motors (two DC motors and one linear actuator) through L298N motor drivers.
- Provide connections between motor encoders and microcontroller.



PCB Assembled with Connections

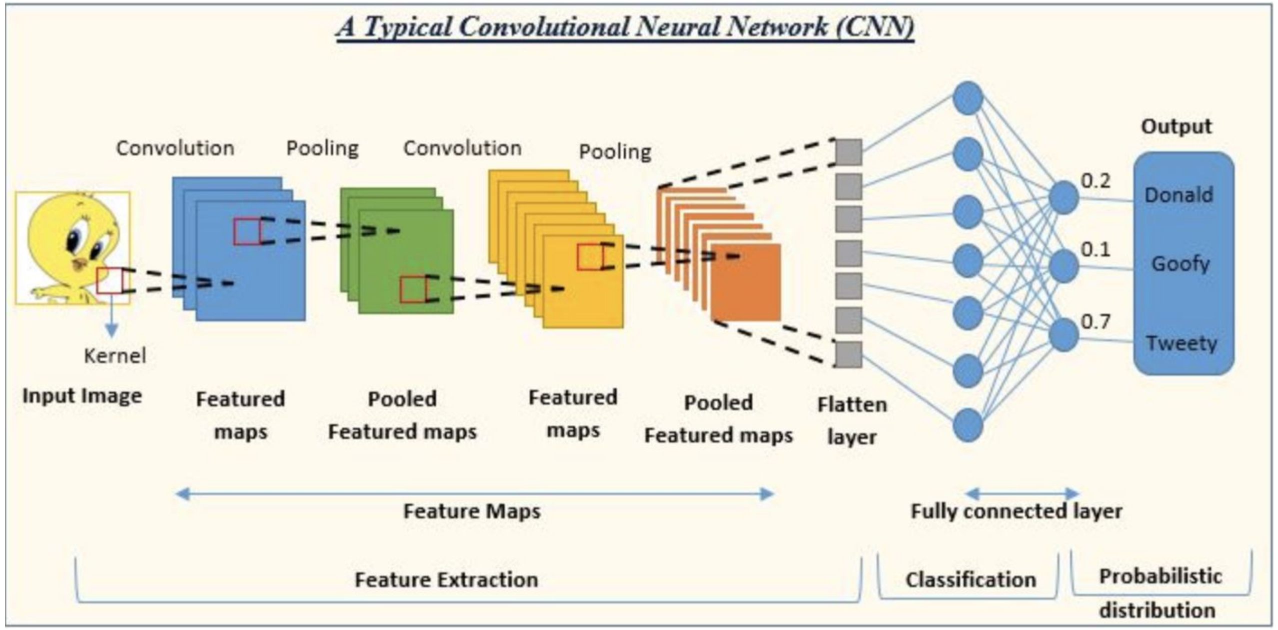


Software Design

Object Detection and Path Planning

Object Detection

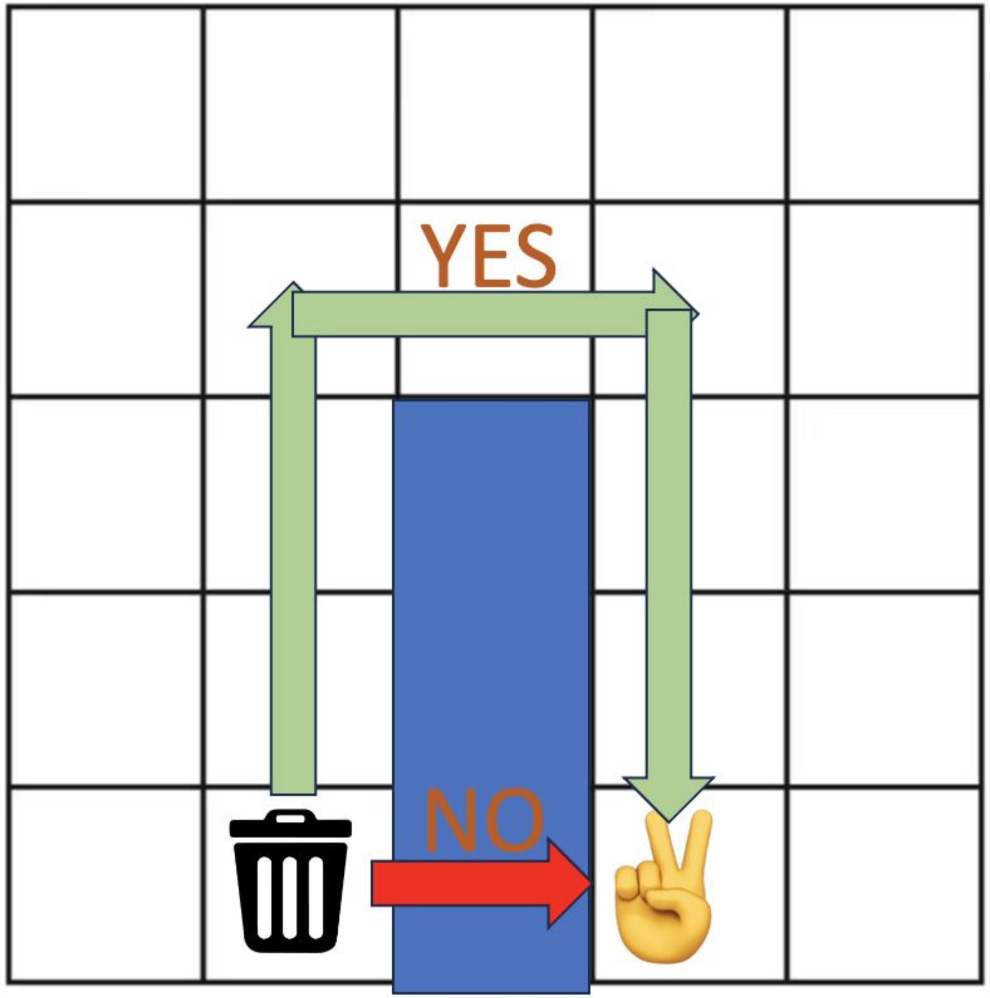
- Detecting the position of the hand in the image and classify the hand gesture is hard to do for one model.
- Break the job into 2 models.
- First model will find the position of the hand from the big image
- Second model will classify the hand gesture to see if the person is calling the trash bin.
- Both models are trained with Convolutional Neural Network(CNN).



https://www.analyticsvidhya.com/blog/2022/01/convolutional-neural-network-an-overview/

Path Planning

- Discretize the room into several squares.
- Build the graph representing the room where each square is a vertex.
- Add an undirected edge between two vertices if they are adjacent to each other and neither of them is an obstacle or a wall.
- Run a simple BFS algorithm from the start point to the end point.



