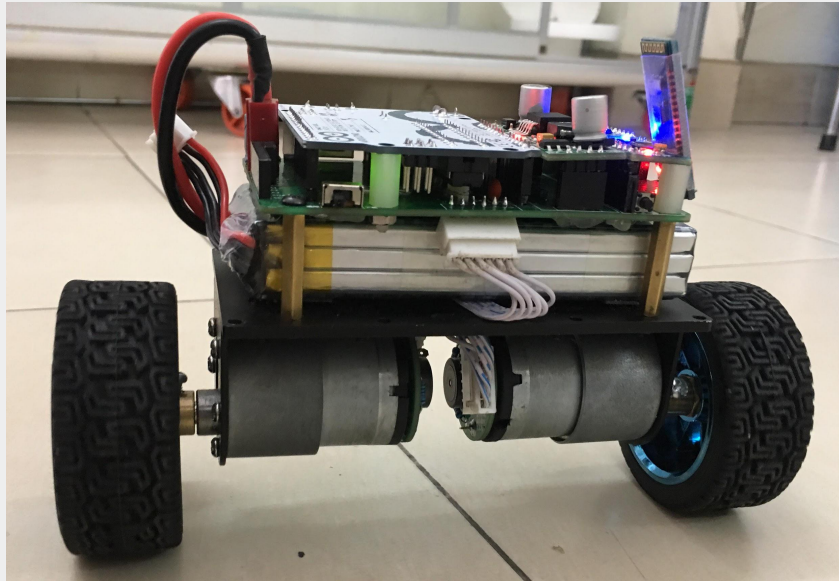


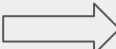
I. Tổng quan

- Giới thiệu mô hình xe tự cân bằng.
- Nguyên lí hoạt động, ý tưởng, tính năng.
- Vật tư, linh kiện.
- Quá trình thực hiện.
- Nạp chương trình, thử nghiệm
- Tổng kết.

1. Giới thiệu mô hình xe tự cân bằng.

- Hình ảnh



Video demo: 

2. Ý tưởng, tính năng và nguyên lí hoạt động.

2.1. Ý tưởng, tính năng.

- Ý tưởng: Lấy cảm hứng từ project của [maker_studio](#) trên instructables.

Link: <http://www.instructables.com/id/Make-a-Self-balancing-Robot-with-Arduino-UNO>

- Tính năng:
 - Tự động cân bằng, khi có hoặc không có lực tác động.
 - Có thể điều khiển thông qua ứng dụng điện thoại.

2.2. Nguyên lí hoạt động.

- Nguyên lí cân bằng.
- Phương thức truyền nhận dữ liệu không dây.

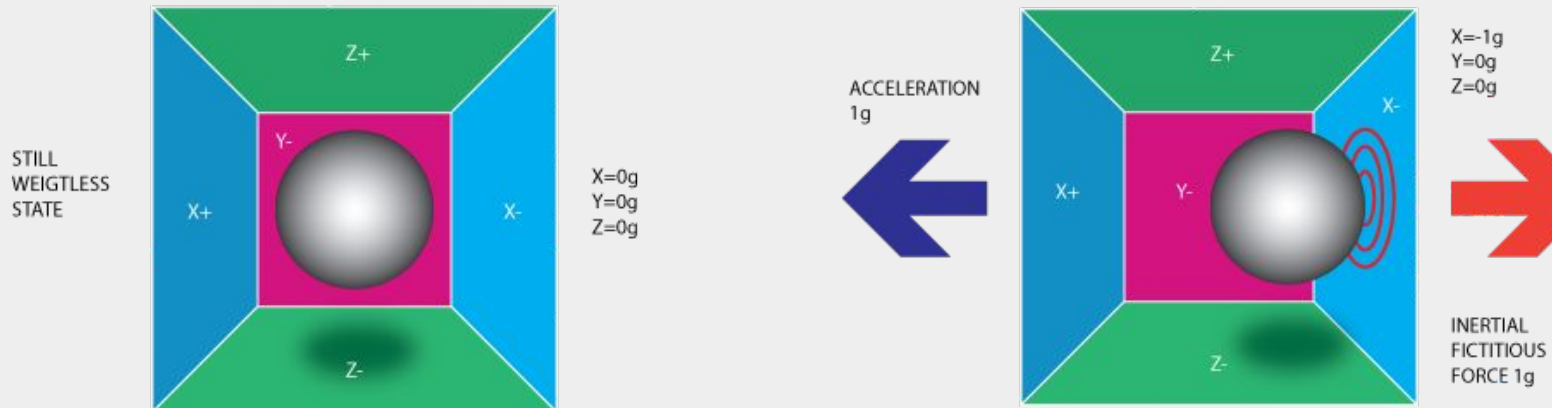
2.2.1. Nguyên lí hoạt động.

