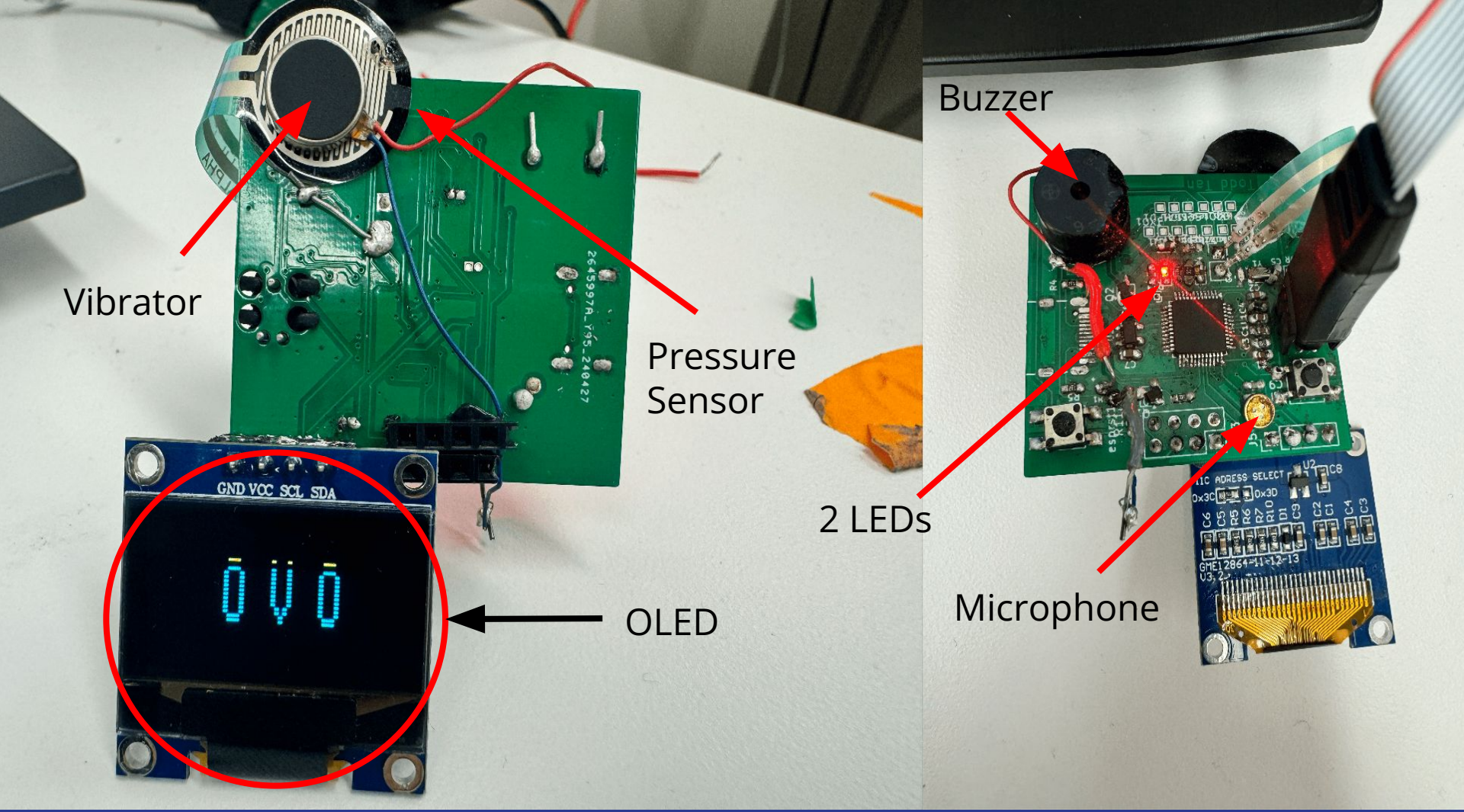
