

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CƠ KHÍ ĐỘNG LỰC
BỘ MÔN Ô TÔ VÀ XE CHUYÊN DỤNG

ĐỀ TÀI KHOA HỌC
NGHIÊN CỨU BỘ ĐO VẬN TỐC THÂN XE

Giảng Viên hướng dẫn : Thầy Trần Thanh Tùng

Sinh viên thực hiện

Trần Văn Đức

Bùi Đức An

Nguyễn Tiến Hiệp

Hà nội tháng 3/2017

MỤC LỤC

Nghiên cứu chế tạo bộ đo vận tốc thân xe.....	3
I. Tổng quan.....	3
I.1 Đặt vấn đề.....	3
I.2 Mục tiêu nghiên cứu.....	3
I.3 Nội dung nghiên cứu.....	4
II. Cơ sở lý thuyết.....	4
II.1 Sự phanh và hệ thống phanh.	4
II.1.1 Sự phanh ô tô.....	4
II.1.2 Hệ thống phanh.....	5
II.2 Các chỉ tiêu đánh giá hiệu quả phanh.....	6
II.2.1 Lực phanh và lực phanh riêng.....	6
II.2.2 Gia tốc chậm dần.....	7
II.2.3 Thời gian phanh.....	7
II.2.4 Quãng đường phanh.....	8
II.3 Phương án thu thập dữ liệu.....	8
III. Thiết bị.....	11
III.1 Phần cứng.....	12
III.1.1 Encoder.....	12
III.1.2 Card arduino UNO R3.....	13
IV. Thí nghiệm.....	26
IV.1 Lắp đặt thiết bị.....	26
IV.2 Cài đặt các thông số ban đầu.....	28
IV.3 Quy trình thí nghiệm.....	28
V. Kết quả và đánh giá.....	29
V.1 Kết quả.....	29
V.2 Đánh giá.....	30

Nghiên cứu chế tạo bộ đo vận tốc thân xe

I. Tổng quan

I.1 Đặt vấn đề.

Hiện nay, giao thông đóng vai trò chủ đạo và ô tô chiếm tỷ trọng lớn trong vận chuyển nội địa. Vì vậy, lĩnh vực ô tô – xe chuyên dụng đã nhận được sự quan tâm của nhiều trường đại học cũng như nhiều cơ quan nghiên cứu có liên quan.

Trong những năm gần đây, ngành công nghiệp ô tô đã có những bước phát triển vượt bậc. Cùng với sự phát triển đó, vấn đề đảm bảo an toàn cho người và xe càng trở lên cần thiết. Trên ô tô hiện có rất nhiều những hệ thống đảm bảo an toàn, trong đó phanh là hệ thống an toàn chủ động hết sức quan trọng nên luôn được các nhà sản xuất ô tô quan tâm, không ngừng nghiên cứu hoàn thiện và nâng cao hiệu quả.

Vì vậy, việc nghiên cứu, kiểm định và thực hiện thí nghiệm về hệ thống phanh đang là vấn đề quan trọng nhưng hiện chưa có các công cụ để đo đạc và thu thập các thông số động lực học trong quá trình phanh ô tô, khiến cho các nghiên cứu trở nên khó khăn.

Với sự trợ giúp của các công nghệ hiện nay, chúng tôi đã lựa chọn đề tài: **“Nghiên cứu chế tạo bộ đo các thông số động lực học trong quá trình phanh ô tô”**.

I.2 Mục tiêu nghiên cứu.

Để đánh giá hệ thống phanh người ta sử dụng 4 chỉ tiêu về hiệu quả phanh (quãng đường phanh, thời gian phanh, gia tốc phanh và lực phanh). Vì vậy để phục vụ các công việc nghiên cứu và thí nghiệm trên ô tô trong quá trình phanh cần phải chế tạo được bộ đo các thông số trên.

I.3 Nội dung nghiên cứu

Thiết kế bộ bánh xe độc lập đo vận tốc ô tô, cụm gá lắp thiết bị trên ô tô.

Nghiên cứu lựa chọn và lắp đặt cảm biến, kết nối, hiển thị và xử lý tín hiệu cảm biến.

Điều khiển và kết nối máy tính, hiển thị kết quả, kiểm tra và phân tích các kết quả thí nghiệm.

II. Cơ sở lý thuyết.

II.1 Sự phanh và hệ thống phanh.

II.1.1 Sự phanh ô tô.

Xe lưu thông trên đường luôn gặp tình huống phải giảm vận tốc như cần phải dừng xe, gặp chướng ngại vật, gặp đường xấu.... Quá trình làm giảm vận tốc ô tô trong các trường hợp kể trên là quá trình phanh ô tô và được thực hiện bằng một hệ thống gọi là hệ thống phanh.

Khi xe chuyển từ v_1 đến v_2 ($v_2 < v_1$) xe thực hiện một chuyển động chậm dần, nghĩa là phải tạo cho xe một gia tốc j_p ngược chiều với chiều chuyển động nghĩa là phải tạo ra một lực, cản lại sự chuyển động của xe. Lực này được gọi là lực phanh.

Tuy nhiên trên ô tô ngoài khối lượng chuyển động tịnh tiến còn có các khối lượng chuyển động quay bao gồm bánh xe và các chi tiết liên quan đến động học bánh xe. Do vậy lực tác dụng vào xe để tạo ra gia tốc j_p phải là:

$$F = m j_p \delta$$

Trong đó δ là hệ số kể đến ảnh hưởng của các khối lượng quay khi phanh xe.

Xe có khối lượng m chạy với vận tốc v sẽ có động năng:

$$L = \frac{m \cdot v^2}{2}$$

Giả sử xe đang chạy với vận tốc v_1 và có nhu cầu giảm vận tốc xuống v_2 . Khi đó cần phải tiêu phí một năng lượng:

$$\Delta L = \frac{m \cdot (v_1^2 - v_2^2)}{2}$$

Như vậy quá trình giảm vận tốc của xe phải giải quyết song song hai vấn đề:

Tạo ra lực cản ngược chiều chuyển động của ô tô.

