MJTRONIKS.COM

Mojobot Nano User Manual

Introduction

Mojobot Nano by MJtroniks is an innovative robot based on Arduino Nano 33 BLE, designed for educational purposes and hobbyist projects. It offers a versatile platform for learning and experimentation.

Programming Platforms

Micropython: Explore advanced robotics and sensor integration with a high-level programming approach.

C++: Dive deep into low-level control and optimization of robotics functionalities.

Makecode: An accessible way for beginners to start programming Mojobot Nano, providing a visual programming experience.

Get Started

- Micropython
- C++
- Makecode
- Scratch
- Java

Features

Path Tracking: Utilizes infrared sensors to follow designated paths.
Obstacle Avoidance: Employs an ultrasonic sensor to detect and avoid obstacles.
LED Indicators: Provides visual feedback through LEDs.
BLE Control: Enables remote control and monitoring via Bluetooth Low Energy.

Robot parts

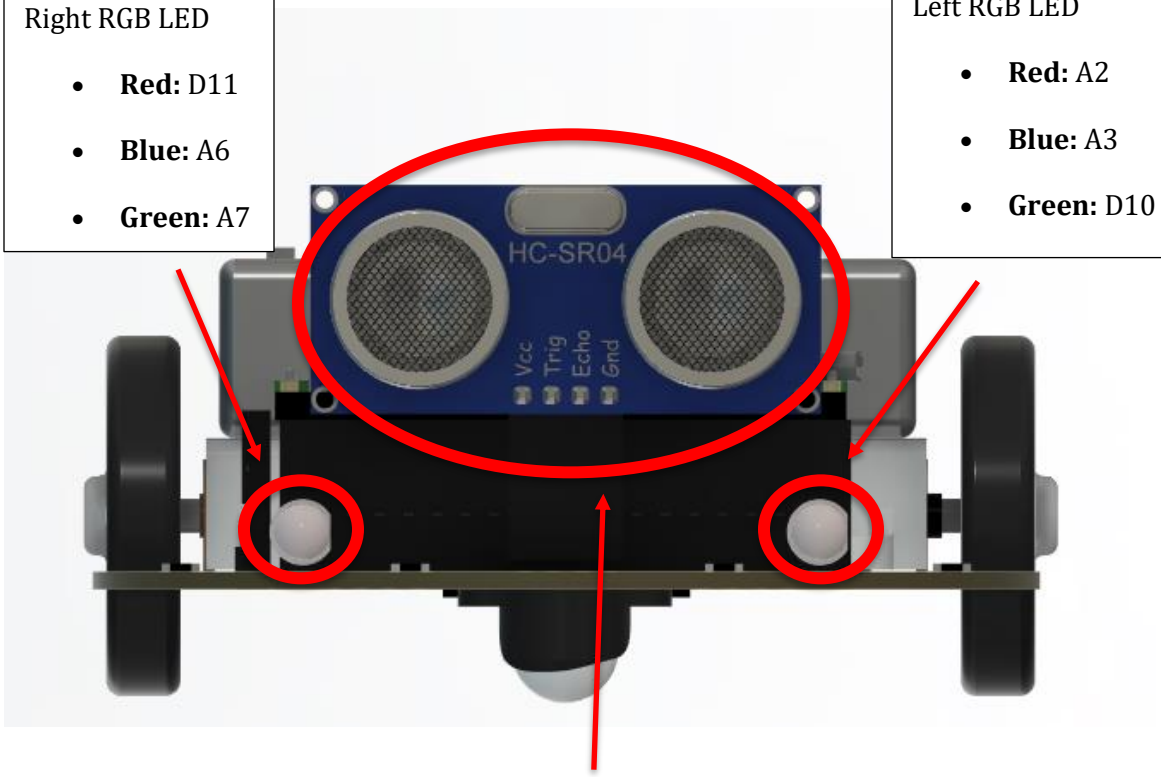
FRONT VIEW

Right RGB LED

- **Red:** D11
- **Blue:** A6
- **Green:** A7

Left RGB LED

- **Red:** A2
- **Blue:** A3
- **Green:** D10



3V Ultrasonic sensor

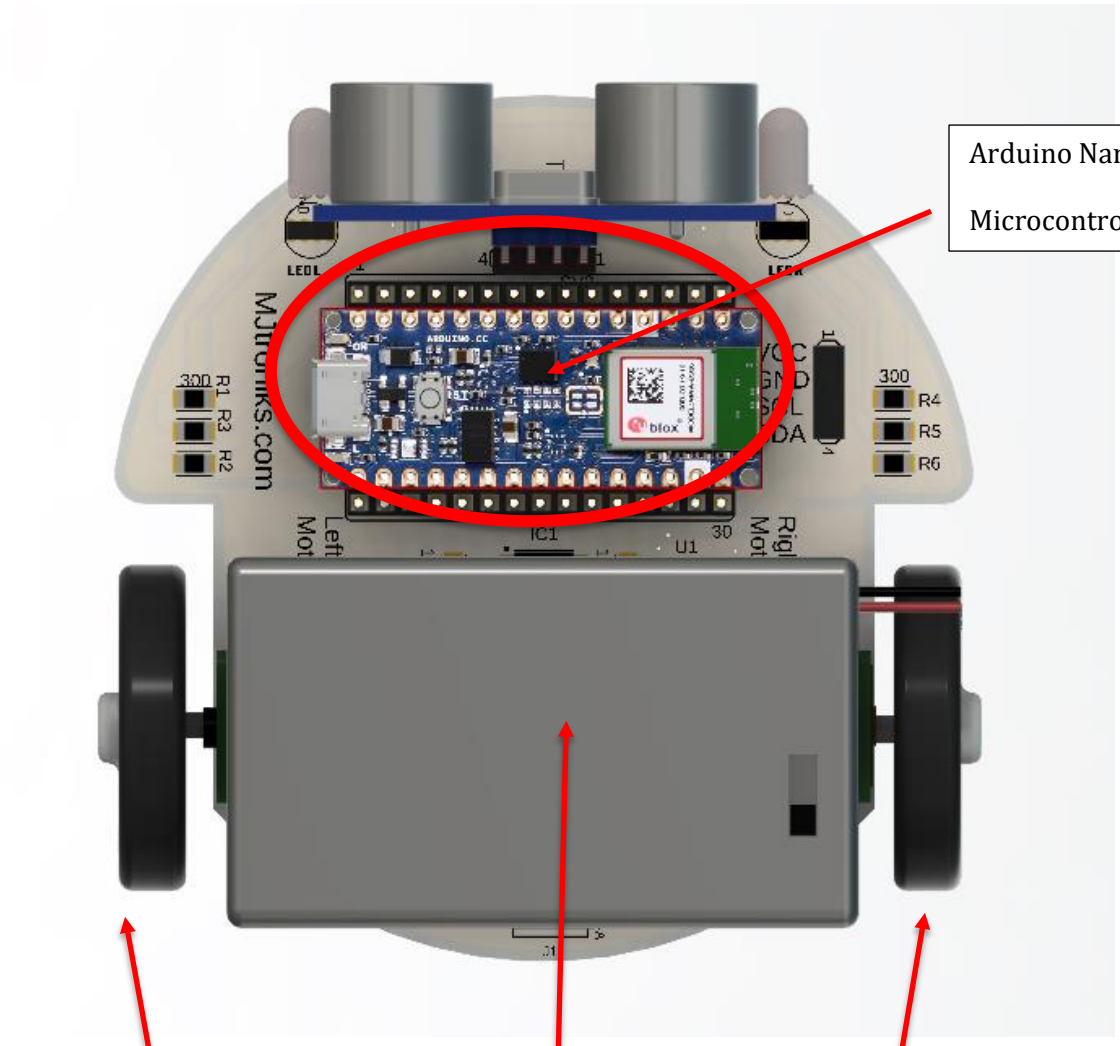
VCC

Trigger (D2)

