Design and Control of Self Balancing Robot

Md. Imran [1],Bishaw Kirti Chakma [2] advised by Md. Khorshed Alom [1,2]
Khulna University of Engineering and Technology, Dept of Electronics and Communication Engineering

Abstract

Balancing robot is a mobile robot that has wheels on the right and left side, has the ability to maintain equilibrium. This robot is a development of an inverted pendulum models placed on a wheeled train.

GY-521 MPU6050 3-axis gyroscope and accelerometer sensors are used to detect the slope as well as to detect angular velocity and to maintain the position of the robot in a position perpendicular to the earth surface. PID control method is used to control the DC motors.

Trial and error is used to determine the value (tune) of the proportional and integral control parameters. The EZ-GUI Ground Station mobile app was used to control the robot's movement through Bluetooth. The robot mobility in this project is still not stable.

Introduction

A self-balancing robot is like an inverted pendulum, unlike a normal pendulum which oscillates back and forth when a force is applied. An inverted pendulum cannot balance it self. An example of this would be, trying to balance a broom stick on your fingers. You can't stay in one place, you have to move your finger back and forth. This is exactly what the vehicle is doing, its moving back and forth so that the center of gravity will always remain under the wheels.

To drive the motors we need some information on the state of the robot. We need to know the direction the robot is falling, how much the robot has tilted and the speed with which it is falling.

To get this information we will be using a combination of gyroscopic sensor and an accelerometer. The sensor I will be using is MPU6050. We combine all these inputs and generate a signal which drives the motors and keeps the robot balanced.

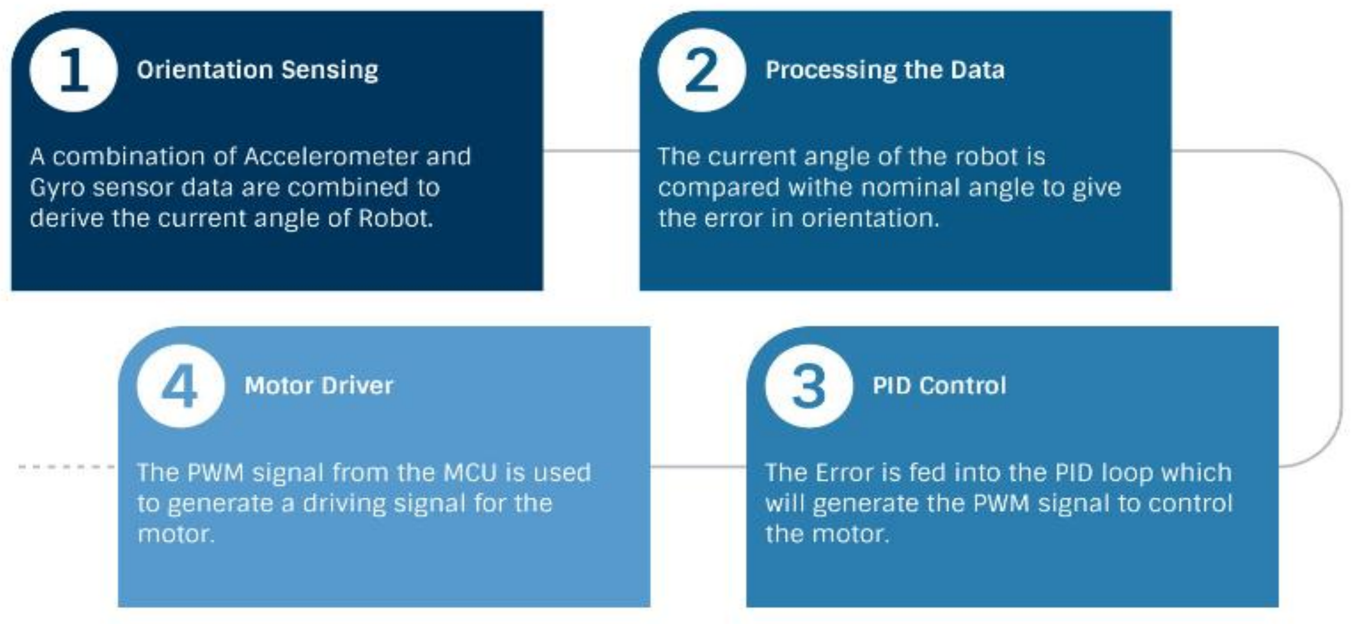

Figure 1. Flow Diagram of a Self-balancing robot

Methodology

To balance the robot, we don't just need the angle the robot is at, we also need to know the rate at which it is falling. The MPU6050 has a 3-axis accelerometer that measures acceleration and a 3-axis gyroscope that measures angular rate about the three axes.

