B.7 Safety and Operating Instructions

This section provides important safety guidance and basic operating procedures to ensure safe and effective use of the pickleball throwing machine.

B.7.1 Safety Instructions

General Safety

- Only operate the machine on a flat, stable surface.
- Keep hands, tools, and loose clothing away from all moving parts during operation.
- Ensure the launch zone is clear before powering on the motors.
- Always supervise the machine while powered on.

Electrical Safety

- Double-check all wiring connections before connecting the battery.
- Use proper wire gauge for power connections (minimum 12 AWG for motor and battery circuits).
- Never short the battery terminals.
- Power off the system before performing any maintenance or component swaps.

Battery Handling

- Use a sealed lead-acid (SLA) battery charger only.
- Do not overcharge or puncture the battery.
- Secure the battery firmly to prevent shifting during use.

Emergency Stop Procedure

- Press the “Stop All” button on the UI to immediately stop all motors and servos.
- Alternatively, use the hardware power switch to cut off power to the entire system.

Environmental Use

- This machine is not weatherproof. Do not operate in rain or high humidity.
- For outdoor use, ensure electronics are shielded and the surface is dry.

B.7.2 Operating Instructions

1. Power On the System
    - Connect the battery.
    - Flip the hardware power switch to “ON”.
    - Wait for the Raspberry Pi and UI to fully boot.

2. Configure Settings on the Touchscreen UI
   - Adjust Motor 1 and Motor 2 sliders to control topspin and backspin.
   - Set Servo 1 (feed speed) to Slow, Fast, or Off.
   - Use Servo 2 controls to rotate the launch angle left or right.
   - Set a launch delay (optional) to allow time to move into position.
   - Press “Start” to begin the launch sequence.
3. Stop Operation
   - Press “Stop” to halt all motors and servos immediately.
   - Power off the system after use to preserve battery life.

B.7.3 Maintenance and Storage Tips

- After each use, inspect moving parts for debris or damage.
- Recharge the battery after each session.
- Store the machine in a dry, temperature-controlled environment.
- Periodically check printed parts for wear or cracks, especially gears and servo mounts.

B.8 Troubleshooting Guide

This guide outlines common issues users may encounter while assembling or operating the pickleball throwing machine, along with suggested solutions.

Issue 1: UI Does Not Launch on Startup

Possible Causes:

- `.desktop` file is not executable
- Incorrect file path in `.desktop` configuration
- Missing Python libraries (e.g., Kivy)

Solutions:

- Run the following command to make the file executable:
  ```
  chmod +x /home/pi/Desktop/PickleballApp.desktop
  ```
- Double-check that the `Exec` line points to the correct path for `UI.py`
- Manually launch the app to test it:
  ```
  python3 /home/pi/UI.py
  ```
- Reinstall libraries with:
  ```
  pip3 install kivy --break-system-packages
  ```

Issue 2: I2C Devices Not Detected (No Response from PCA9685)

Possible Causes:

- SDA/SCL wiring is incorrect
- I2C interface is not enabled
- 3.3V or GND is not connected

Solutions:

- Check wiring:
  SDA → GPIO 2 (Pin 3)
  SCL → GPIO 3 (Pin 5)
- Confirm that I2C is enabled via `sudo raspi-config`
- Run `i2cdetect -y 1` to scan for devices. You should see an address like `0x40` for PCA9685
- If no address appears, double-check voltage and ground connections

---

Issue 3: Motors Not Spinning

Possible Causes:

- Incorrect GPIO wiring

- Motor drivers not receiving power
- Python script not sending signals properly

Solutions:

- Confirm motor driver connections to GPIO pins as per your circuit diagram
- Use a multimeter to check voltage across motor terminals
- Test basic motor control via Python to isolate software/hardware:

    -
      ```
      import RPi.GPIO as GPIO

      GPIO.setmode(GPIO.BCM)

      GPIO.setup(20, GPIO.OUT)

      GPIO.output(20, GPIO.HIGH)
      ```

Issue 4: Servos Not Responding or Spinning Constantly

Possible Causes:

- PWM signal not reaching servo
- Incorrect I2C address or driver issue
- Servo control code not functioning

Solutions:

- Verify I2C connection and PCA9685 wiring
- Try Adafruit's ServoKit test example to isolate servo response
- Double-check servo channel indexing in your code (e.g., `kit.continuous_servo[0]`)

Issue 5: Raspberry Pi Crashes or Freezes

Possible Causes:

- Underpowered system
- Voltage drop from motors or screen
- Overloaded CPU

Solutions:

- Ensure Raspberry Pi is powered through a regulated 5V line using the LM2596 step-down converter
- Power the Pi separately from high-draw components like motors
- Avoid running background processes that consume CPU resources

---

General Tips:

- Always reboot after enabling new interfaces or modifying config files

- Use short, clean wiring and proper connectors to minimize noise and voltage drops

- Label wires to make troubleshooting easier

- Search online forums (e.g., Raspberry Pi Stack Exchange) for issues related to specific libraries or hardware setups