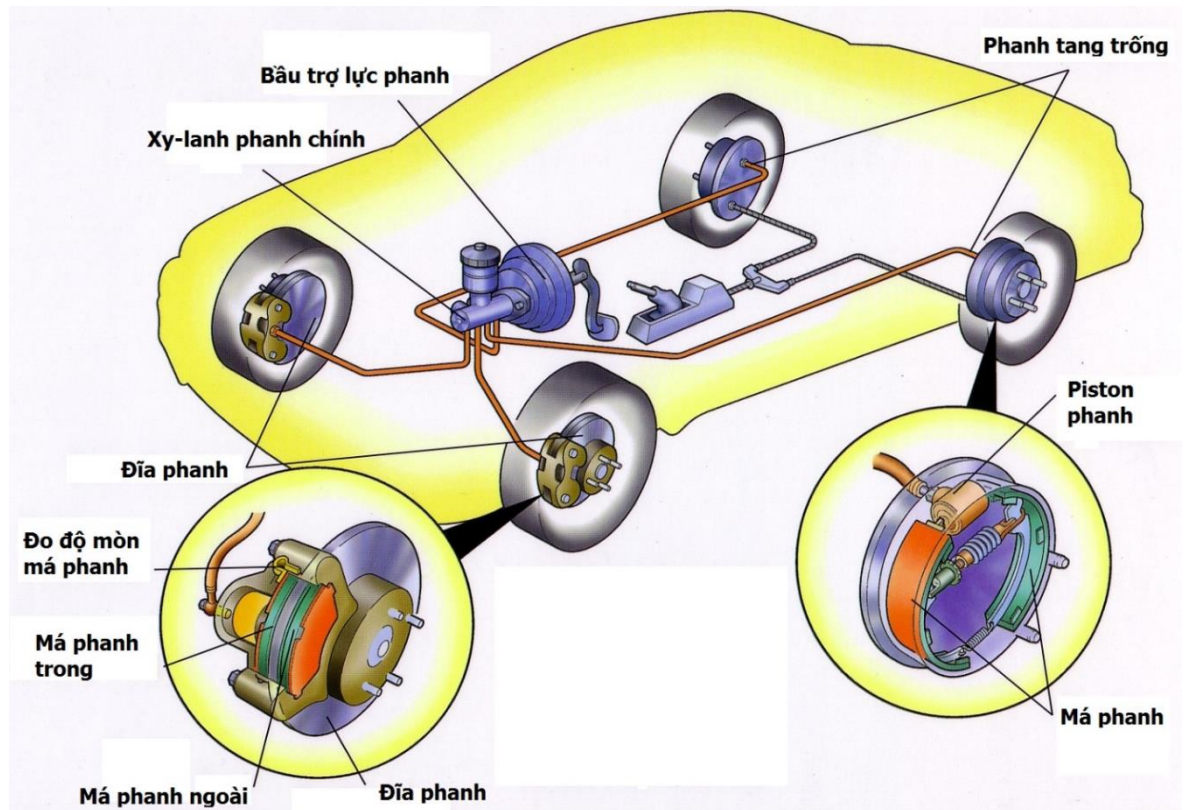
Tiêu tán đi một lượng động năng của xe.

II.1.2 Hệ thống phanh.

Trên ô tô người ta lắp một hệ thống để thực hiện quá trình phanh nói trên, hệ thống này được gọi là hệ thống phanh ô tô. Hệ thống phanh trên ô tô thường có 2 bộ phận chính:

- Bộ phận cơ cấu phanh: Bộ phận này có chức năng sinh ra mô men phanh tức mô men cản lại sự quay của bánh xe. Hiện nay trên ô tô mô men phanh được sinh ra nhờ ma sát. Cơ cấu phanh có một cặp chi tiết ma sát. Một chi tiết hay (hoặc một cụm chi tiết) gắn với bánh xe, quay cùng bánh xe; một chi tiết (hoặc cụm chi tiết) gắn với thân xe (thông qua cầu xe hay một bộ phận nào đó). Khi phanh người lái điều khiển cho cặp chi tiết ma sát này tiếp xúc với nhau, mô men ma sát do cặp chi tiết ma sát này tạo ra chính là mô men phanh. Cách này có thể tạo ra mô men phanh theo yêu cầu của người lái.

- Bộ phận dẫn động phanh: Bộ phận này truyền lực điều khiển từ người lái hoặc từ một nguồn năng lượng nào đó đến cơ cấu phanh để điều khiển cơ cấu phanh sinh ra mô men phanh.



Hình 1: Hệ thống phanh trên ô tô.

II.2 Các chỉ tiêu đánh giá hiệu quả phanh.

II.2.1 Lực phanh và lực phanh riêng.

Mỗi loại xe tùy theo trọng lượng, momen phanh yêu cầu được xác định và hệ thống phanh thiết kế phải đảm bảo sinh ra lực phanh yêu cầu.

Trị số lực phanh F_p chỉ có ý nghĩa đối với từng loại xe cụ thể. Xe có trọng lượng khác nhau đòi hỏi lực phanh khác nhau. Để đánh giá tổng quát hơn người ta đưa ra thông số lực phanh riêng p_p :

$$p_p = \frac{F_p}{G}$$

- G là trọng lượng đặt lên bánh xe phanh

• Tại các trạm đăng kiểm của Việt Nam, xe khiểm tra nếu $p = 0,5$ là đạt yêu cầu.

II.2.2 Gia tốc chậm dần.

Xét trường hợp phanh xe trên đường bằng không kéo mooc, bỏ qua lực cản lưn và lực cản không khí, ta có:

$$F_p = F_{p1} + F_{p2} = mj_p \delta$$

Khi phanh với lực cực đại gia tốc phanh cũng cực đại.