Acelerometer và gyroscope

- Xác định góc của xe trong không gian

- Acelerometer

- Đo gia tốc (mô tả sự thay đổi vận tốc theo thời gian)
- Hữu ích cho các ứng dụng xác định hướng

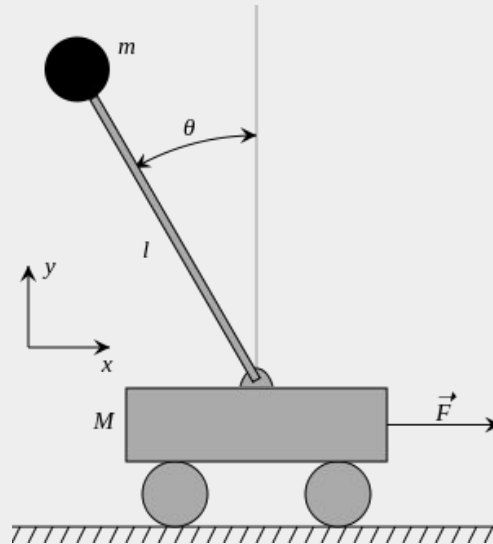


2.2.1. Nguyên lí hoạt động.

- Gyroscope
 - Đo tốc độ quay quanh của vật xung quanh trục
 - Độ chính xác cao, ít phát sinh nhiễu
 - Tốc độ đáp ứng khá chậm.
- Accelerometer kết quả đọc phát sinh nhiễu khá cao
- Accelerometer + Gyroscope giúp cải thiện độ chính xác

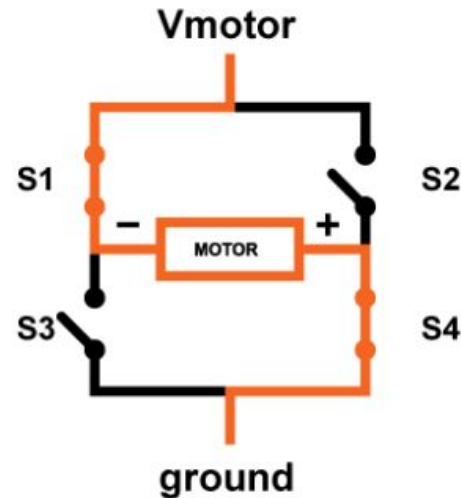
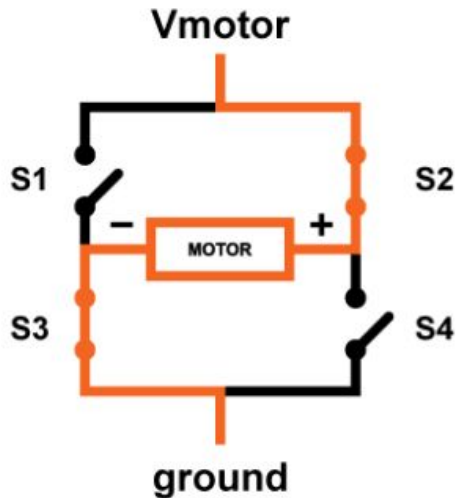
2.2.1. Nguyên lí hoạt động.