Software Design

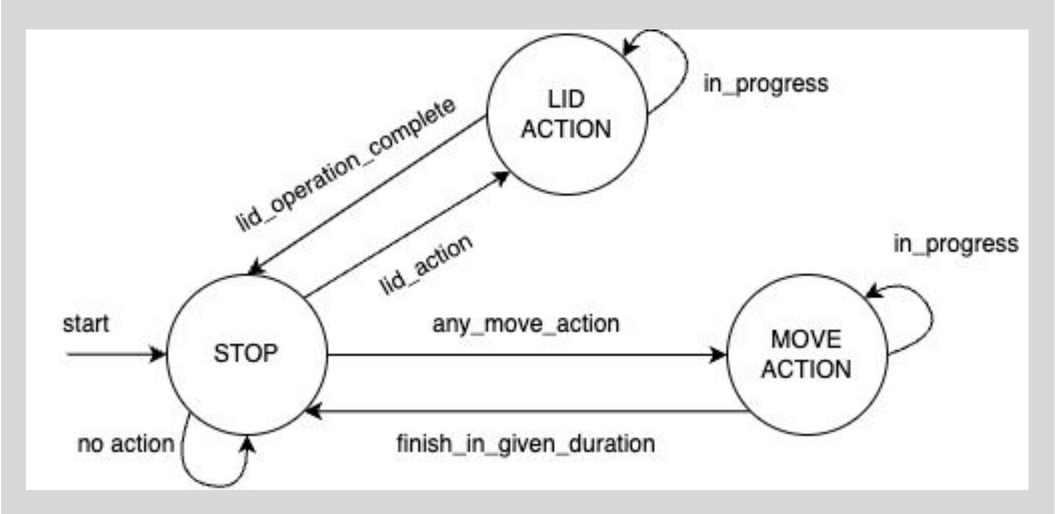
MCU Control and Wireless Communication

Main State Machine

States:

- STOP: no motion
- LID ACTION: lid operation in progress
- MOVE ACTION: bin movement control in progress

The state transitions for moving from STOP to other states is managed by the wireless communication APIs.

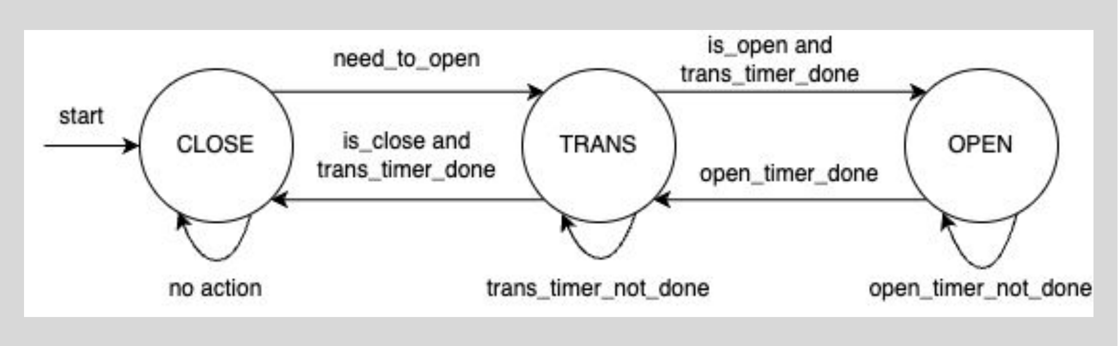


Lid Operation State Machine

States:

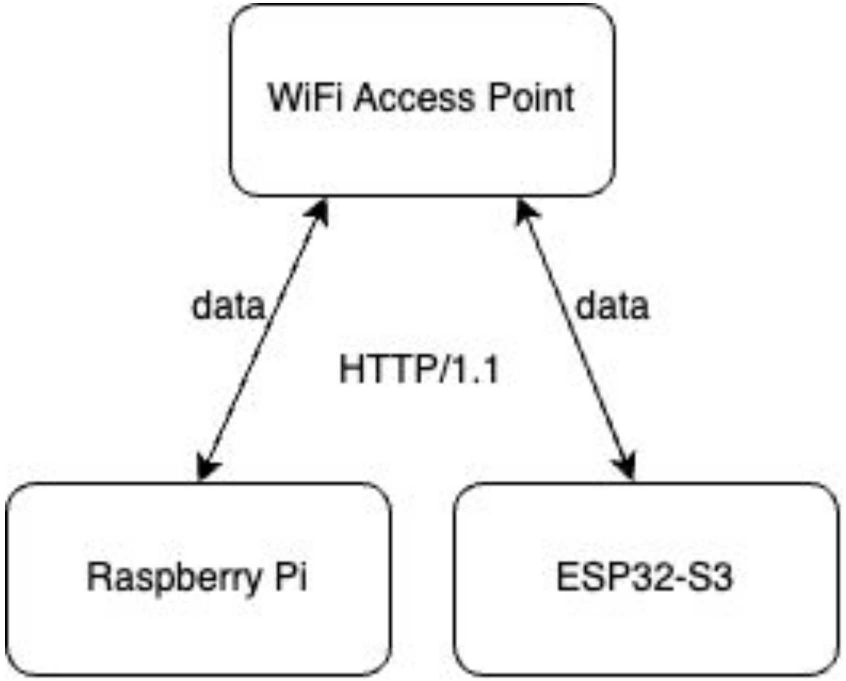
- CLOSE: lid is closed
- TRANS: lid is opening / closing
- OPEN: lid is opened

The wireless communication APIs can only trigger the opening of the lid. The rest of the operations (open → stay → close) are managed by a timer.