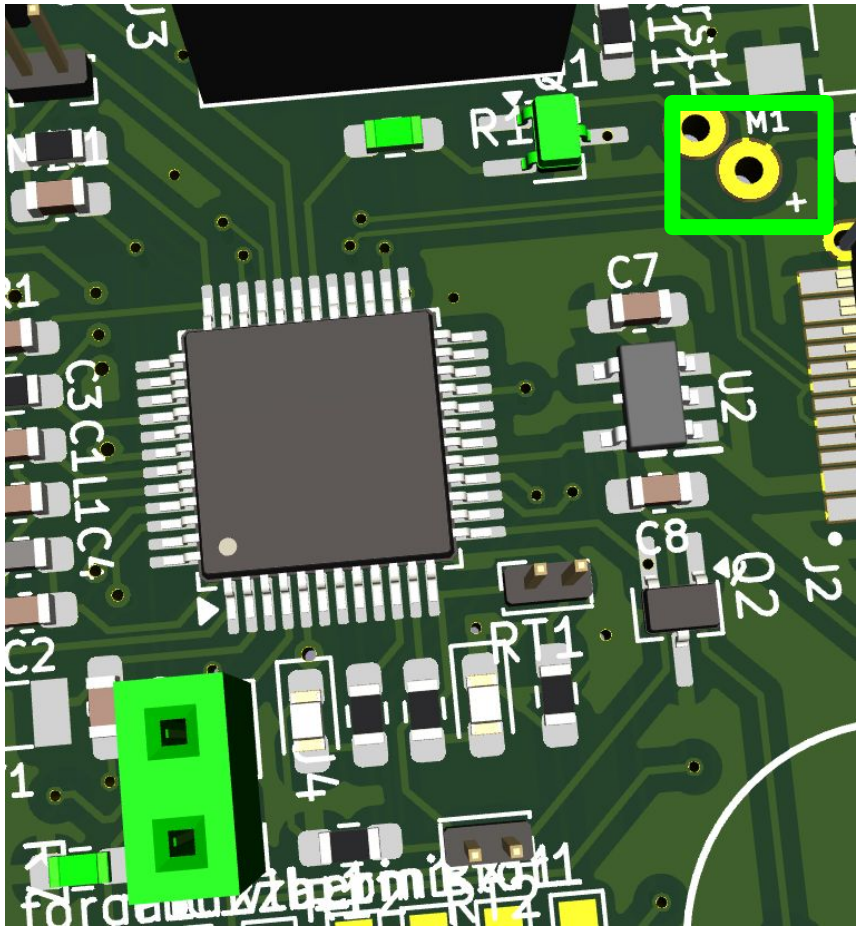
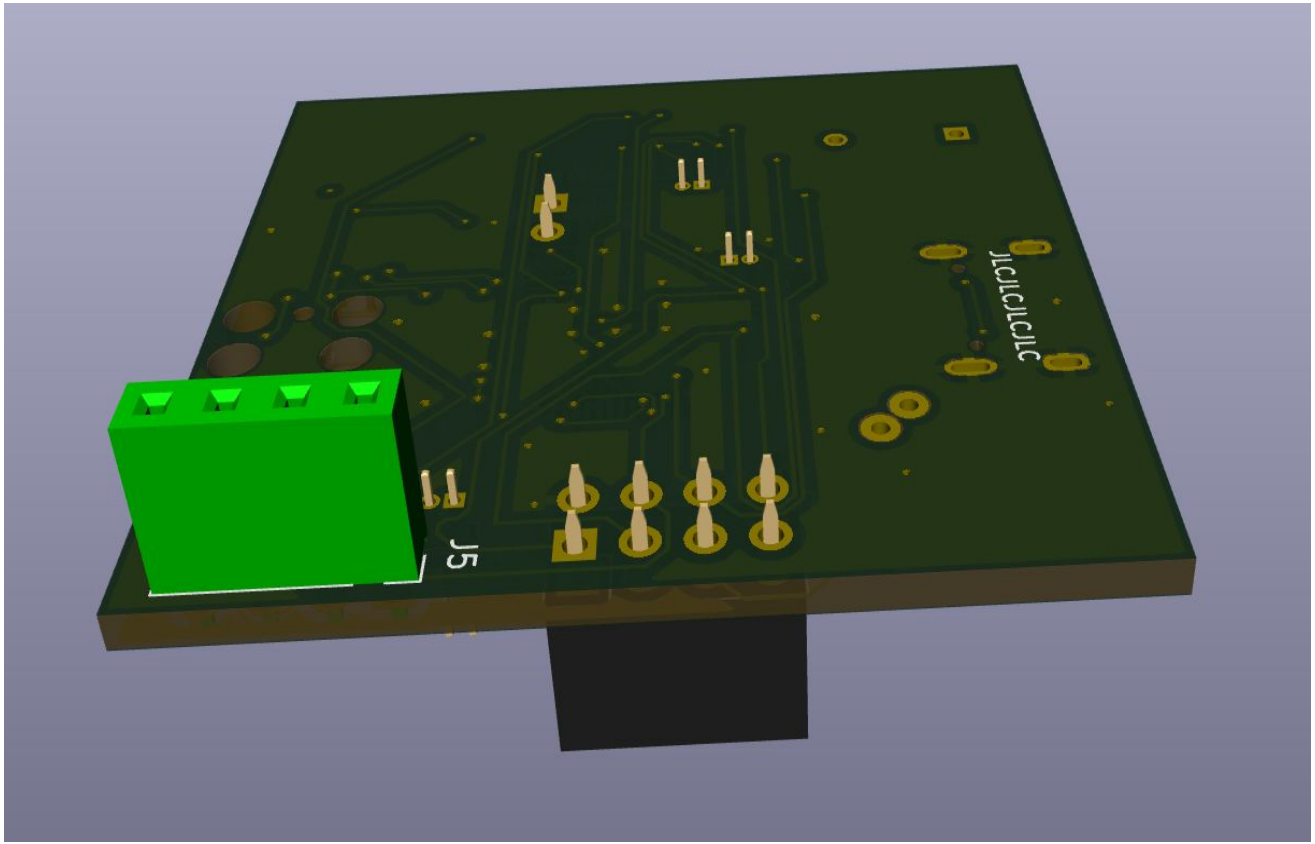
Youtube: https://youtu.be/oqg4fr0NS5o