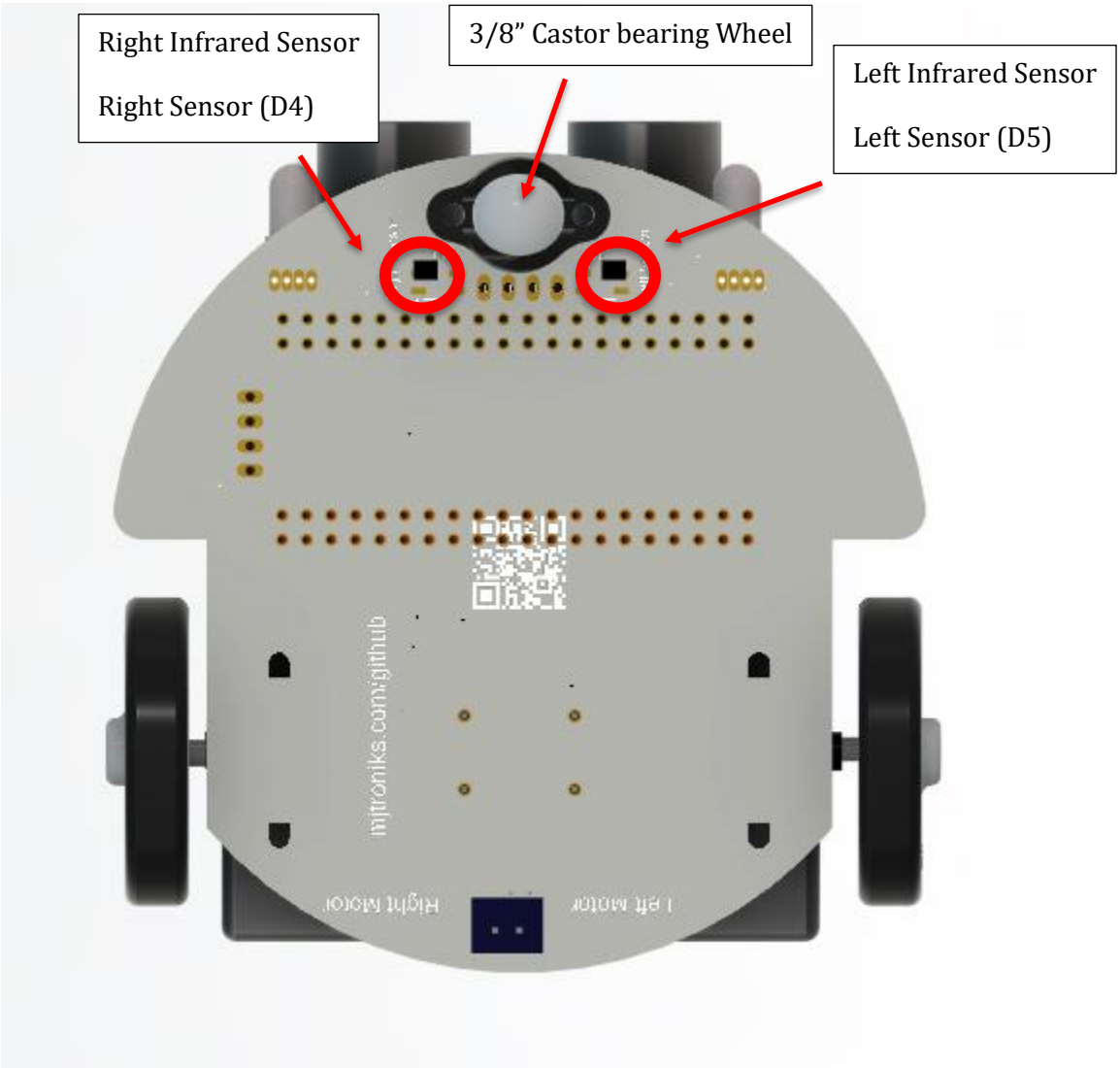
Echo (D3)

GND

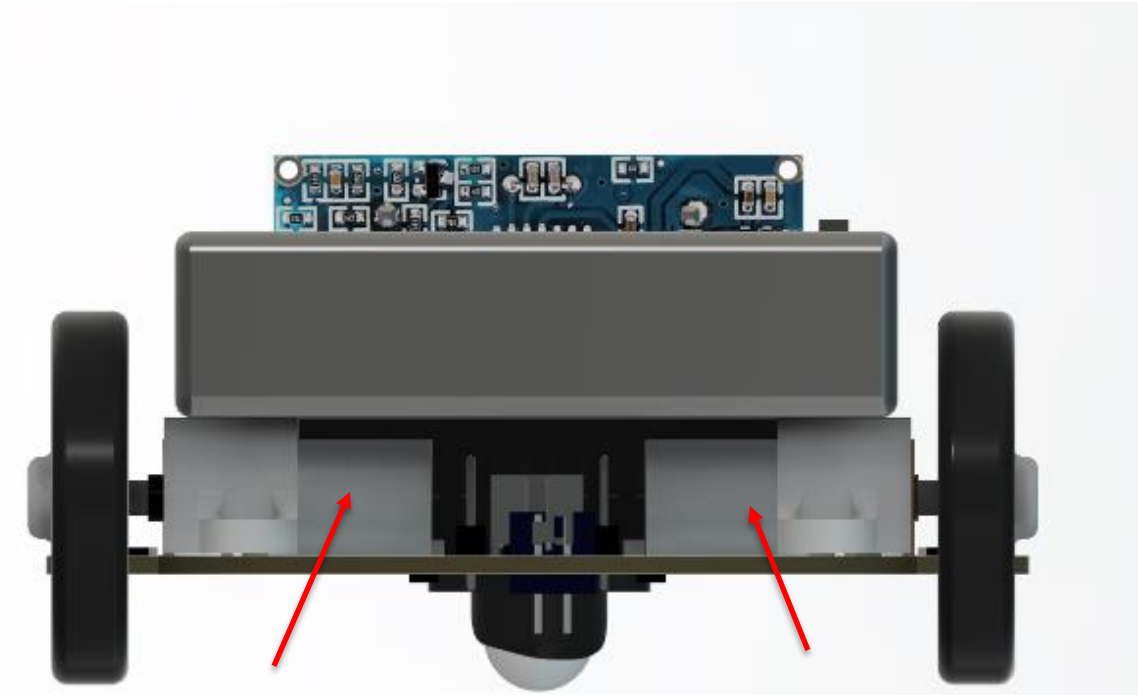
TOP VIEW


Arduino Nano 33 BLE
Microcontroller
Left Wheel
3 X AAA Battery holder
Right Wheel