Gia tốc chậm dần của xe thay đổi trong quá trình phanh. Khi đo gia tốc chậm dần người ta quan tâm đến 2 giá trị:

Gia tốc chậm dần cực đại.

Gia tốc chậm dần trung bình.

II.2.3 Thời gian phanh.

Khi phanh gia tốc của ô tô là gia tốc chậm dần, vector gia tốc ngược chiều với chiều chuyển động của xe. Ta có:

$$j_p = -\frac{dv}{dt} = \frac{F_p}{m\delta} \rightarrow dt = -\frac{m\delta}{F_p} dv$$

Thời gian phanh từ khi xe có vận tốc v_1 đến khi xe có vận tốc v_2 là:

$$t = \int_{v_1}^{v_2} -\frac{m\delta}{F_p} dv = \int_{v_2}^{v_1} \frac{m\delta}{F_p} dv$$

Khi phanh với lực phanh cực đại sẽ có thời gian phanh cực tiểu:

$$t_{min} = \int_{v_2}^{v_1} \frac{m\delta}{F_{pmax}} dv = \frac{m\delta}{F_{pmax}} (v_1 - v_2)$$

Nếu phanh cho xe dừng lại ($v_2 = 0$):

$$t_{min} = \frac{m\delta}{F_{pmax}} v_1$$

II.2.4 Quãng đường phanh.

Ta có:

$$j_p = \frac{F_p}{m\delta} = -\frac{dv}{dt}$$

Nhân hai vế của biểu thức trên với ds ta có:

$$\frac{F_p}{m\delta} ds = -\frac{ds}{dt} dv = -v dv \rightarrow ds = -\frac{m\delta}{F_p} v dv$$

Quãng đường phanh từ khi xe có vận tốc v_1 đến khi xe có vận tốc v_2 là:

$$s = \int_{v_1}^{v_2} -\frac{m\delta}{F_p} v dv = \int_{v_2}^{v_1} \frac{m\delta}{F_p} v dv = \frac{m\delta}{2F_p} (v_1^2 - v_2^2)$$

Khi phanh với lực cực đại thì quãng đường phanh sẽ là cực tiểu:

$$s_{min} = \frac{m\delta}{2F_{pmax}} (v_1^2 - v_2^2)$$

Nếu phanh cho đến khi xe dừng lại ($v_2 = 0$):

$$s_{min} = \frac{m\delta}{2F_{pmax}} v_1^2$$

II.3 Phương án thu thập dữ liệu.

Ta nhận thấy rằng trong quá trình phanh, bánh xe sẽ bị trượt tức là sẽ không thể đo chính xác được thông số động học của xe thông qua các bánh xe. Vì vậy ta sẽ lắp thêm trên xe một bánh xe nữa gọi là bánh xe số 5. Bánh xe này do không chịu tác dụng của lực phanh nên sẽ không bị trượt và ta có thể đo được vận tốc chậm dần, gia tốc chậm dần thông qua bánh xe này.

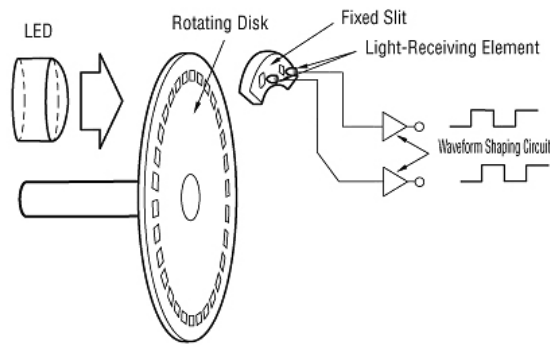


Hình 2. Bánh xe số 5 lắp trên xe Toyota HIACE.

Để có thể thu thập được các thông số qua bánh xe số 5, ta sử dụng một cảm biến vận tốc gọi là encoder. Encoder dạng chuyển động quay (shaft encoder) là thiết bị điện cơ dùng để biến đổi các vị trí góc của trục quay từ giá trị analog sang mã digital.

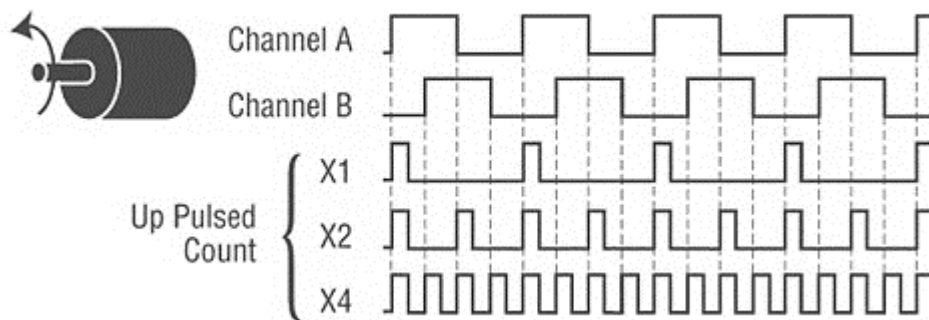
Encoder cấu tạo gồm hai thành phần:

- Một cặp diode thu phát tín hiệu quang học.
- Một đĩa tròn (chuyển động tròn) hoặc một thước thẳng (chuyển động thẳng) có đục lỗ hay sơn để tạo các điểm có khoảng cách đều, đặt ở giữa cặp diode thu phát.



Hình 3. Cấu tạo của encoder tròn.

Trục của encoder sẽ được ăn khớp với trục của bánh xe số 5, khi bánh xe số 5 quay sẽ làm đĩa encoder quay theo. Trên đĩa có các lỗ (rãnh). Người ta dùng một đèn led để chiếu lên mặt đĩa. Khi đĩa quay, chỗ không có lỗ (rãnh), đèn led không chiếu xuyên qua được, chỗ có lỗ (rãnh), đèn led sẽ chiếu xuyên qua. Khi đó, phía mặt bên kia của đĩa, người ta đặt một con mắt thu. Với các tín hiệu có hoặc không có ánh sáng chiếu qua, thiết bị tạo ra một chuỗi xung báo về bộ phận xử lý.