- Nguyên lí cân bằng
 - Nguyên lí con lắc ngược (inverted pendulum)



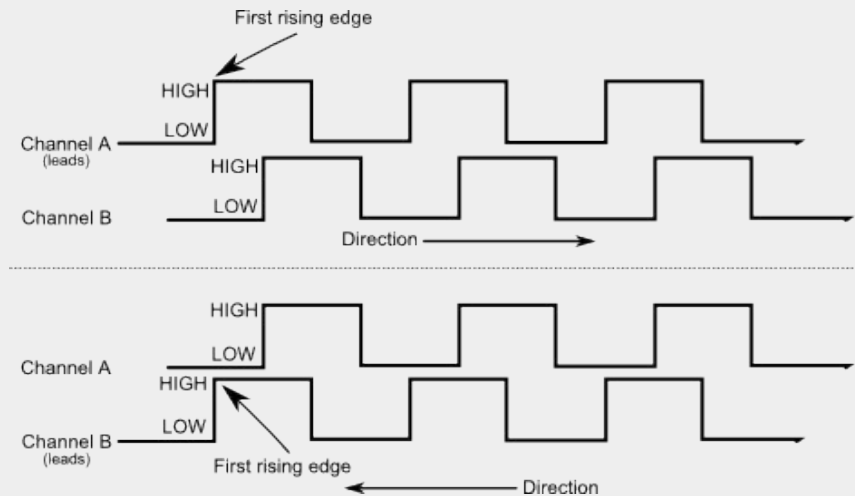
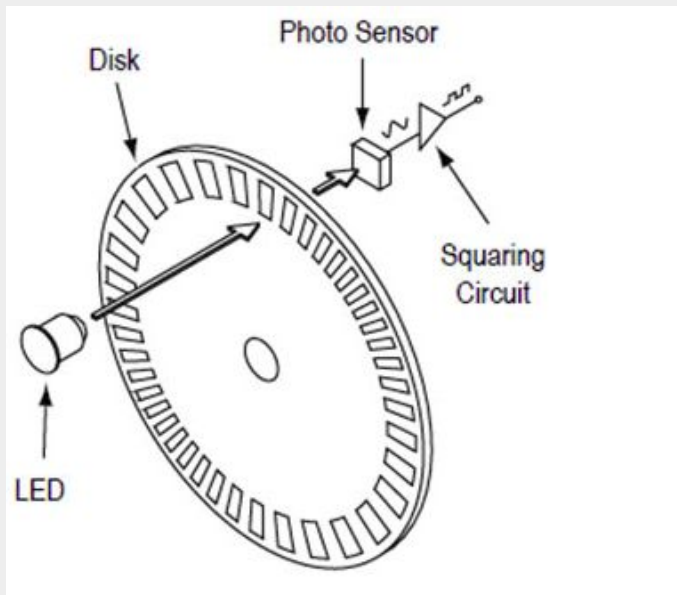
2.2.1. Nguyên lí hoạt động.

- Điều khiển động cơ



2.2.1. Nguyên lí hoạt động.

- Encoder



2.2.1. Nguyên lí hoạt động.

- Arduino và module Bluetooth



Nguồn ảnh: <http://microcontrollerslab.com>

3. Vật tư, linh kiện

- Bảng liệt kê vật tư, linh kiện của ứng dụng

STT	Tên linh kiện	Chức năng	Số lượng	Giá thành(vnd)
1	Khung xe (khung + 2 bánh xe)	Cố định các module	1	100.000
2	Động cơ kèm encoder	Giúp di chuyển, tính toán vận tốc của xe	2	160.000
3	Arduino Uno	Xử lý tín hiệu, điều khiển thiết bị	1	135.000
4	Mạch nguồn LM2596	Chuyển đổi điện áp từ pin sang 5V	1	25.000
5	Pin Lipo 12V-2200Mah	Nguồn cung cấp cho toàn bộ hệ thống	1	300.000
6	MPU6050	Xác định hướng của vật trong không gian	1	50.000
7	Mạch điều khiển động cơ TB612FNG	Điều khiển 2 động cơ	1	70.000
8	Connectivity shield	Kết nối các thiết bị	1	70.000
9	Module Bluetooth BT04	Truyền, nhận tín hiệu bluetooth	1	45.000
Tổng chi phí phần cứng				905.000

4. Quá trình thực hiện

- Chọn nền tảng phát triển
- Thiết kế khung xe
- Điều khiển động cơ
- Giao tiếp giữa Arduino Uno và Encoder
- Arduino Uno và MPU6050
- Xác định các thông số P.I.D
- Arduino Uno giao tiếp với module Bluetooth