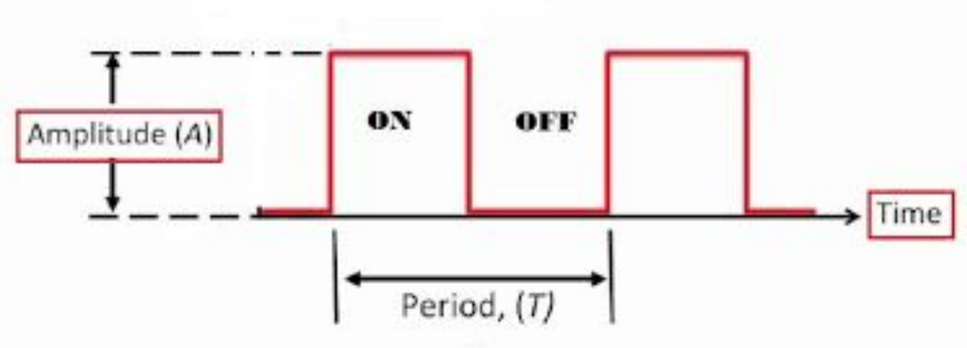
Wireless Communication

- Allow Raspberry Pi to control the trash bin far away, creating more flexibility in placing the Pi and the trash bin
- Use WiFi (station mode) to gain access to WiFi access points in the local area network, enabling a larger range of WiFi communication
- Create an asynchronous web server on the microcontroller to establish different communication API
- Use HTTP/1.1 protocol for data transmission, allowing both microcontroller and Raspberry Pi to parse the data easily

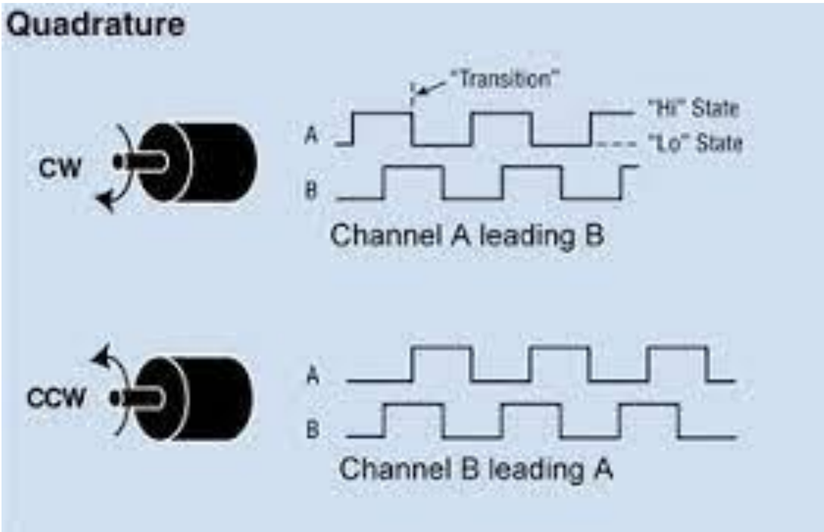


Motor Control

- Use PWM to control the speed of the DC motors and linear actuator
    - Microcontroller generates PWM
    - The motor driver (L298) amplifies the PWM to the motors
- Count pulses coming from the hall encoders on the motors with hardware interrupts in order to estimate the current speed of the DC motors and determine the turning direction of the DC motors
- Use a PID controller to maintain the speed of the DC motors and make speed changes smoother



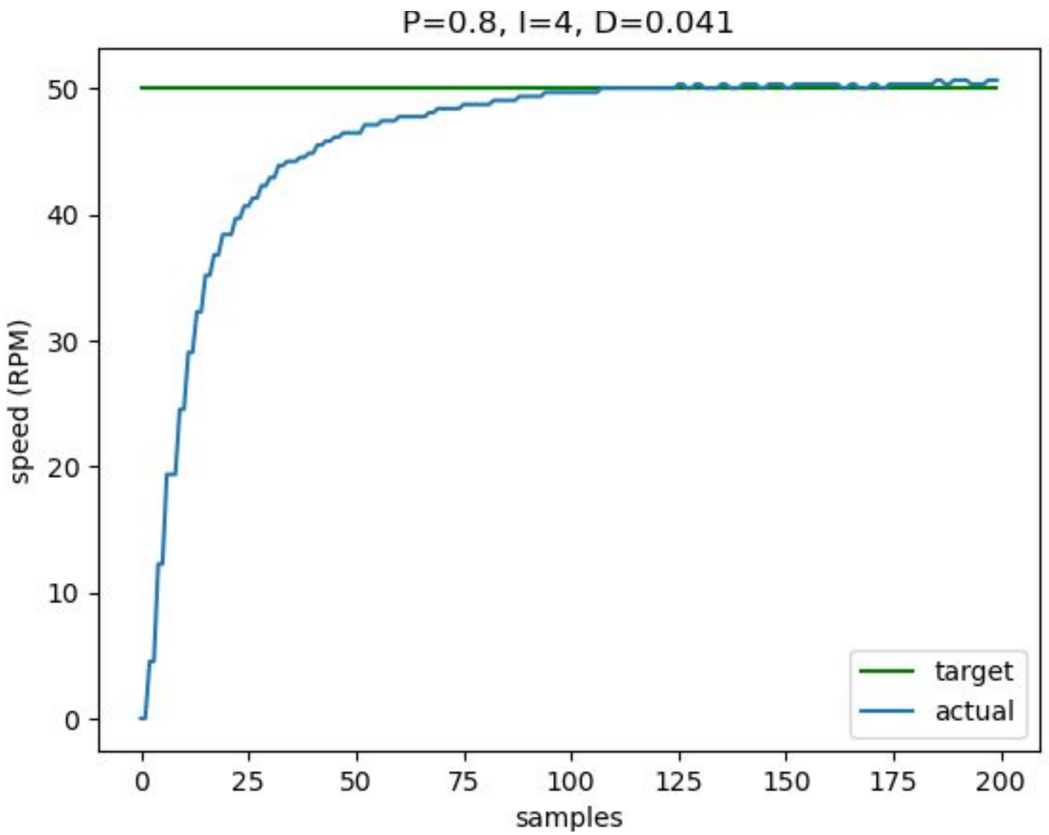
https://www.geeksforgeeks.org/duty-cycle/