Hình 4. Xung và các kiểu đếm xung của encoder.

Tín hiệu từ encoder sẽ được gửi về một thiết bị thu thập và xử lý dữ liệu. thiết bị này sẽ đếm số xung của tín hiệu trong khoảng thời gian, rồi tính toán ra được vận tốc quay của encoder.

$$\omega_e = \frac{1000 \cdot C}{P \cdot X \cdot T}$$

Trong đó:

C là số xung trong khoảng thời gian T (ms)

P là số xung khi encoder quay đủ một vòng.

X có giá trị 1, 2, 4 tùy theo kiểu đếm xung X1, X2, X4.

Từ bánh xe số 5, ta sẽ biết được vận tốc chuyển động thẳng:

$$v_{xe} = v_{bx} = \omega_{bx} R_{bx} = u \omega_e R_{bx}$$

Trong đó:

u là tỷ số truyền vận tốc từ bánh xe số 5 đến trục của encoder.

ω_{bx} là vận tốc quay của bánh xe số 5.

R_{bx} là bán kính vòng lăn của bánh xe số 5.

Vì thời gian mỗi lần đếm xung của encoder rất nhỏ (cỡ mili giây) nên ta có thể coi trong khoảng thời gian đó, chuyển động của xe là thẳng biến đổi đều, ta có:

$$a = 1000 \frac{v_{xe1} - v_{xe2}}{T}$$

a là gia tốc của xe trong khoảng thời gian T (ms).

v_{xe1} là vận tốc đầu khoảng thời gian T.

v_{xe2} là vận tốc cuối khoảng thời gian T.

Trong trường hợp xe đang phanh: $a = -j_p$

Quãng đường chuyển động của xe trong khoảng thời gian T:

$$s = v_{xe2} T + \frac{a T^2}{2}$$

III. Thiết bị

III.1 Phần cứng.

III.1.1 Encoder

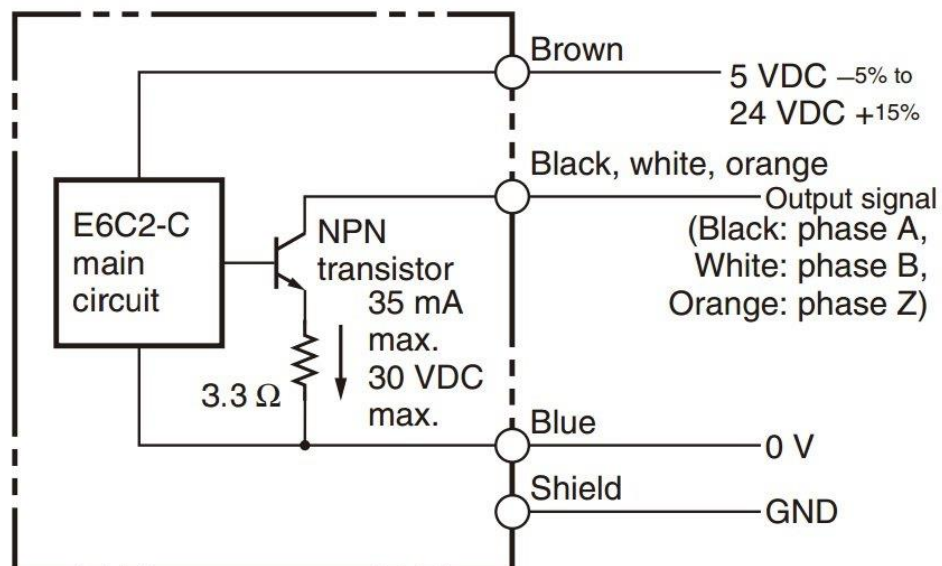
Sử dụng Encoder Omron E6C2-CWZ6C 1000P/R 2M.

Thông số kỹ thuật :

- + Nguồn điện: 5-24 VDC.
- + Ngõ ra: NPN.
- + Các pha ngõ ra: A, B, Z.
- + Tần số tối đa 100kHz.
- + Tốc độ tối đa: 6000 vòng/phút.



Hình 6. Encoder.



Hình 7. Sơ đồ nguyên lý.