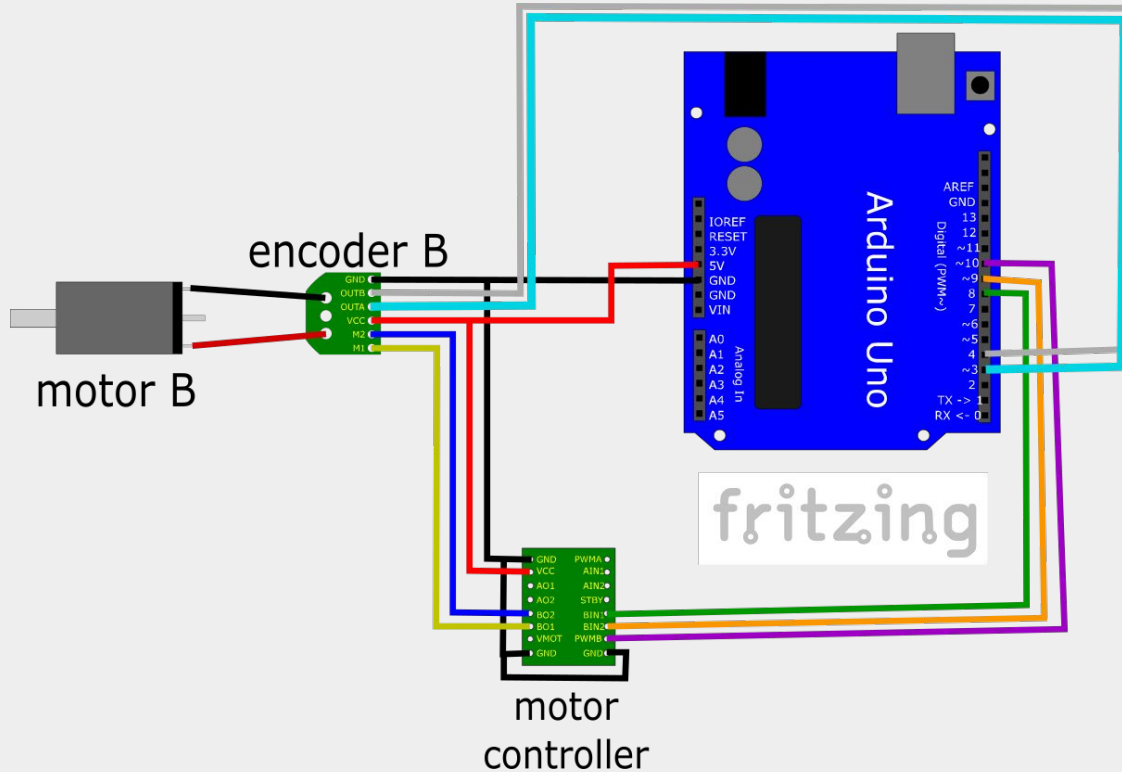
4.1. Chọn nền tảng phát triển

- Arduino
 - Dễ sử dụng, đơn giản với người mới bắt đầu
 - Nhiều thư viện hỗ trợ
 - Cộng đồng phát triển lớn