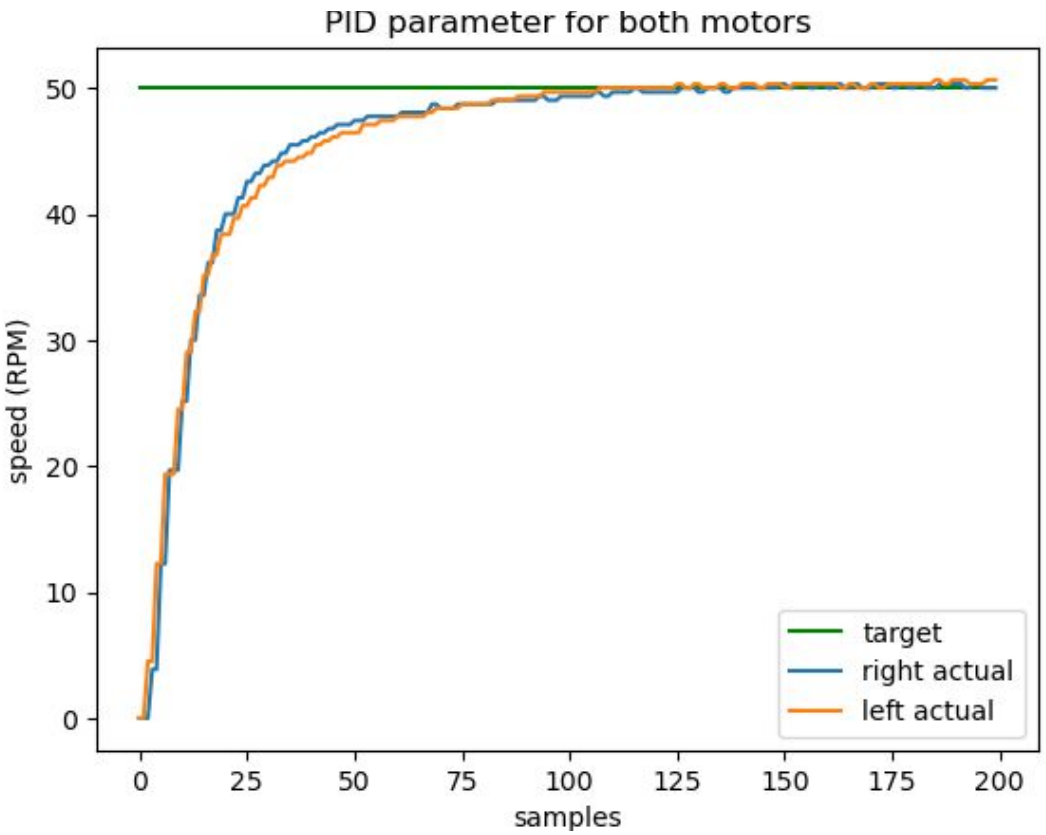


https://www.allaboutcircuits.com/industry-articles/measure-position-speed-control-of-a-dc-motor-using-an-analog-pid-controller/

PID Tuning and Results

For our motor, the best sample rate of the PID is 10Hz (0.1s)





Demo

Demo

Challenges

Challenges

- **Correctly train the CV model so that it classify hand gestures correctly**
- **Correctly localize the position of the hand**
- **Finding the best sample time and PID parameters**
- **Troubleshooting the PCB with controlling the motors**