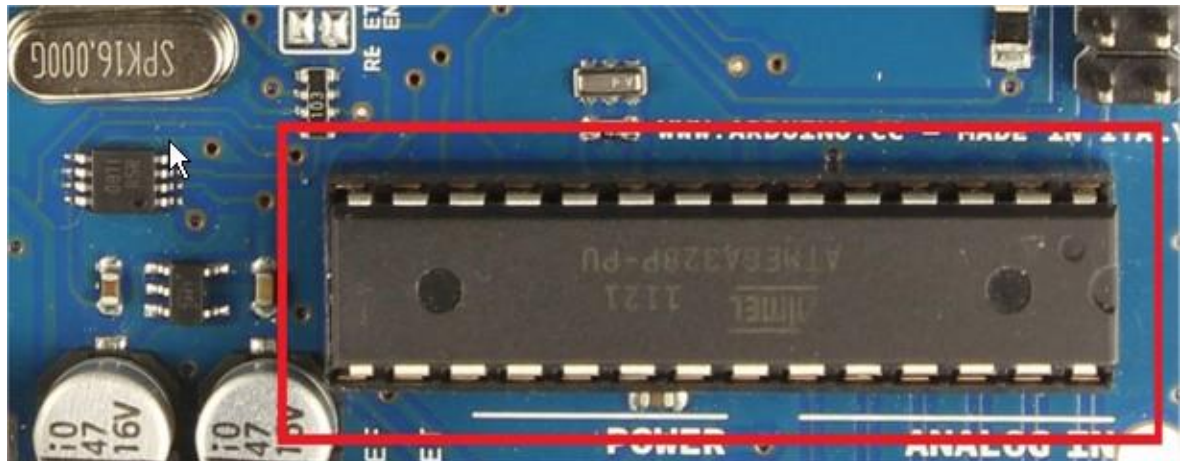
III.1.2 Card arduino UNO R3

Một vài thông số của Arduino UNO R3

Vi điều khiển	ATmega328 họ 8 bit
Điện áp hoạt động	5V DC (chỉ được cấp qua cổng USB)
Tần số hoạt động	16 MHz
Dòng tiêu thụ	Khoảng 30mA
Điện áp vào khyên dùng	7-12 V DC
Điện áp vào giới hạn	6-20 V DC
Số chân Digital I/O	14 (6 chân hardware PWM)
Số chân Analog	6 (độ phân giải 10 bit)
Dòng tối đa trên mỗi chân I/O	30 mA
Dòng ra tối đa (5V)	500 mA
Dòng ra tối đa (3.3V)	50mA

Bộ nhớ flash	32 KB (ATmega328) với 0.5KB dùng bởi bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)

Vi điều khiển



Arduino UNO có thể sử dụng 3 vi điều khiển họ 8 bit AVR là ATmega8, ATmega168, ATmega328. Bộ não này có thể xử lý những tác vụ đơn giản như điều khiển đèn LED nhấp nháy, xử lý tín hiệu cho xe điều khiển từ xa, làmfg một trạm đo nhiệt độ- độ ẩm và hiển thị lên màn hình LCD...

Năng lượng

Arduino UNO có thể được cấp nguồn 5V thông qua cổng USB hoặc cấp nguồn ngoài với điện áp khuyến dùng là 7-12V DC và giới hạn là 6-20V. Thường thì cấp nguồn bằng pin vuông 9v là hợp lý nếu không có nguồn cấp sẵn từ cổng USB. Nếu cấp nguồn quá giới hạn trên sẽ làm hỏng Arduino UNO.

Các chân năng lượng

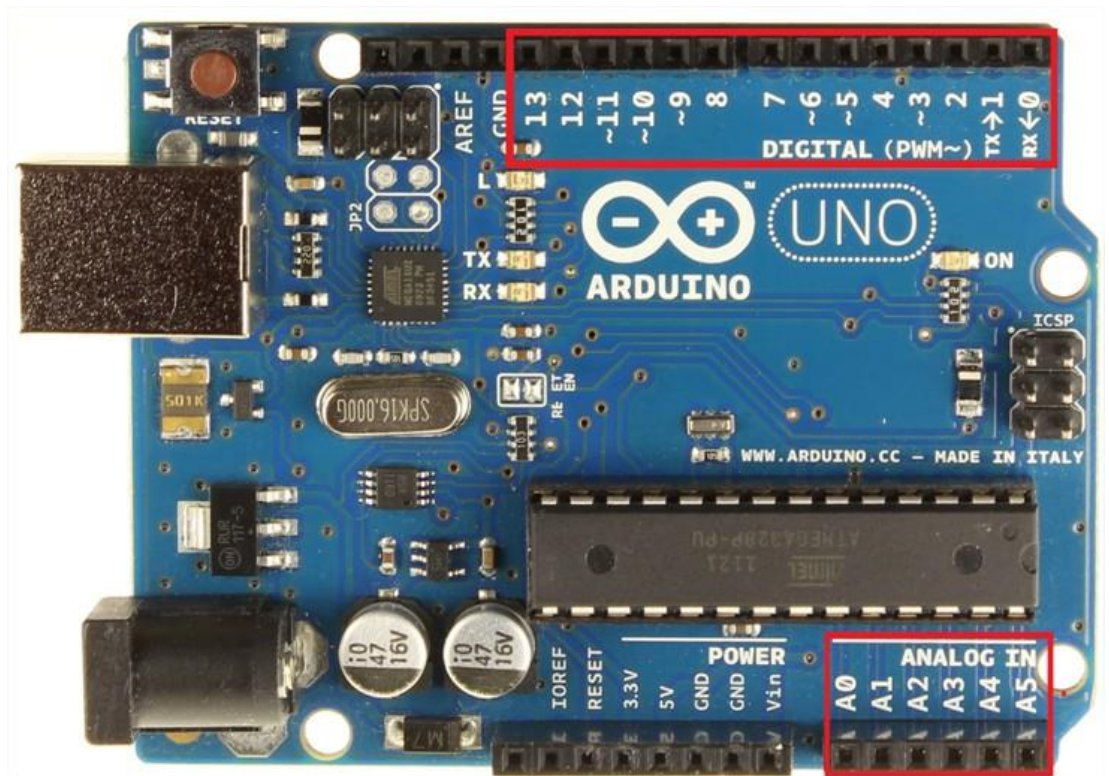
- **GND** (Ground): cực âm của nguồn điện cấp cho Arduino UNO. Khi bạn dùng các thiết bị sử dụng những nguồn riêng biệt thì những chân này phải được nối với nhau.
- **5V**: cấp điện áp đầu ra. Dòng tối đa cho phép ở chân này là 500mA.
- **3.3V**: cấp điện áp 3.3V đầu ra. Dòng tối đa cho phép ở chân này là 50mA.
- **Vin** (Voltage Input): Để cấp nguồn ngoài cho Arduino UNO, nối cực dương của nguồn với chân này và cực âm của nguồn nối với chân GND.
- **IOREF**: điện áp hoạt động của vi điều khiển trên Arduino UNO có thể được đo ở chân này. Điện áp hoạt động là 5V. Đây không phải là chân cấp nguồn.
- **RESET**: việc nhấn nút Reset trên board để reset vi điều khiển tương đương với việc chân RESET được nối với GND qua 1 điện trở 10KΩ.