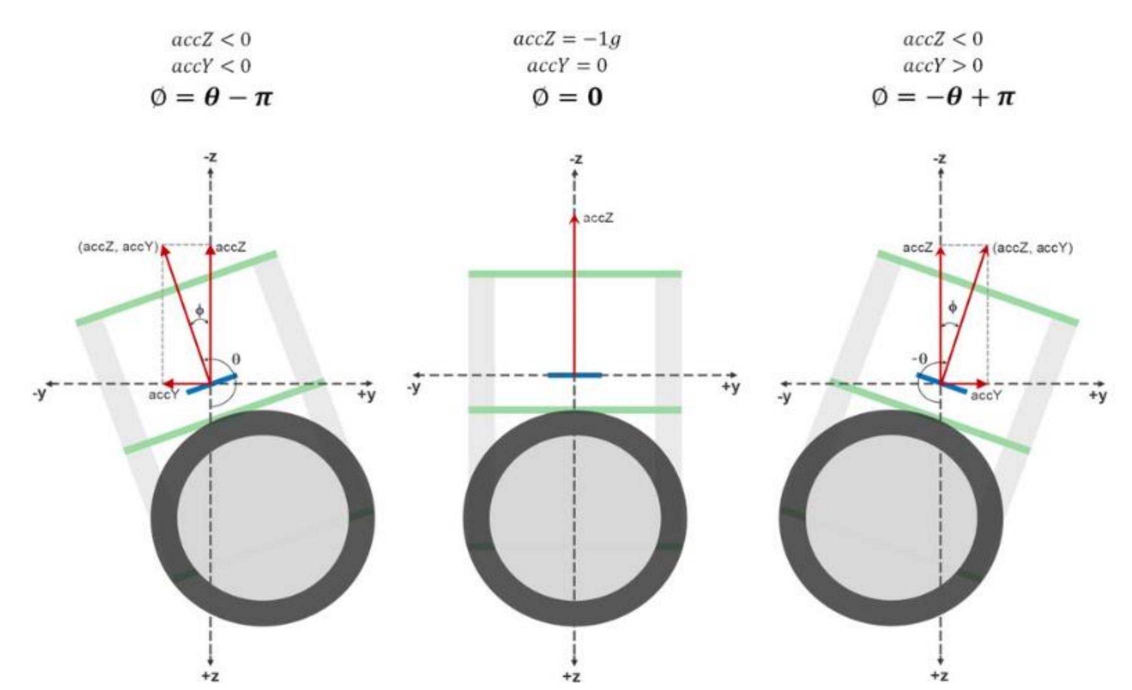

Figure 2. Measuring the angle of inclination

Gyro sensor provides the x, y, and z data to a microcontroller. Input data is sent to the microcontroller triggers via Bluetooth from an android device. Arduino nano balances the robot by determining precise data from the PID control system.