Future Work

Future Work

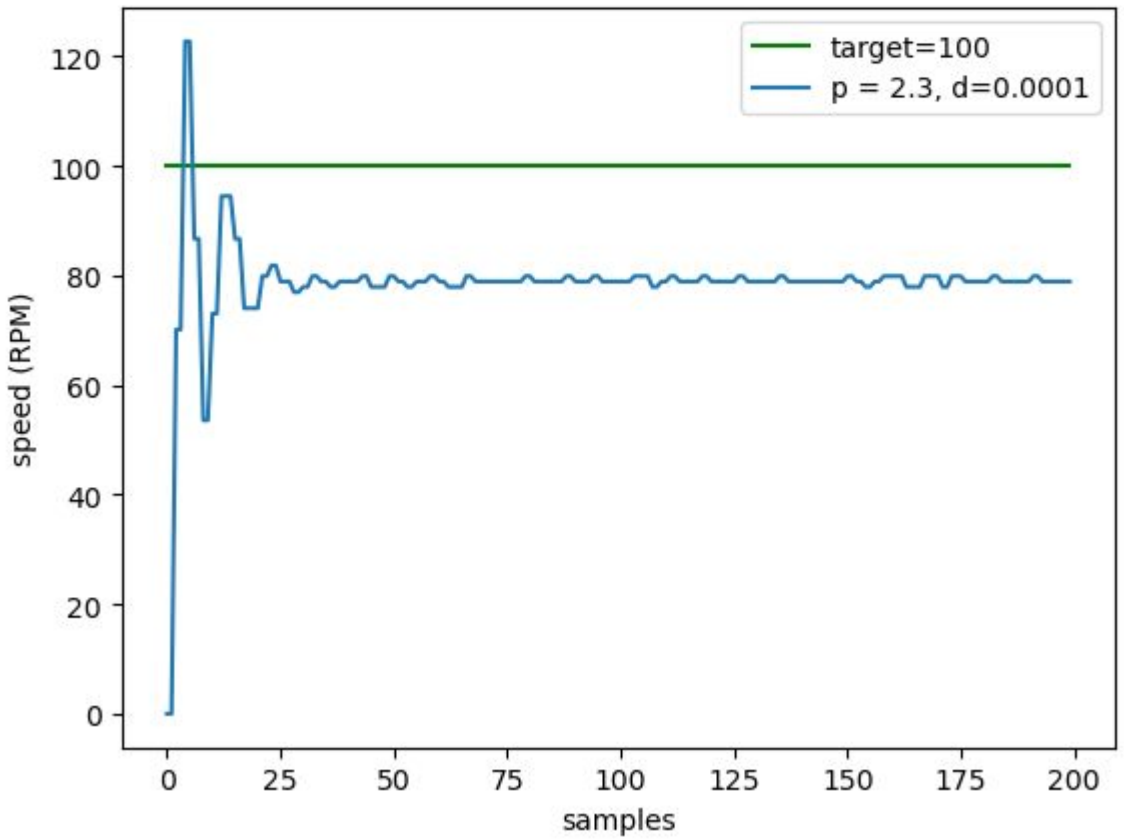
- Create a customized protocol using UDP to speed up wireless communication
- Include a IMU in our design for maintaining the heading angle of the trash bin and improving the PID control
- Include a HD camera to better capture the image of the room and estimate the location of people in the room.
- The CV models will be able to distinguish multiple people’s hand thus being able to serve multiple people.
- Users can take a picture of their room and the CV model can generate the map automatically.