Bộ nhớ

Vi điều khiển Atmega328 tiêu chuẩn cung cấp cho người dùng:

- **32KB bộ nhớ Flash**: những đoạn leecnhjlaapj trình sẽ được lưu trữ trong bộ nhớ Flash của vi điều khiển.
- **2KB cho SRAM** (Static Random Accessn Memory): giá trị các biến khai báo khi lập trình sẽ lưu ở đây.
- **1KB cho EEPROM** (Electrically Eraseble Programmable Read Only Memory): nơi đọc và ghi dữ liệu.

Các cổng vào/ra



Arduino UNO có 14 chân digital dùng để đọc hoặc xuất tín hiệu. Chúng chỉ có 2 mức điện áp là 0V và 5V với dòng vào/ra tối đa trên mỗi chân là 40mA. Ở mỗi chân đều có các điện trở pull-up từ được cài đặt ngay trong vi điều khiển ATmega328 (mặc định thì các điện trở này không được kết nối).

Một số chân digital có các chức năng đặc biệt như sau:

- **2 chân Serial:** 0 (RX) và 1 (TX): dùng để gửi (transmit – TX) và nhận (receive – RX) dữ liệu TTL Serial. Arduino Uno có thể giao tiếp với thiết bị khác thông qua 2 chân này.
- **Chân PWM (~):** 3, 5, 6, 9, 10, và 11: cho phép bạn xuất ra xung PWM với độ phân giải 8bit (giá trị từ 0 → 255 tương ứng với 0V → 5V) bằng hàm `analogWrite()`.
- **Chân giao tiếp SPI:** 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ngoài các chức năng thông thường, 4 chân này còn dùng để truyền phát dữ liệu bằng giao thức SPI với các thiết bị khác.

- **ED 13:** trên Arduino UNO có 1 đèn led màu cam (kí hiệu chữ L). Khi bấm nút Reset, đèn này nhấp nháy để báo hiệu. Nó được nối với chân số 13. Khi chân này được người dùng sử dụng, LED sẽ sáng.

Arduino UNO có 6 chân analog (A0 → A5) cung cấp độ phân giải tín hiệu 10bit (0 → 2¹⁰-1) để đọc giá trị điện áp trong khoảng 0V → 5V.

Lập trình cho Arduino

```
/*
```

```
-----
```

-

- Pin A của Encoder nối với pin D2 của Arduino
- Pin B của Encoder không cần nối
- Pin nguồn của Encoder nối với pin 5V của Arduino
- Pin đất của Encoder nối với pin GND của Arduino

```
*/
```

```
// Khai báo hằng số
```

```
#define encoderPinA 2 // Khai báo chân D2
```

```
#define encoderPinB 3 // Khai báo chân D3
```

```
#define radius 242.55 // Khai báo bán kính bánh xe (mm)
```

```
#define pi 3.14159 // Khai báo số Pi
```

```
#define pulsesPerTurn 100 // Khai báo thông số xung/giây của Encoder
```

```
#define timeBetweenReadings 10 // Khai báo khoảng thời gian lấy mẫu  
(ms), tương ứng 1s lấy 10 mẫu ~ 10 Hz
```

```
// Khai báo biến số
```

```
int state = 0;
```

```
volatile int countold, count; // Biến đếm đầu và cuối trong một khoảng  
thời gian
```

```
unsigned long timeBegin = 0, timeNow = 0; // Biến thời điểm đầu trong  
một khoảng thời gian
```

```
float vehicle; // Biến vận tốc đầu và cuối trong một khoảng thời gian
```

```

// Thiết lập thông số khởi động
void setup()
{
    count = 0;
    countold = 0;
    timeBegin = 0;
    timeNow = 0;
    vehicle = 0;

    pinMode(encoderPinA, INPUT_PULLUP); // Thiết lập chế độ Pullup
    cho encoderPin D2
    pinMode(encoderPinB, INPUT_PULLUP); // Thiết lập chế độ Pullup
    cho encoderPin D3
    attachInterrupt(0, sigA, CHANGE); // Gọi hàm up khi ngắt, số 0
    tương ứng Pin D2, số 1 tương ứng Pin D3, kiểu kích hoạt ngắt là CHANGE
    attachInterrupt(1, sigB, CHANGE); // Gọi hàm down khi ngắt
    Serial.begin(115200); // Thiết lập baudrate cho Serial
}

// Hàm chính của chương trình
void loop()
{
    // Điều khiển Arduino qua giá trị của biến state
    if(Serial.available())
    {
        char temp = Serial.read();
        if(temp == '0')
            state = 0;
        if(temp == '1')
            state = 1;
        if(temp == '2')

```

```

    state = 2;
}

switch(state)
{
    // state = 0: dừng Arduino
    case 0:
        break;

    // state = 1: bắt đầu đo
    case 1:
        // Xác định khởi gian lấy mẫu
        timeBegin = millis();
        while (millis() - timeBegin < timeBetweenReadings)
        {
            // Tắt ngắt
            detachInterrupt(0);
            detachInterrupt(1);

            // Vì thời gian lấy mẫu nhỏ nên có thể coi chuyển động là tuyến
            // tính trong khoảng thời gian đó
            vehicle = (count-countold)*radius*pi/(2*pulsesPerTurn*(millis()-
            timeBegin)); // Tính vận tốc

            // Gọi hàm ngắt
            attachInterrupt(0, sigA, CHANGE);
            attachInterrupt(1, sigB, CHANGE);
        }

        // Gán thời điểm cuối cùng của khoảng thời gian hiện tại vào thời
        // điểm cuối cùng của khoảng thời gian kế tiếp, tương tự với các biến vận tốc
        // và gia tốc