Vibrator
Pressure Sensor
GND VCC SCL SDA
OLED
Buzzer
2 LEDs
Microphone



- Both vibrator and pressure sensor are extendable from the main PCB so that they could be attached together to create haptic feedbacks to the user



Other components are on the back while only OLED is facing the user

Tap, Double Tap, Hold

Start a 2 seconds timer when the first tap(i.e. force sensor reading > 200) is captured

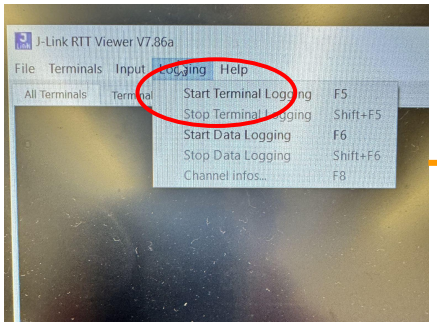
HIGH - LOW - HIGH = “DOUBLE_TAP”

HIGH - LOW = “SINGLE_TAP”

HIGH - HIGH = “LONG_PRESS”

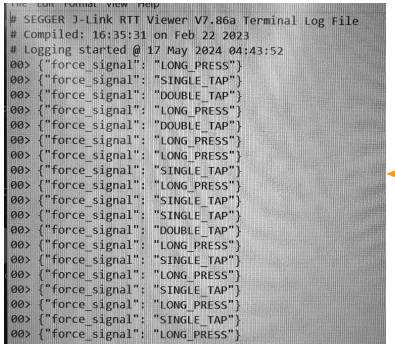
```
//no signal(pressure sensor not pressed)
// No Signal: No force applied at all, no signal means nothing to do; continue the previous action
// Single Tap: Force applied only once, if after 4 seconds there's not a second tap, perform action #1
// Double Tap: Force applied twice, within 4 seconds there are 2 taps(High - Low - Hight)
// Long Press: Force applied only once, but within 4 seconds, analogRead always High
// Timing starts when the first tap occurs
force_current_time = millis();
current_force_val = analogRead(force_sensor);
if(has_tap == false && current_force_val <= 200){
  //rtt.write("{\"force_signal\": \"NO_SIGNAL\"}");
}
//singal tap
else if(has_tap == false && current_force_val > 200){
  has_tap = true;
  //start the 4 second window
  force_last_time = millis();
  prev_force_val = current_force_val;
}
else if(has_tap == true){
  if(force_current_time - force_last_time >= 2000){//check if 2s timeframe has passed
    if(has_release == true){//single tap
      rtt.write("{\"force_signal\": \"SINGLE_TAP\"}\n");
      led_state = 1;
    }else{//long press
      rtt.write("{\"force_signal\": \"LONG_PRESS\"}\n");
      led_state = 0;
      delay(2000);
    }
    //reset
    has_tap = false;
    has_release = false;
    prev_force_val = 0;
    force_last_time = millis();
  }
  else if(current_force_val <= 200){
    has_release = true;
  }
  else if(current_force_val > 200 && has_release == true){
    rtt.write("{\"force_signal\": \"DOUBLE_TAP\"}\n");
```

PCB <-> Robot Communication Pipeline



RTT Viewer

write



data.log

read

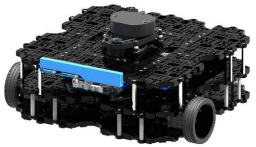
```
def read_last_line(file_path):
    with open(file_path, 'r') as file:
        lines = file.readlines()
        if len(lines) > 3:
            # Return the last line afte
            return lines[-1].strip()
        else:
            return None
```

read_data.py

HTTP GET

FLASK APP

Amazon EC2

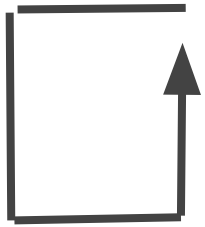


turtlebot3

HTTP GET

{“force_signal”: “**SINGLE_TAP**”}

Robot Commander



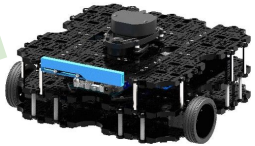
Follow

Drive

`DOUBLE_TAP`

`SINGLE_TAP`

`LONG_PRESS`



```python
def run_command(command):
    global current_process
    # Terminate the current process if it is running
    if current_process and current_process.poll() is None:
        current_process.terminate()
        current_process.wait()
    try:
        # Start a new process
        current_process = subprocess.Popen(command, shell=True)
    except subprocess.CalledProcessError as e:
        print(f"Command '{command}' failed with error: {e}")

def main():
    prev_action = None
    while True:
        force_signal = get_force_signal()
        # force_signal = input()
        if force_signal != prev_action:
            if force_signal == "SINGLE_TAP":
                print("SINGLE_TAP(drive_square) command going to run...")
                run_command("rosrun final_project drive_square.py")
            elif force_signal == "DOUBLE_TAP":
                print("DOUBLE_TAP(follow_person) command going to run...")
                run_command("rosrun final_project follow_person.py")
            elif force_signal == "LONG_PRESS":
                print("LONG PRESS(stop_robot) command going to run...")
                run_command("rosrun final_project stop_robot.py")
            prev_action = force_signal
        time.sleep(1)
```