BOTTOM VIEW


Right Infrared Sensor
Right Sensor (D4)
3/8” Castor bearing Wheel
Left Infrared Sensor
Left Sensor (D5)
mitroniks.com/github
Right Motor
Left Motor

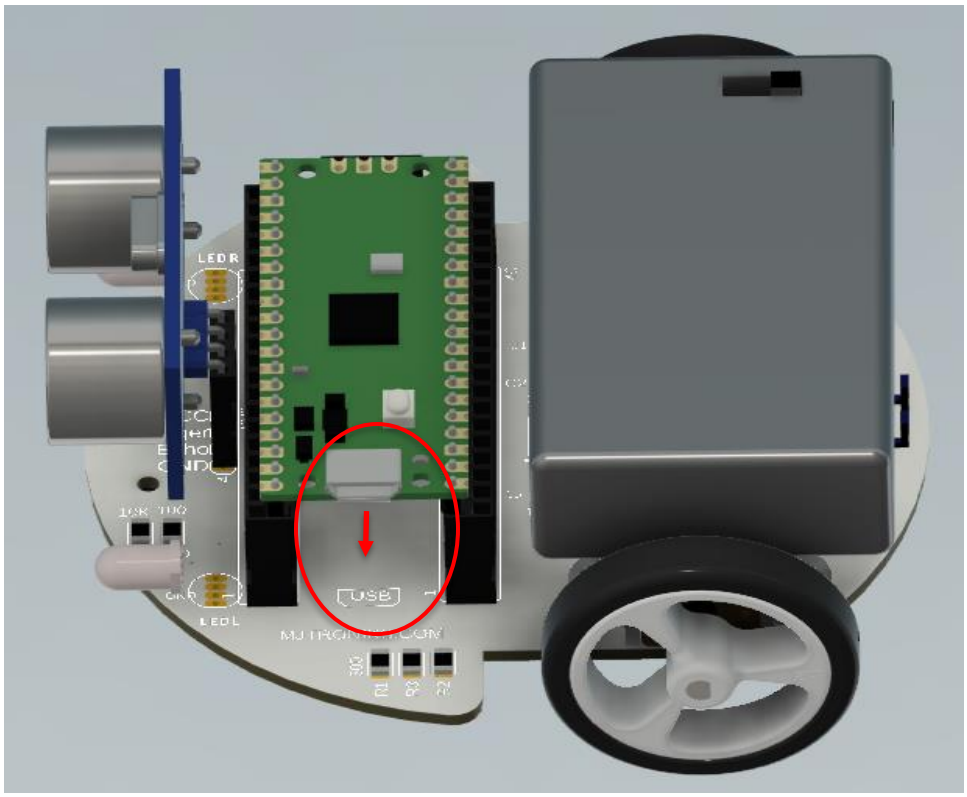
BACK VIEW



LEFT Motor M1 Motor 1 PWM (D7) (motor 1 speed) Direction (D8) Forward/Backward	RIGHT Motor M2 Motor 2 PWM (D6) (motor 2 speed) Direction (D9) Forward/Backward