```

```

timeNow = timeNow + millis() - timeBegin;
countold = count;

// Hiện thị thời gian và vận tốc qua Serial
Serial.print(timeNow);
Serial.print("|");
Serial.println(velocity);
break;

// state = 2: Reset dữ liệu và thời gian về 0
case 2:
    count = 0;
    countold = 0;
    timeBegin = 0;
    timeNow = 0;
    vehicle = 0;
    break;
}
}

// Hàm up, tăng một giá trị cho biến đếm nếu xung tăng
void sigA()
{
    if (digitalRead(encoderPinA) == HIGH)
    {
        if (digitalRead(encoderPinB) == LOW)
        {
            count++;
        }
    }
    else
    {
        count--;
    }
}

```

```
    }  
  }  
  else  
  {  
    if (digitalRead(encoderPinB) == HIGH)  
    {  
      count++;  
    }  
    else  
    {  
      count--;  
    }  
  }  
}
```

// Hàm down, giảm một giá trị cho biến đếm nếu xung giảm

```
void sigB()  
{  
  if (digitalRead(encoderPinB) == HIGH)  
  {  
    if (digitalRead(encoderPinA) == HIGH)  
    {  
      count++;  
    }  
    else  
    {  
      count--;  
    }  
  }  
  else  
  {  
    if (digitalRead(encoderPinA) == LOW)
```

```
    {  
        count++;  
    }  
else  
    {  
        count--;  
    }  
}  
}
```

```
quadrature_encoder | Arduino 1.8.1  
File Edit Sketch Tools Help  
quadrature_encoder  
- Pin B của Encoder không cần nối  
- Pin nguồn của Encoder nối với pin 5V của Arduino  
- Pin đất của Encoder nối với pin GND của Arduino  
*/  
  
// Khai báo hằng số  
#define encoderPinA 2 // Khai báo chân D2  
#define encoderPinB 3 // Khai báo chân D3  
#define radius 242.55 // Khai báo bán kính bánh xe (mm)  
#define pi 3.14159 // Khai báo số Pi  
#define pulsesPerTurn 100 // Khai báo thông số xung/giây của Encoder  
#define timeBetweenReadings 10 // Khai báo khoảng thời gian lấy mẫu (ms), tương ứng là lấy 10 mẫu - 10 Hz  
  
// Khai báo biến số  
int state = 0;  
volatile int countOld, count; // Biến đếm đầu và cuối trong một khoảng thời gian  
unsigned long timeBegin = 0, timeNow = 0; // Biến thời điểm đầu trong một khoảng thời gian  
float vehicle; // Biến vận tốc đầu và cuối trong một khoảng thời gian  
  
// Thiết lập thông số khởi động  
void setup()  
{  
    count = 0;  
    countOld = 0;  
    timeBegin = 0;  
    timeNow = 0;  
    vehicle = 0;  
  
    pinMode(encoderPinA, INPUT_PULLUP); // Thiết lập chế độ Pullup cho encoderPin D2  
    pinMode(encoderPinB, INPUT_PULLUP); // Thiết lập chế độ Pullup cho encoderPin D3  
    attachInterrupt(0, sigA, CHANGE); // Gọi hàm up khi ngắt, số 0 tương ứng Pin D2, số 1 tương ứng Pin D3, kiểu kích hoạt ngắt là CHANGE  
    attachInterrupt(1, sigB, CHANGE); // Gọi hàm down khi ngắt  
    Serial.begin(115200); // Thiết lập baudrate cho Serial  
}
```

```
quadrature_encoder | Arduino 1.8.1  
File Edit Sketch Tools Help  
quadrature_encoder  
  
pinMode(encoderPinA, INPUT_PULLUP); // Thiết lập chế độ Pullup cho encoderPin D2  
pinMode(encoderPinB, INPUT_PULLUP); // Thiết lập chế độ Pullup cho encoderPin D3  
attachInterrupt(0, sigA, CHANGE); // Gọi hàm up khi ngắt, số 0 tương ứng Pin D2, số 1 tương ứng Pin D3, kiểu kích hoạt ngắt là CHANGE  
attachInterrupt(1, sigB, CHANGE); // Gọi hàm down khi ngắt  
Serial.begin(115200); // Thiết lập baudrate cho Serial  
}  
  
// Hàm chính của chương trình  
void loop()  
{  
    // Điều khiển Arduino qua giá trị của biến state  
    if(Serial.available())  
    {  
        char temp = Serial.read();  
        if(temp == '0')  
            state = 0;  
        if(temp == '1')  
            state = 1;  
        if(temp == '2')  
            state = 2;  
    }  
  
    switch(state)  
    {  
        // state = 0: dừng Arduino  
        case 0:  
            break;  
    }  
}
```

Tín hiệu góc quay của encoder được gửi về encoder kết hợp với việc đếm thời gian ta tính ra được vận tốc quay của bánh xe, lấy tín hiệu góc quay kết hợp với bán kính bánh xe ta tính ra được vận tốc dài của xe. Encoder tính toán xử lý xong dữ liệu sẽ gửi tín hiệu qua cổng monitor được kết nối với một chương trình giao diện được viết bằng Visual Studio