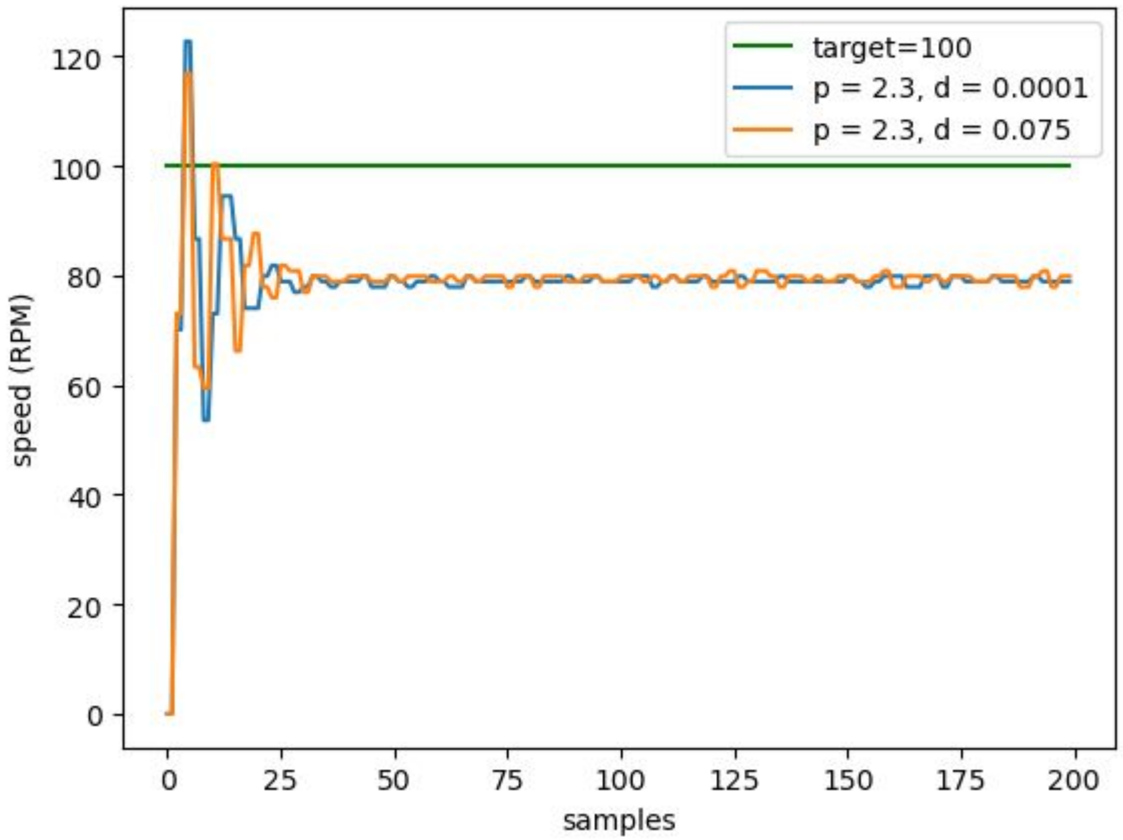
Thank You

The Grainger College of Engineering

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

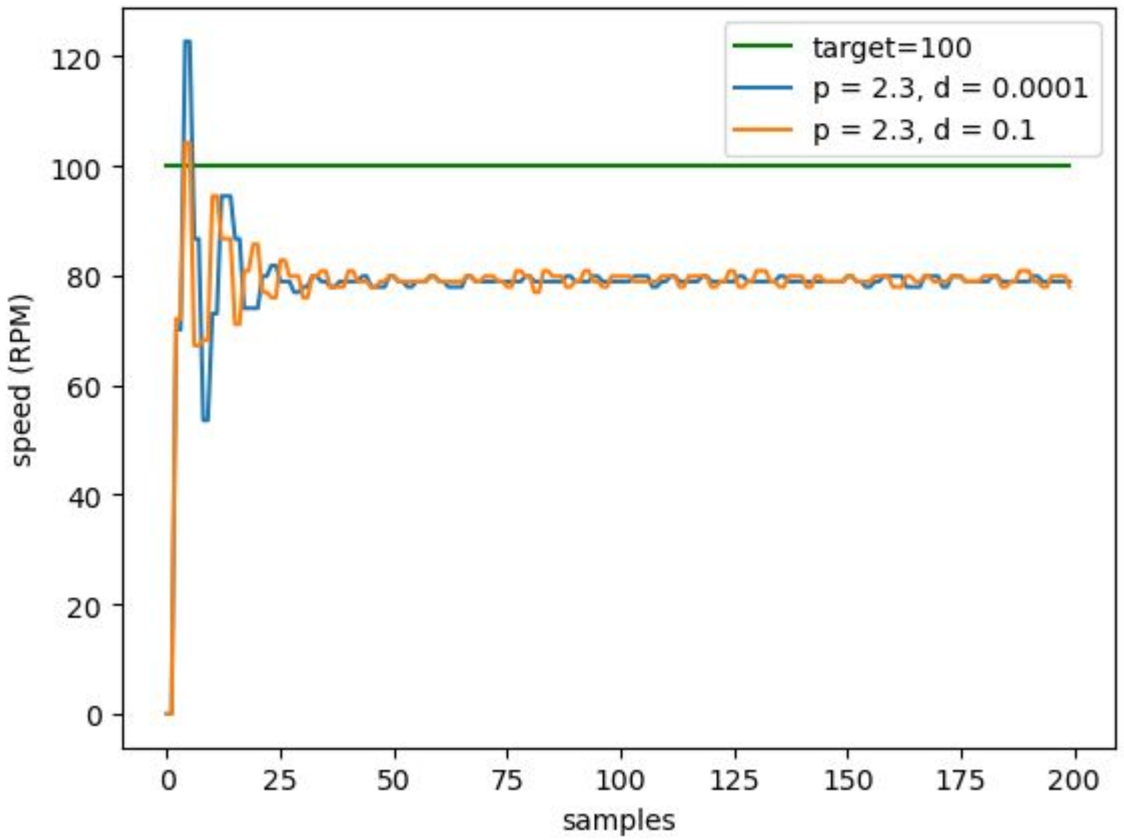