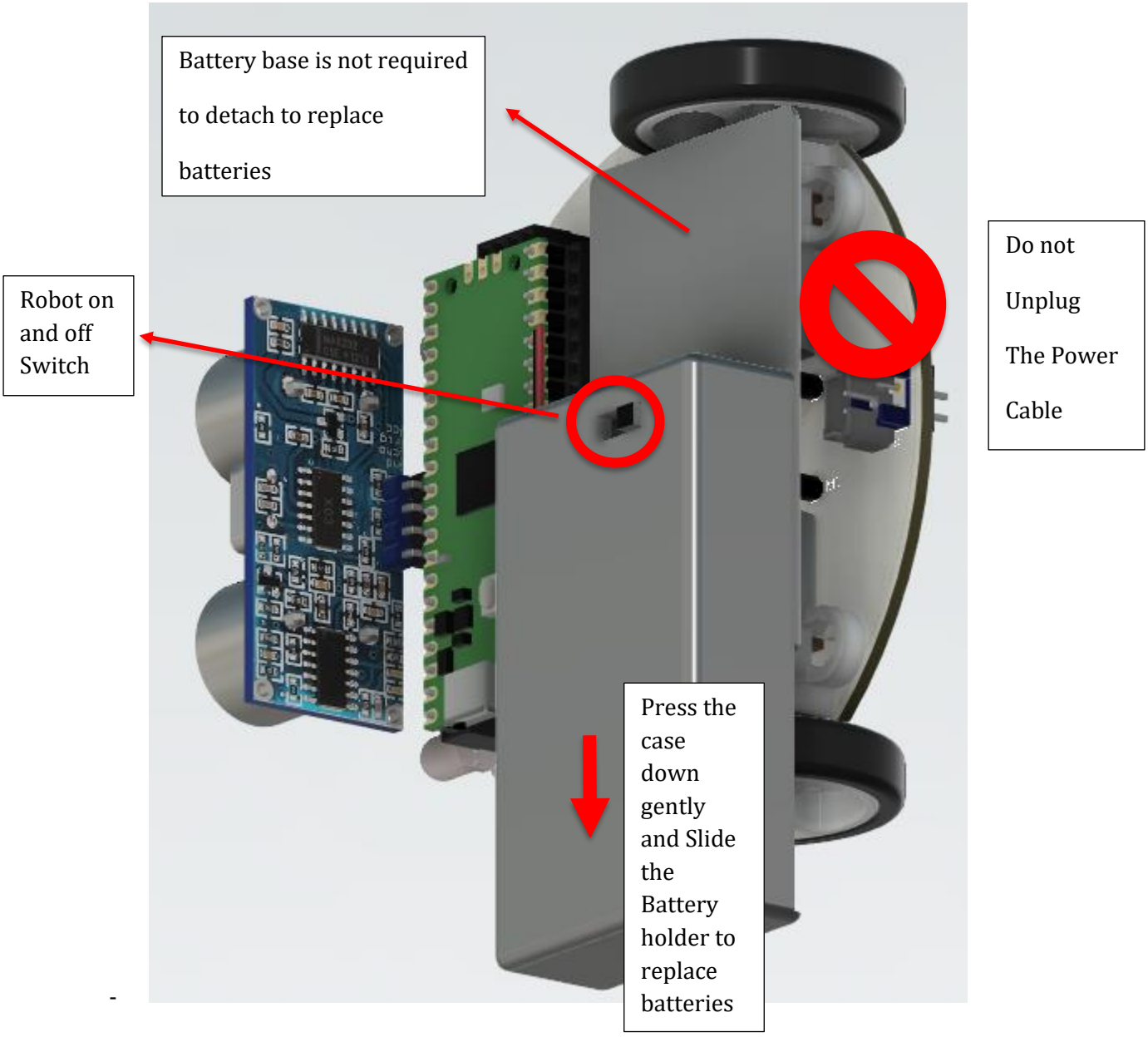
Handling Instructions

1. Microcontroller USB Alignment: Ensure the microcontroller USB matches the PCB board USB mark.

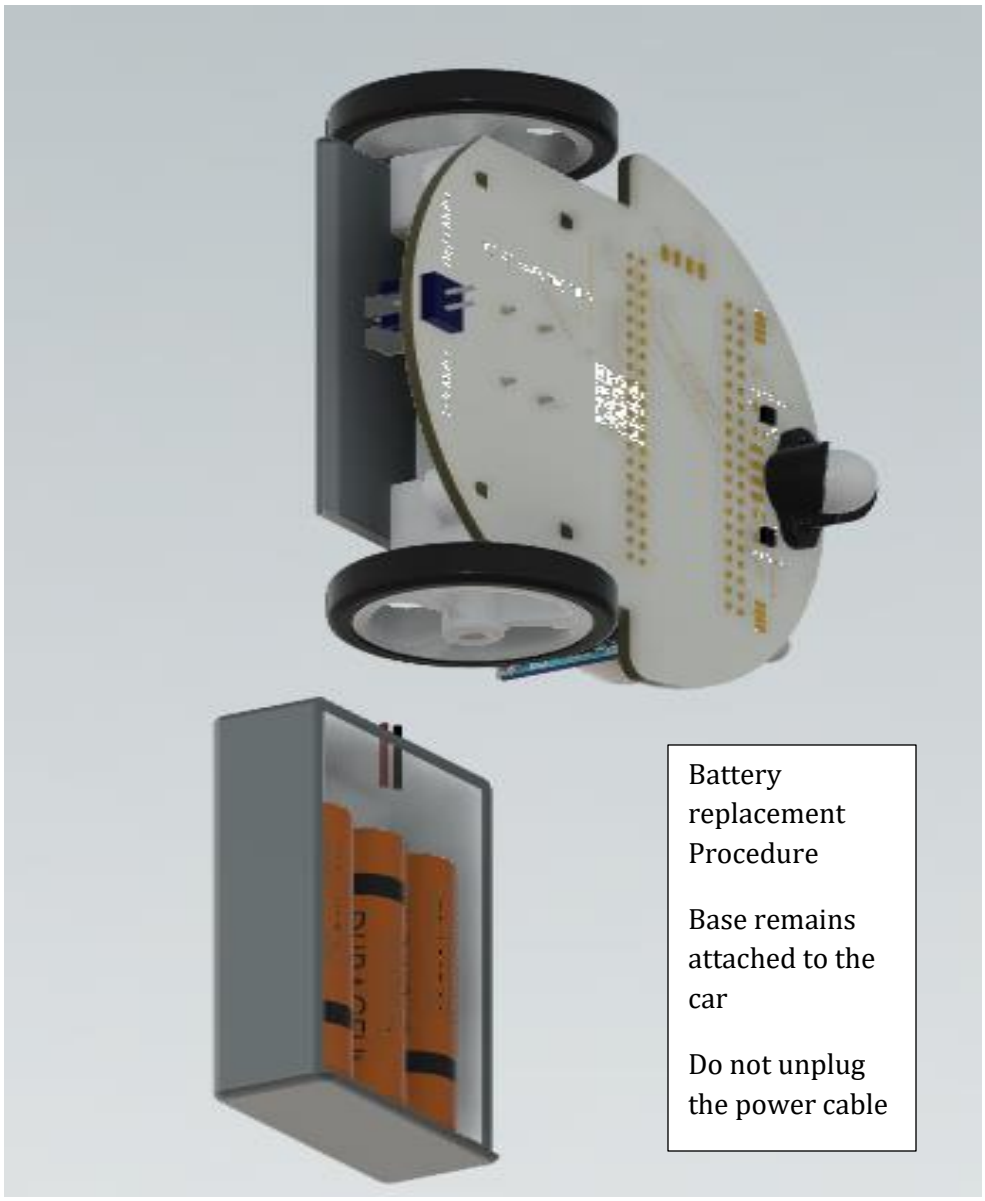


2. Battery Pack:

- Do not unplug the battery pack from the power connector.
- Remove batteries by sliding open the battery holder case.
