

## Our team

# TEAM ISAAC



## Roles:

Trent: Scrum Master (facilitate the planning, daily (5 min))

Micah: Engineer

Rohim: Architect/Developer

Kayden: Documentation

## Code:

<https://github.com/Cruzbuddy41/Isaac>

# Sprint 1

# Sprint 1

1. Planning
    - a. Documentation
    - b. Refinement
    - c. Acceptance criteria/ conditions of satisfaction
  2. Daily
    - a. 5 min
  3. Demonstration
    - a. Demo
  4. Retrospective
- 

## What to do in a daily

1. What we did yesterday
  2. What are we going to do today
  3. Do we have and if so what are the blockers
- 

## Robot parts

1. 18 volt rechatchable battery
  2. raspberry pi 5
  3. 720 p usb camera
  4. 18 volt to 12 volt step down transformer
  5. Stepper motor hat
  6. Casey parts
  7. battery holder/flashlight
  8. Multi meters
  9. an array of pliers
- 

## Building Plan

1. Develop chassis
2. Add Wheels
3. Attach Battery and Connect to Transformer
4. Attach and Connect Stepper Motor to Transformer

5. Attach and Connect Stepper Motor to Motor
6. Attach and Connect Stepper Motor to Raspberry Pi
7. Install PyCharm and Raspberry OS