PID Tuning Data

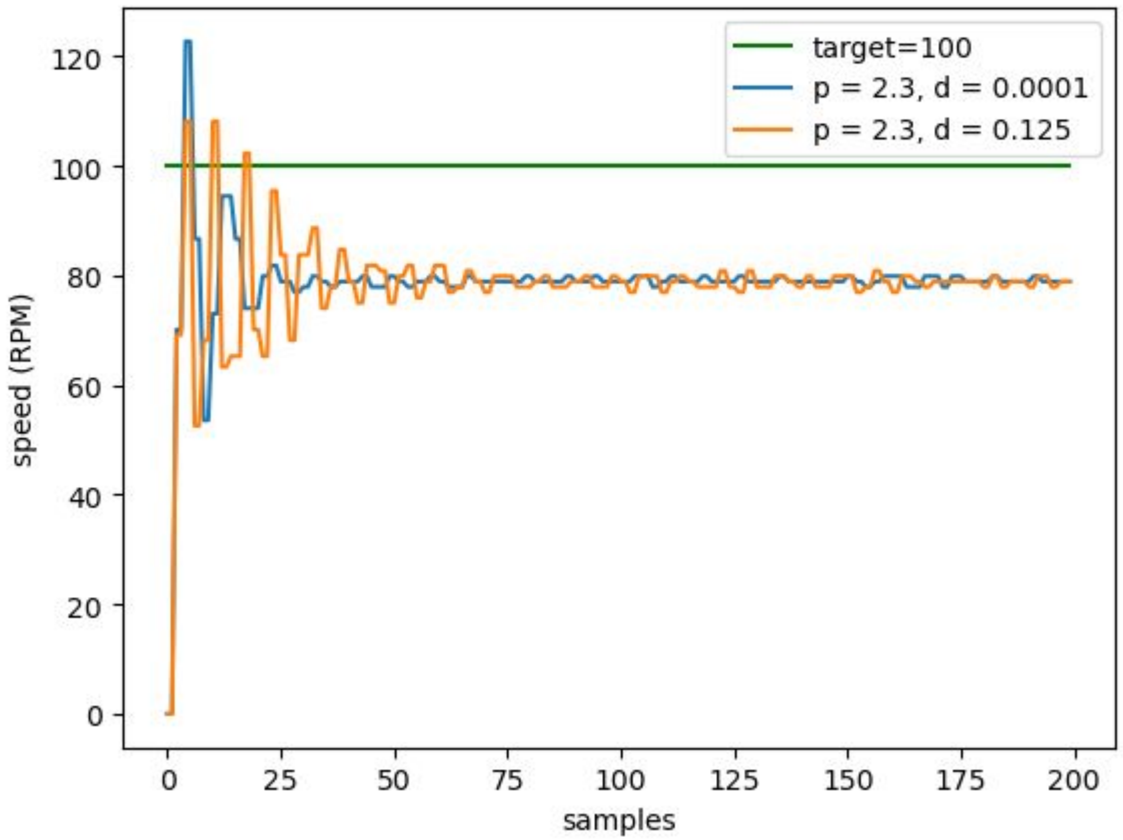




Back

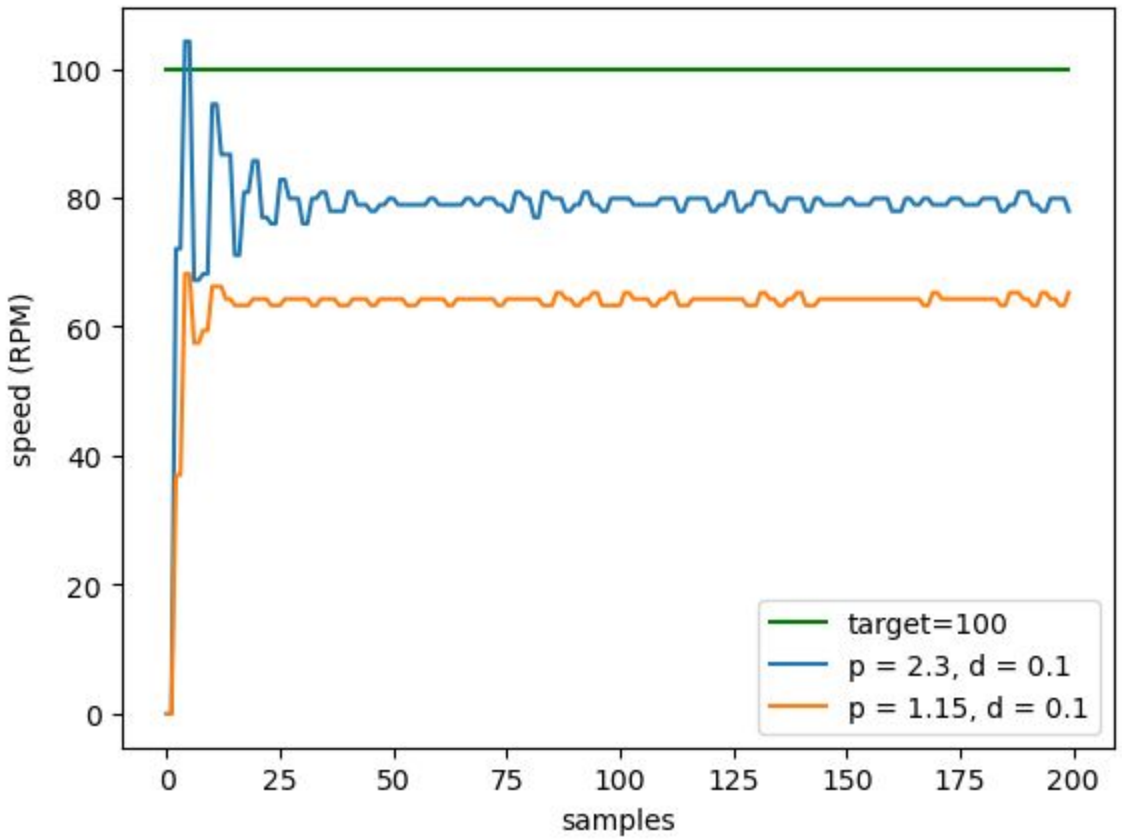
PID Tuning Data