4.2. Thiết kế khung xe, điều khiển động cơ

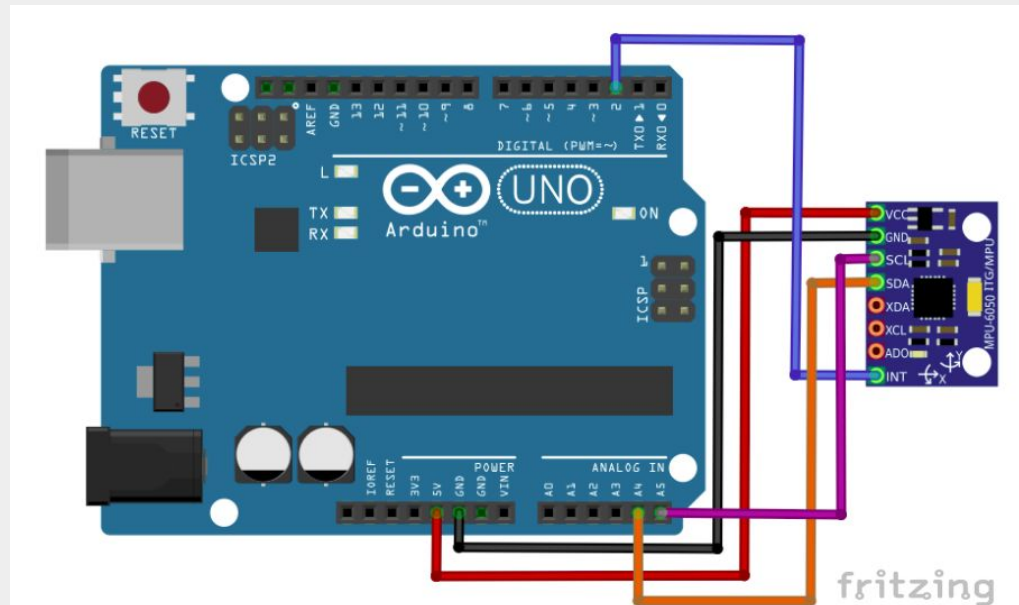
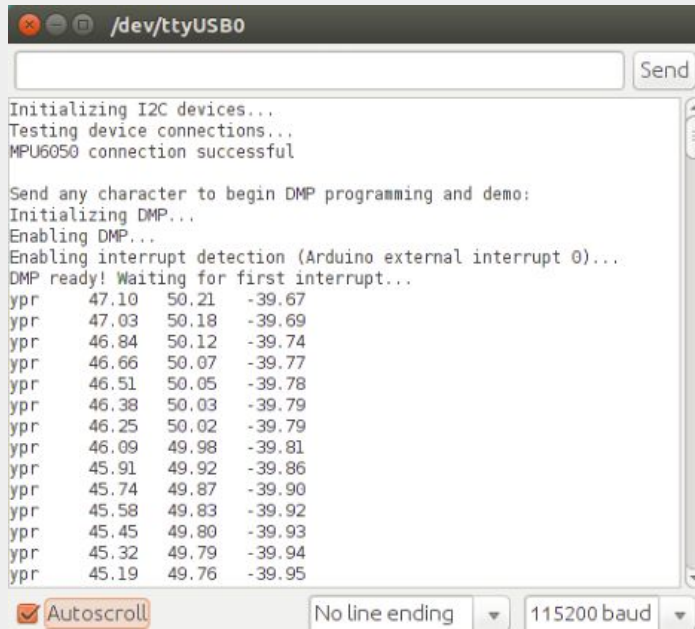
- Thiết kế khung xe
- Điều khiển động cơ