---

#### Programming Plan:

1. Install Raspberry OS
2. Connect Robot to computer with raspberry os
3. Install Libraries BCM2835, WiringPi, and Python on Raspberry PI
4. Import DRV8825 module
5. Download sample code with functions for motor use
6. Configure and Connect Raspberry PI ports 16,17,20,21,22,27.
7. Test run sample code.
8. Record any successful run, measuring distance and elapsed time to calculate speed.
9. Calculate the correct time for the robot to move 2 feet.
10. Adjust code accordingly and run the program.

```
import RPi.GPIO as GPIO
import time
from DRV8825 import DRV8825

try:
    Motor1 = DRV8825(dir_pin=13, step_pin=19, enable_pin=12, mode_pins=(16, 17,
20))
    Motor2 = DRV8825(dir_pin=24, step_pin=18, enable_pin=4, mode_pins=(21, 22,
27))

    """
    # 1.8 degree: nema23, nema14
    # software Control :
    # 'fullstep': A cycle = 200 steps
    # 'halfstep': A cycle = 200 * 2 steps
    # '1/4step': A cycle = 200 * 4 steps
    # '1/8step': A cycle = 200 * 8 steps
    # '1/16step': A cycle = 200 * 16 steps
    # '1/32step': A cycle = 200 * 32 steps
    """

    Motor1.SetMicroStep('software', 'fullstep')
    Motor1.TurnStep(Dir='forward', steps=200, stepdelay = 0.005)
    time.sleep(0.5)
    Motor1.TurnStep(Dir='backward', steps=400, stepdelay = 0.005)
    Motor1.Stop()

    """
```

```

# 28BJY-48:
# software Control :
# 'fullstep': A cycle = 2048 steps
# 'halfstep': A cycle = 2048 * 2 steps
# '1/4step': A cycle = 2048 * 4 steps
# '1/8step': A cycle = 2048 * 8 steps
# '1/16step': A cycle = 2048 * 16 steps
# '1/32step': A cycle = 2048 * 32 steps
"""

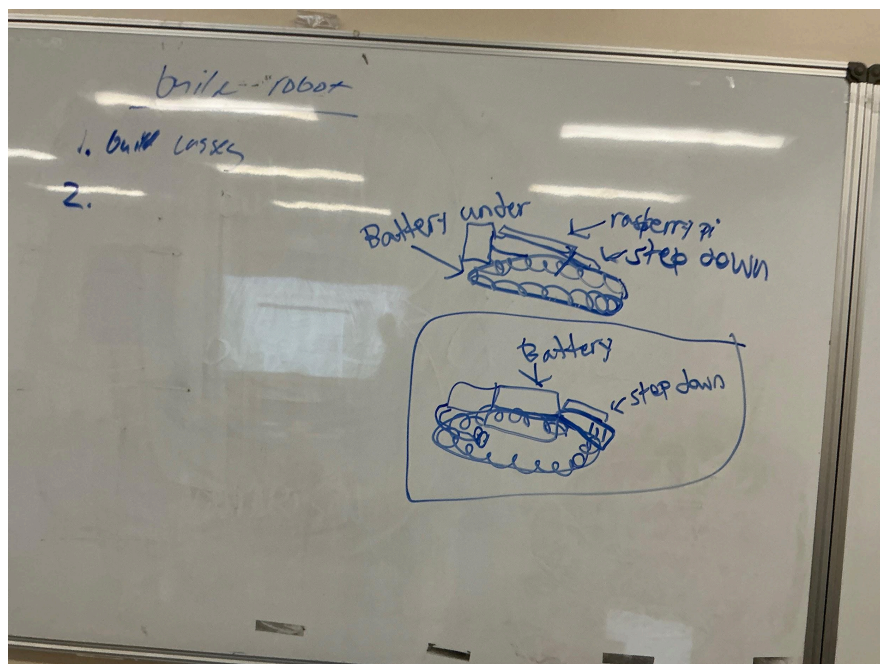
Motor2.SetMicroStep('hardward' , 'halfstep')
Motor2.TurnStep(Dir='forward', steps=2048, stepdelay=0.002)
time.sleep(0.5)
Motor2.TurnStep(Dir='backward', steps=2048, stepdelay=0.002)
Motor2.Stop()

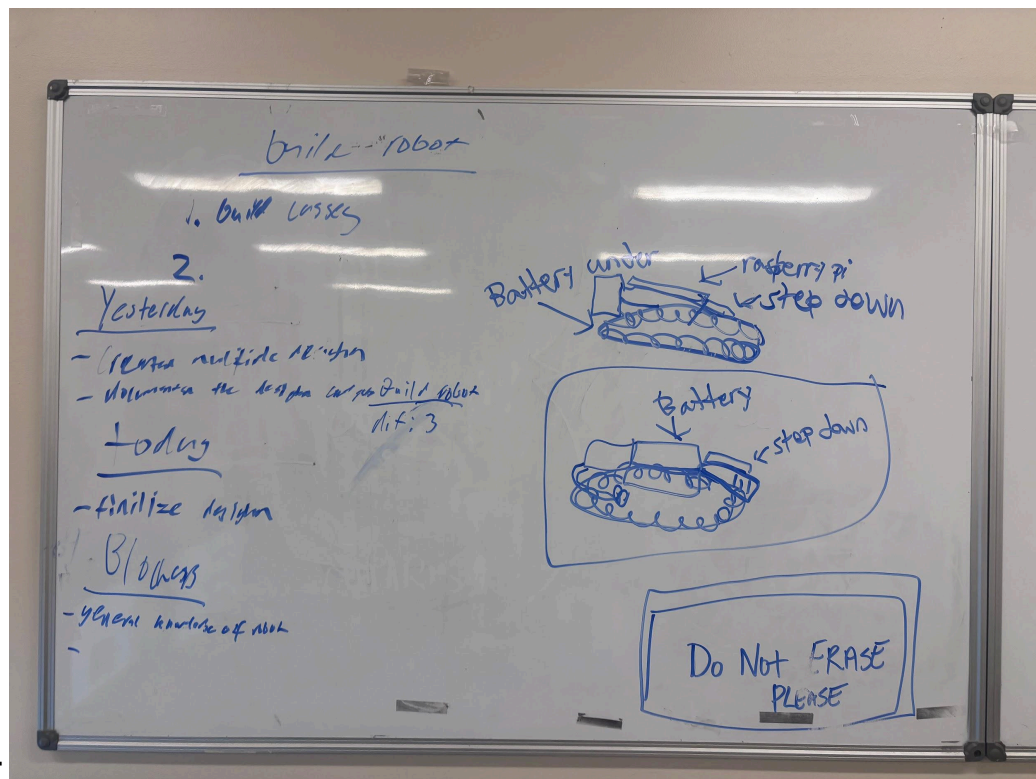
Motor1.Stop()
Motor2.Stop()

except:
    # GPIO.cleanup()
    print ("\nMotor stop")
    Motor1.Stop()
    Motor2.Stop()
    exit()

```

## Daily 1:





Daily 2:

raspberry pi to stepper motor waveshare python