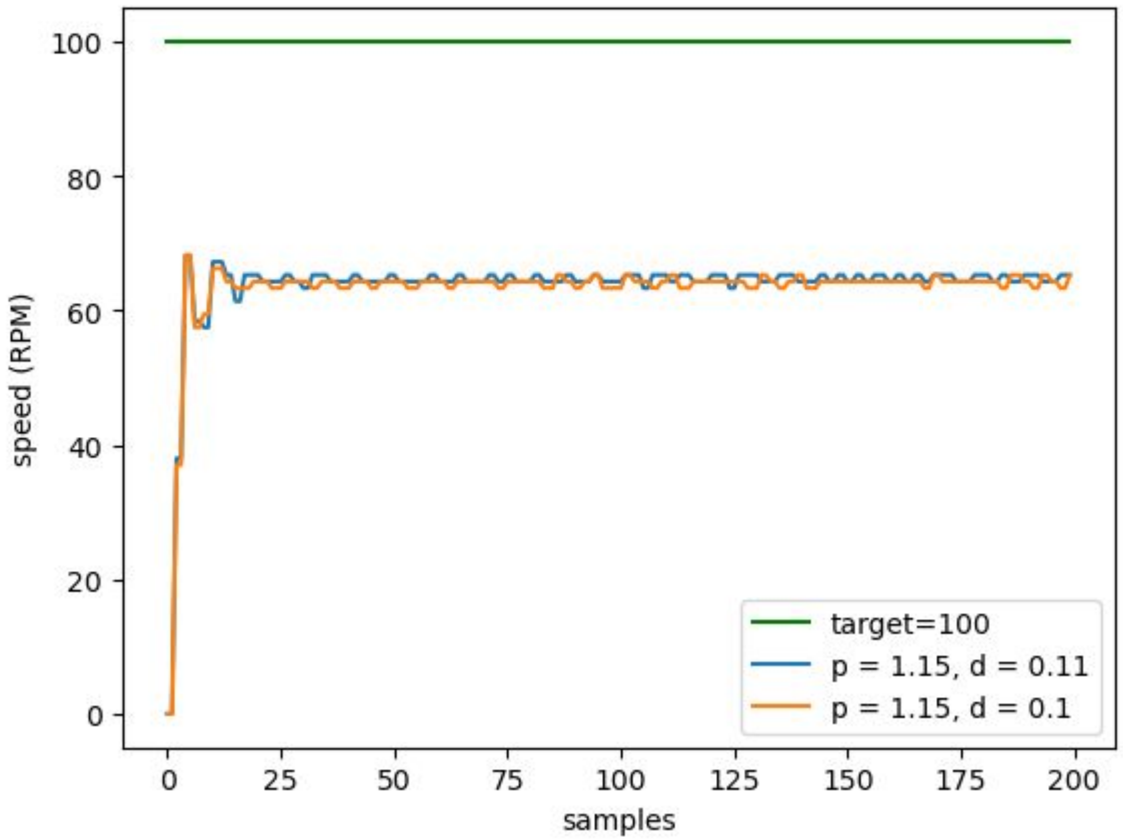




Back

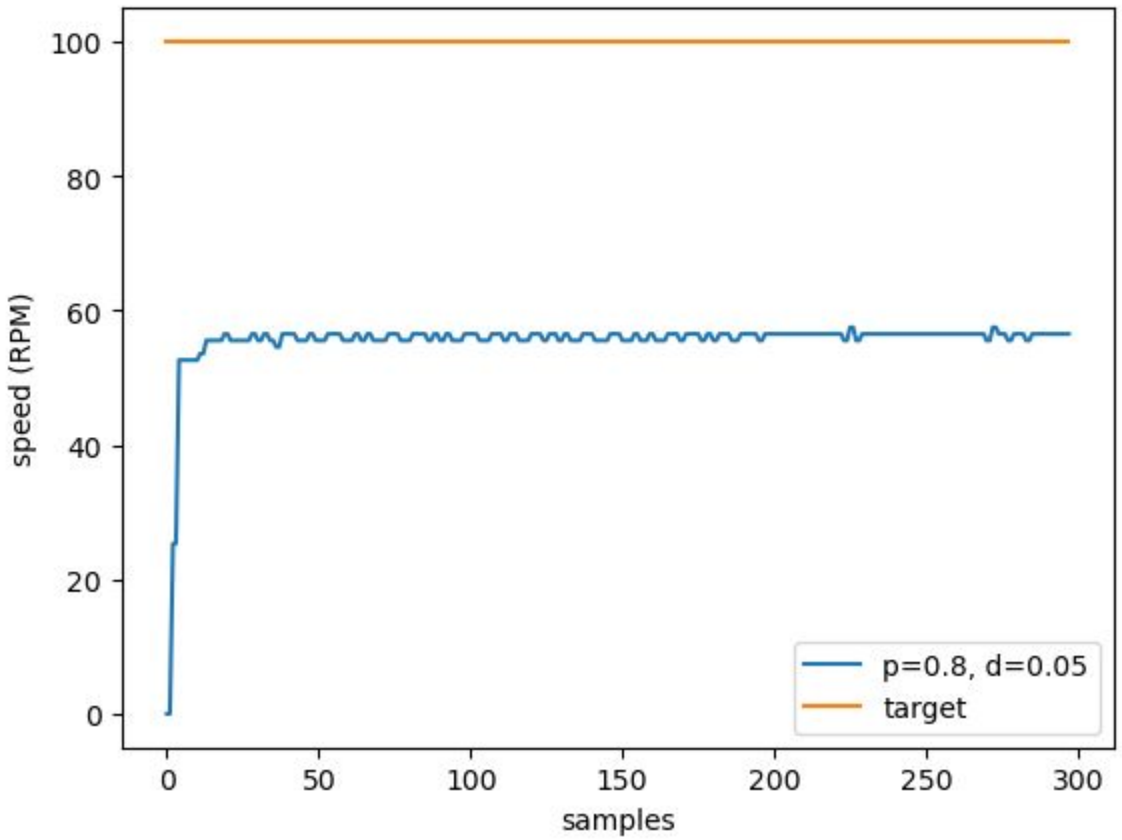
PID Tuning Data

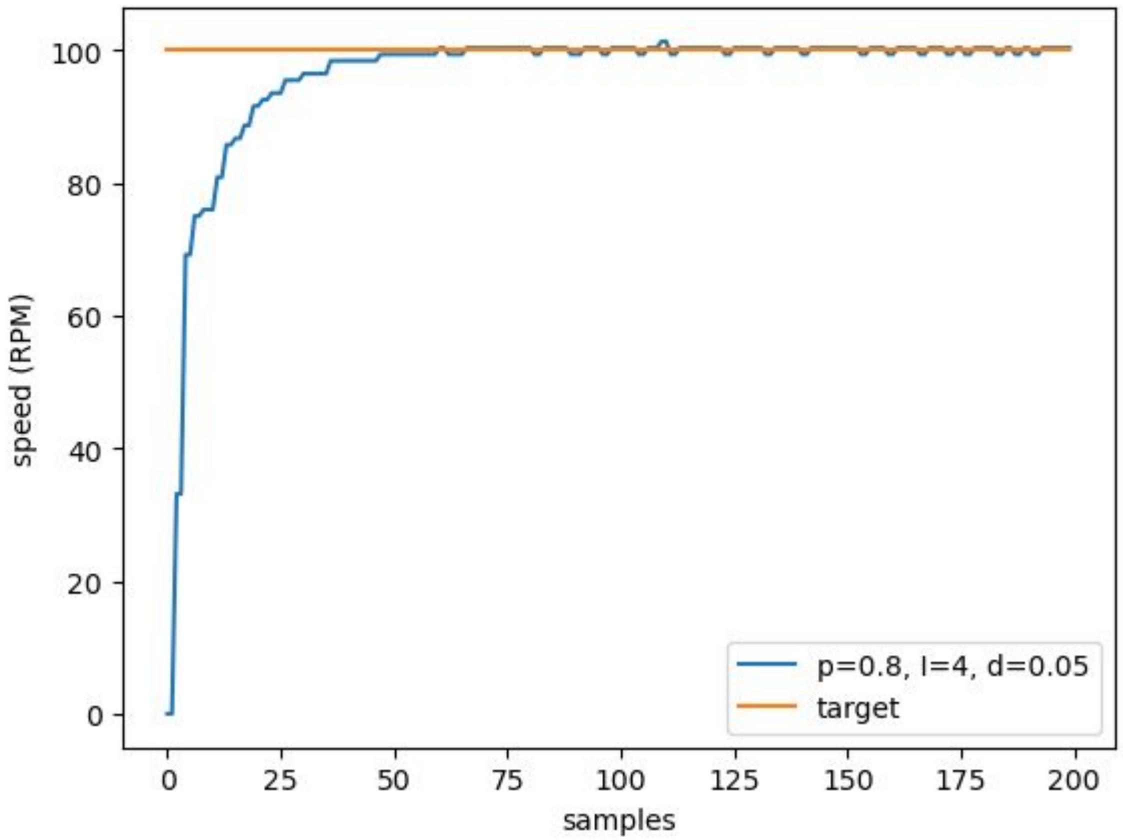




Back

PID Tuning Data





Back