- Use the battery holder case switch to turn the robot on and off.



-

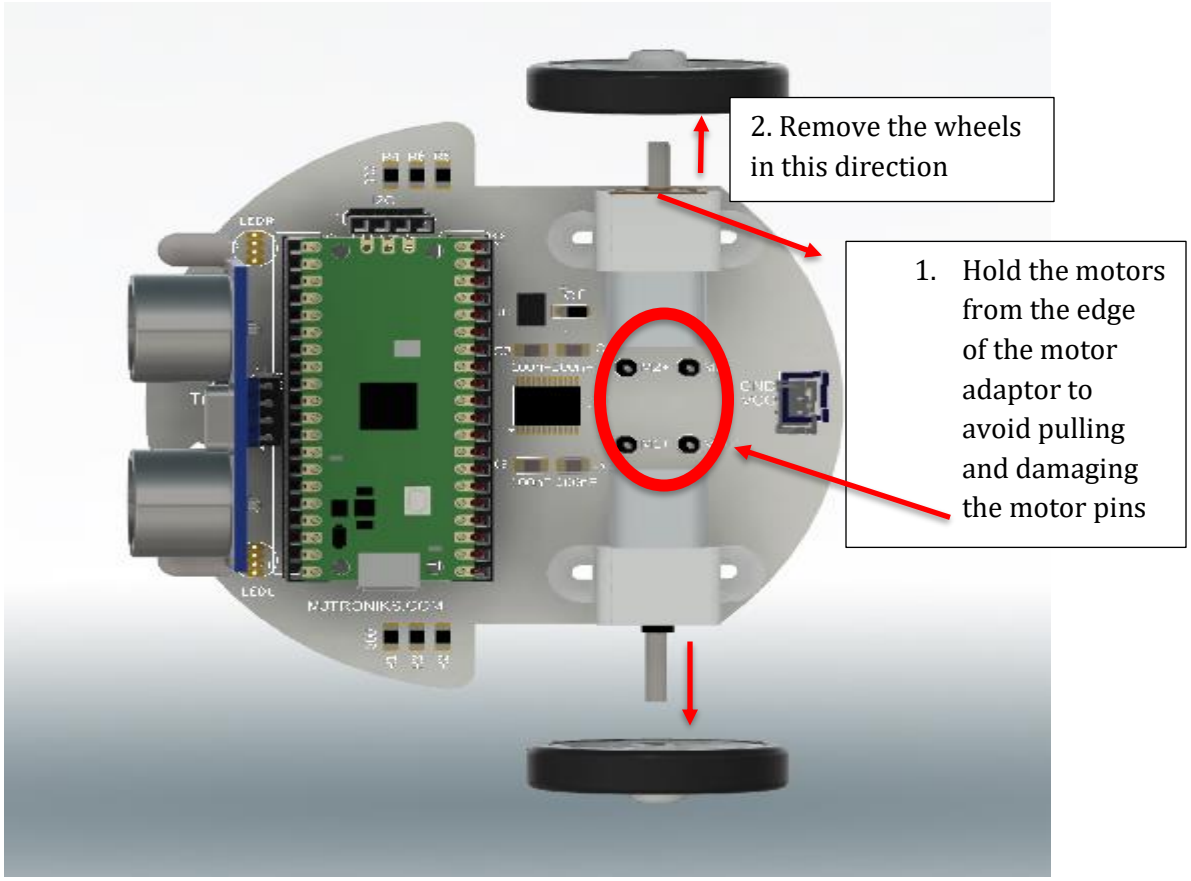


Battery replacement Procedure

Base remains attached to the car

Do not unplug the power cable

3. Wheel Removal: Remove the battery holder and the proceed to remove the wheels gently, holding the motors to avoid damaging motor pins.