4.3. Giao tiếp giữa Arduino Uno và Encoder



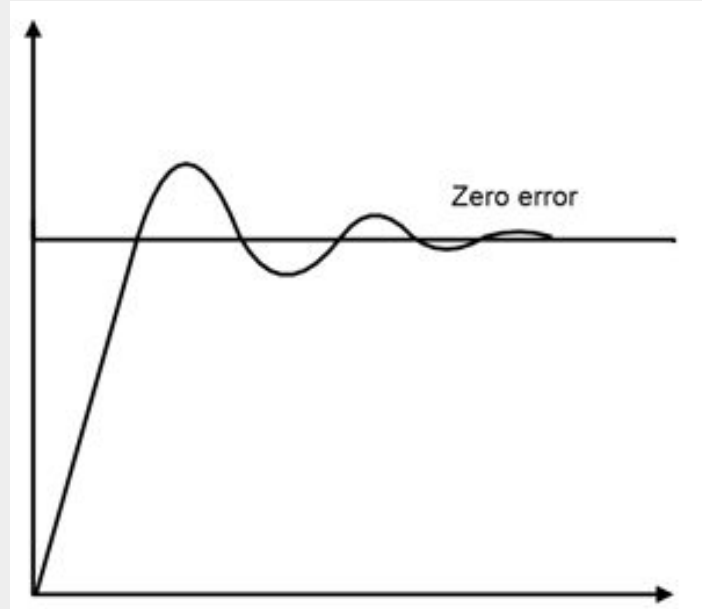
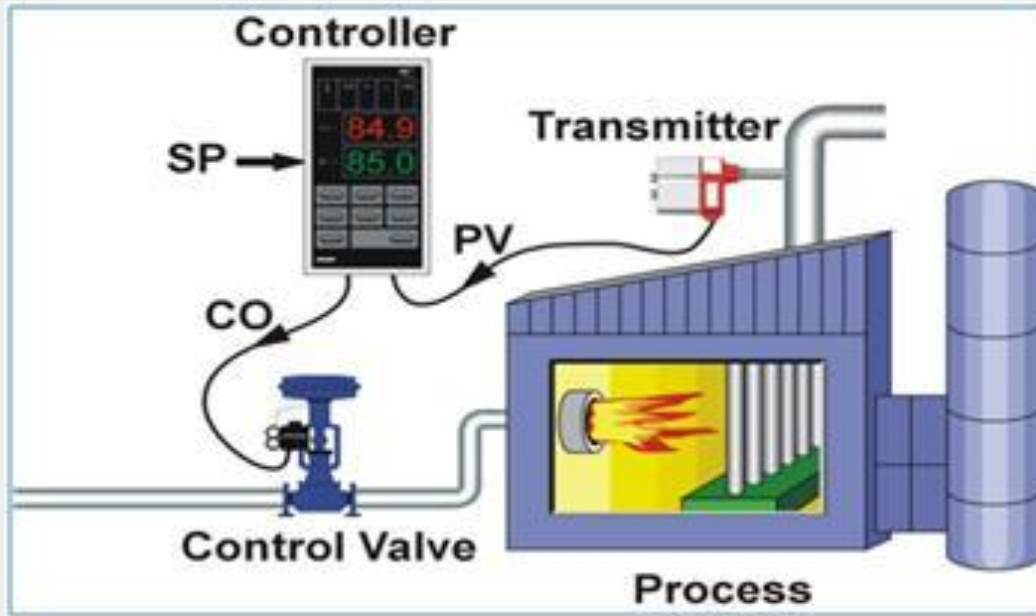
4.4. Arduino Uno và MPU6050

- Truyền nhận qua giao thức I2C
- Đọc giá trị accelerometer, gyroscope



4.5. Hệ thống PID controller

- Hệ thống điều khiển có phản hồi (feedback)
- Độ chính xác tăng dần theo thời gian



4.6. Xác định các thông số P.I.D (Proportional-Integral-Derivative)

- Desired state: Trạng thái mong muốn
- $e(t)$: Phần sai số

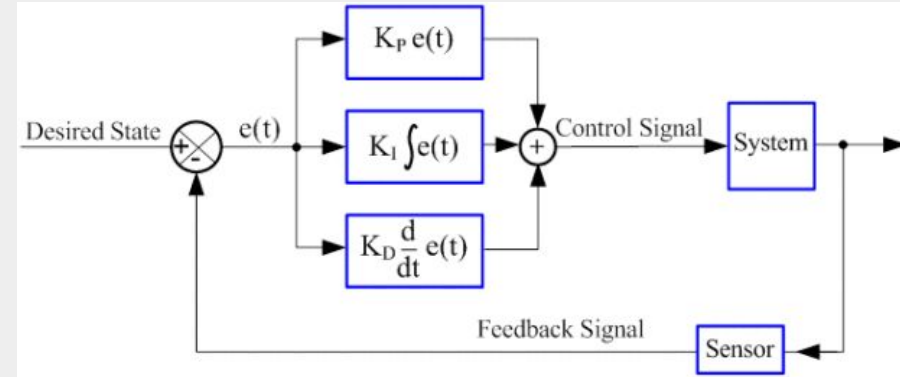
- $K_p \cdot e(t)$: Proportional part

- $K_i \cdot e(t)$: Integral part

- $K_d \cdot \frac{d}{dt} e(t)$: Derivative part

- \oplus : Các phần thêm vào nhau, tạo tín hiệu điều khiển -> Arduino Uno

- MPU6050: Gửi vị trí hiện tại của xe -> Arduino Uno -> tính toán -> feedback





SELF-BALANCING ROBOT WITH ARDUINO UNO



5. Chạy chương trình, thử nghiệm

Thank You