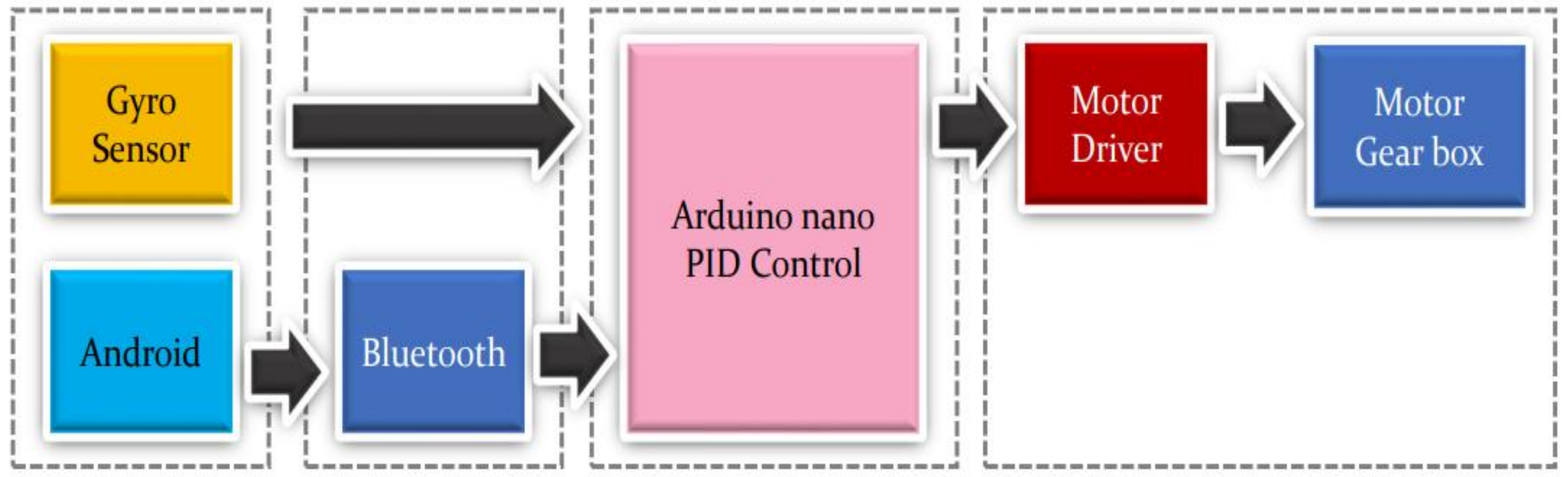

Figure 3. Block diagram of Balancing Robot

Project Requirements

1. Arduino Nano ________________ 1x
2. MPU605 Gyro sensor ____________ 1x
3. Nema 17 Stepper motors __________ 2x
4. 75mm Wheels ________________ 2x
5. A4988 Stepper driver IC ___________ 2x
6. HC-05 Bluetooth module __________ 1x
7. Arduino CNC Shield V3____________ 1x
8. Cardboard, LIPO Battery 2200 mAh ___ 1x
9. 145 mm Rods _________________ 4x
10. Wire, nuts and bolts (as required)

Useability & Application

In terms of general useability, we tried to make operating this robot as simple as possible. The user simply has to connect the robot to power and send command via Bluetooth from the android app. It is recommended that the robot lay down on a flat surface as the robot is powered on, because that allows it to successfully complete the gyroscope calibration procedure before it begins operation.

Some of it's applications are:

- The vehicle can rotate in place to instantly change its direction of motion and precisely navigate tight spaces that a three or four-wheeled robot cannot.
- It can be used in restaurants to serve food to customer just like waiters do today.
- They can also be used in hospitals to carry necessary medical supplies to patients autonomously.
- They can be used in malls for welcoming people entering the malls.

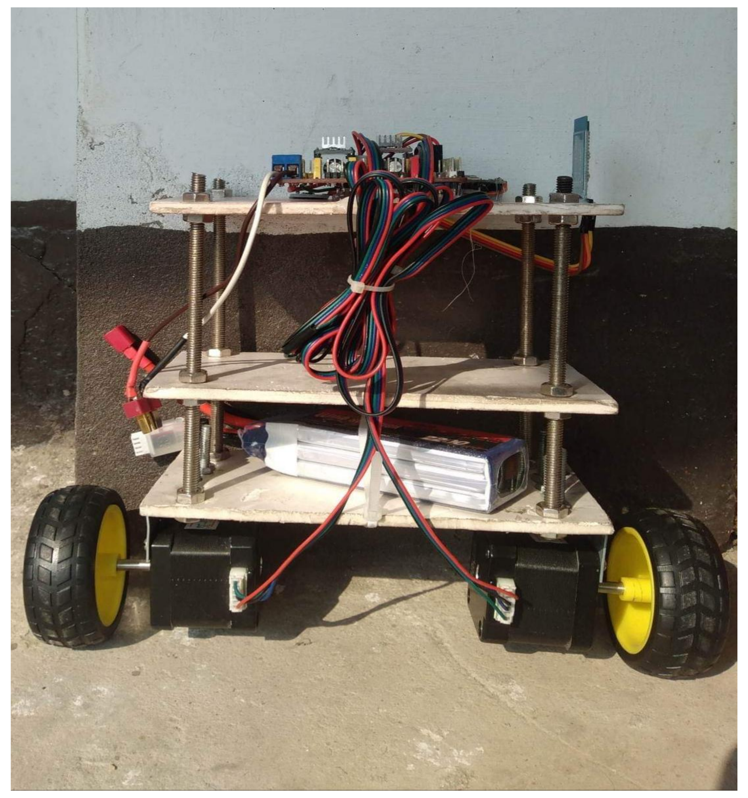

Figure 4. Robot in Balanced Position

Future Work

- Improve the design by using lightweight material resulting in better balancing.
- Fix the wobble by setting a minimum PWM duty cycle that is just below the threshold at which the motors begin to spin.
- Ability to carry load on its top while balancing.
- Implement automatic tilt calibration so that the robot will always remain balanced even if its center of gravity is shifted, requiring no action by the user.
- Implement manual remote steering of the robot, commanding it to move forward, backward, and rotate clockwise or counter-clockwise.
- Add a light sensing system to the bottom of the robot and make it autonomously follow a dark path drawn on the ground.