Assembly and Components

- Components Included:

- 1 x Mojobot car

- 1 x Battery Holder

- 1 x HC-SR04 Ultrasonic Sensor

- 1 x Arduino Nano 33 BLE

- 1 x Robot stand

- 1 x Line following map

Pins description

Sensors and Actuators

Infrared Sensors:
Left: Connected to Pin D5
Right: Connected to Pin D4
Ultrasonic Sensor:
Trigger Pin: D2
Echo Pin: D3

Motors

Left Motor:
PWM: D7
Direction: D8
Right Motor:
PWM: D6
Direction: D9

LEDs

Left LEDs:
Red: A2
Blue: A3
Green: D10
Right LEDs:
Red: D11
Blue: A6
Green: A7

Pin Utilization

Used Pins
D2, D3, D4, D5, D6, D7, D8, D9, D10, D11
A2, A3, A6, A7
Available Pins (Available pins can be used by soldering 2 15x1 Female headers)
A0, A1, A4, A5
D0, D1, D12, D13 (Note: D13 typically has an LED connected to it on most Arduino boards)

Power Requirements

- Voltage: 3.5V to 4.5V

- Batteries: 3 X AAA batteries (ensure correct placement)

Software Setup Overview

Detailed setup instructions can be found at https://github.com/mjtroniks/Mojobot/wiki

1. Micropython:

- Install Thonny IDE from thonny.org
- Download Micropython firmware for Arduino Nano 33 BLE from micropython.org/download/
- Follow the Thonny installation guide to flash the firmware onto the Arduino Nano 33 BLE.
- Write and execute your Micropython code within Thonny IDE.

2. MakeCode:

- Visit makecode.microbit.org
- Create a new project and select the device as Arduino Nano 33 BLE.
- Use the drag-and-drop blocks to create your program.
- Download the generated UF2 file and drag it to the Nano 33 BLE mounted as a USB drive.

3. C++:

- Install Visual Studio Code and CMake.
- Set up the Nano 33 BLE SDK and toolchain by following the official Arduino Nano 33 BLE C++ setup guide.
- Write your C++ code and use CMake to build the project.
- Flash the binary onto the Nano 33 BLE using a drag-and-drop method.

Tutorials

Micropython Introduction and Blink (Click for detailed instructions)

To begin with Micropython, you'll need to install the necessary software and set up your environment. Here's a simple blink program to get you started:

```python
import machine
import utime

led = machine.Pin(13, machine.Pin.OUT)  # Use built-in LED pin 13

while True:
    led.value(1)
    utime.sleep(1)
    led.value(0)
    utime.sleep(1)
```

MakeCode Introduction and Blink (Click for detailed instructions)

MakeCode is a user-friendly platform for beginners. Here's how to create a blink program:

1. Go to makecode.microbit.org.
2. Create a new project.
3. Drag the `show leds` block and create a pattern.
4. Use `pause` to control the blinking interval.

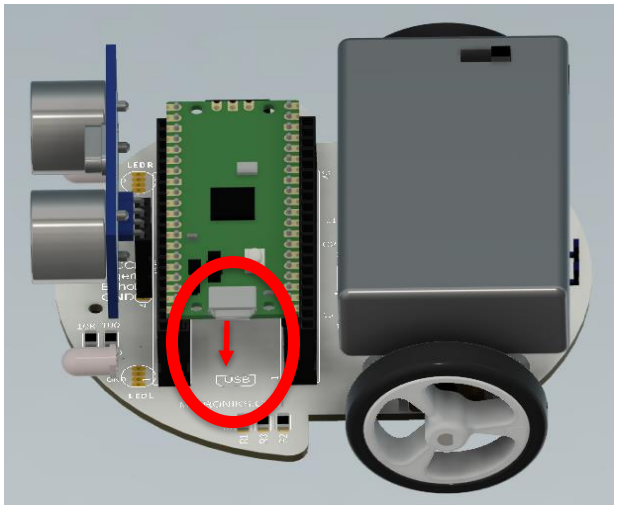
C++ Introduction and Blink (Click for detailed instructions)

For C++, set up your Arduino development environment and write your first program to blink an LED:

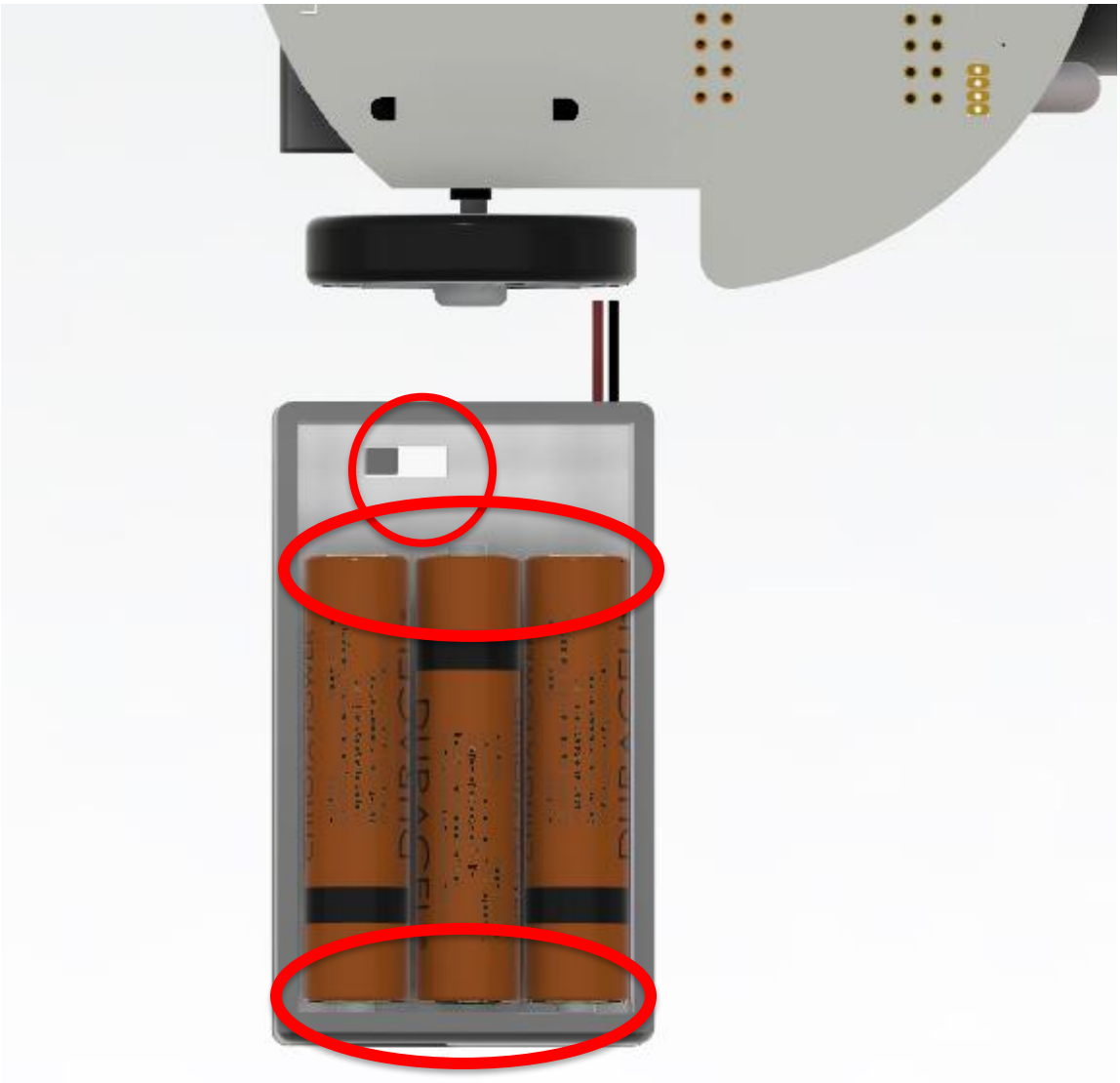
```cpp
#include <Arduino.h>

void setup() {
  pinMode(13, OUTPUT);  // Use built-in LED pin 13
}

void loop() {
  digitalWrite(13, HIGH);  // Turn the LED on
  delay(1000);             // Wait for 1 second
  digitalWrite(13, LOW);   // Turn the LED off
  delay(1000);             // Wait for 1 second
}
```

Troubleshooting

1. Microcontroller Placement: Check proper placement of the microcontroller, making sure the USB matches the USB mark on the board.



2. Battery Polarity: Ensure batteries are correctly placed with proper polarity.



3. When switching programming languages the UF2 image fails to upload, to resolve Re upload the image using Thony IDE as described in [this tutorial] section Steps to resolve

For further details and images, refer to the MJtronics Mojobot GitHub Repository (https://github.com/mjtroniks/Mojobot).

Applications

Mojobot Nano is ideal for educational purposes, allowing users to explore robotics, sensor integration, and programming. It serves as a great platform for learning about motor control, ultrasonic sensing, and infrared communication.

This versatile and user-friendly robot encourages creativity and experimentation in the field of robotics, making it an excellent choice for students and hobbyists alike.