Visual Studio

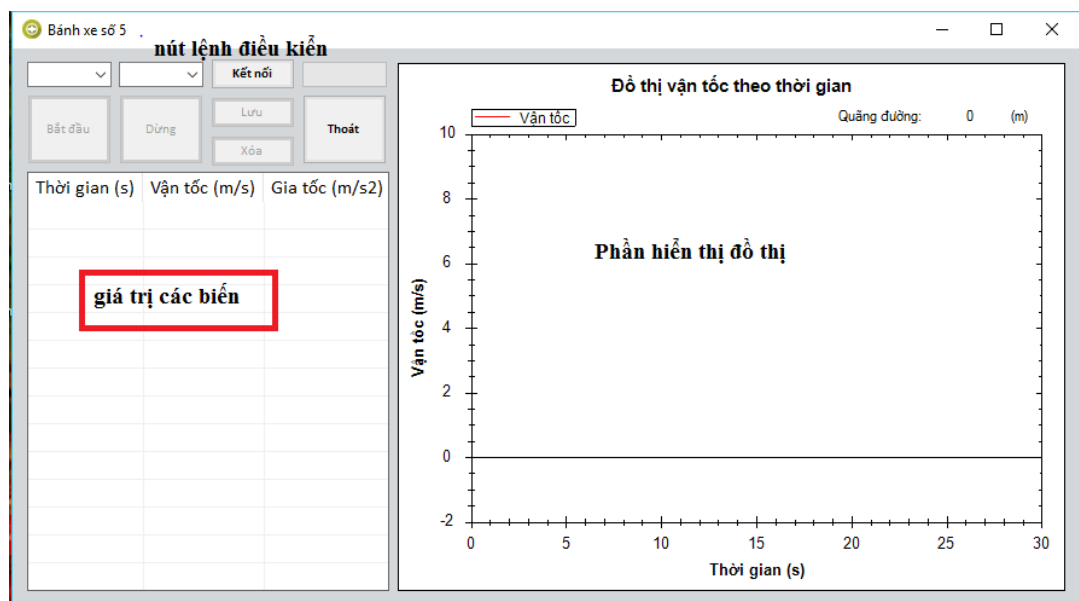
Microsoft Visual Studio là một môi trường phát triển tích hợp (IDE) từ Microsoft. Nó được sử dụng để phát triển chương trình máy tính cho Microsoft Windows, cũng như các trang web, các ứng dụng web và các dịch vụ web. Visual Studio sử dụng nền tảng phát triển phần mềm của Microsoft như Windows API, Windows Forms, Windows Presentation Foundation, Windows Store và Microsoft Silverlight. Nó có thể sản xuất cả hai ngôn ngữ máy và mã số quản lý.

Visual Studio bao gồm một trình soạn thảo mã hỗ trợ IntelliSense cũng như cải tiến mã nguồn. Trình gỡ lỗi tích hợp hoạt động cả về trình gỡ lỗi mức độ mã nguồn và gỡ lỗi mức độ máy. Công cụ tích hợp khác bao gồm một mẫu thiết kế các hình thức xây dựng giao diện ứng dụng, thiết kế web, thiết kế lớp và thiết kế giản đồ cơ sở dữ liệu. Nó chấp nhận các plug-in nâng cao các chức năng ở hầu hết các cấp bao gồm thêm hỗ trợ cho các hệ thống quản lý phiên bản (như Subversion) và bổ sung thêm bộ công cụ mới như biên tập và thiết kế trực quan cho các miền ngôn ngữ cụ thể hoặc bộ công cụ dành cho các khía cạnh khác trong quy trình phát triển phần mềm.

Visual Studio hỗ trợ nhiều ngôn ngữ lập trình khác nhau và cho phép trình biên tập mã và gỡ lỗi để hỗ trợ (mức độ khác nhau) hầu như mọi ngôn ngữ lập trình. Các ngôn ngữ tích hợp gồm có C,[4] C++ và C++/CLI (thông qua Visual C++), VB.NET (thông qua Visual Basic.NET), C# (thông

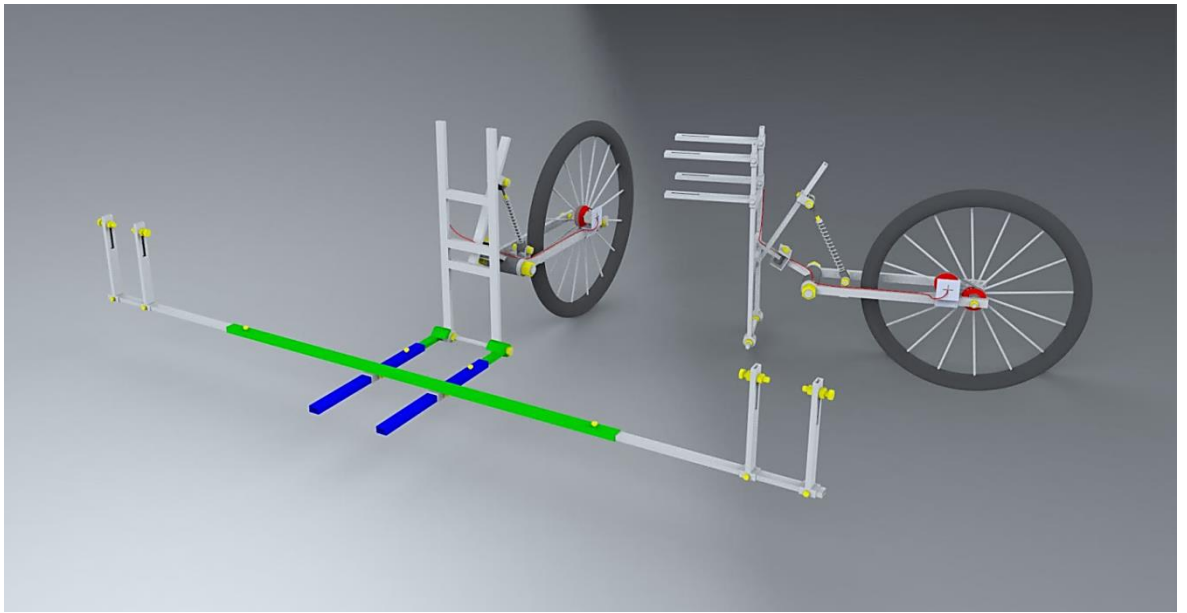
qua Visual C#) và F# (như của Visual Studio 2010[5]). Hỗ trợ cho các ngôn ngữ khác như J++/J#, Python và Ruby thông qua dịch vụ cài đặt riêng rẽ. Nó cũng hỗ trợ XML/XSLT, HTML/XHTML, JavaScript và CSS

Giao diện hiển thị dữ liệu

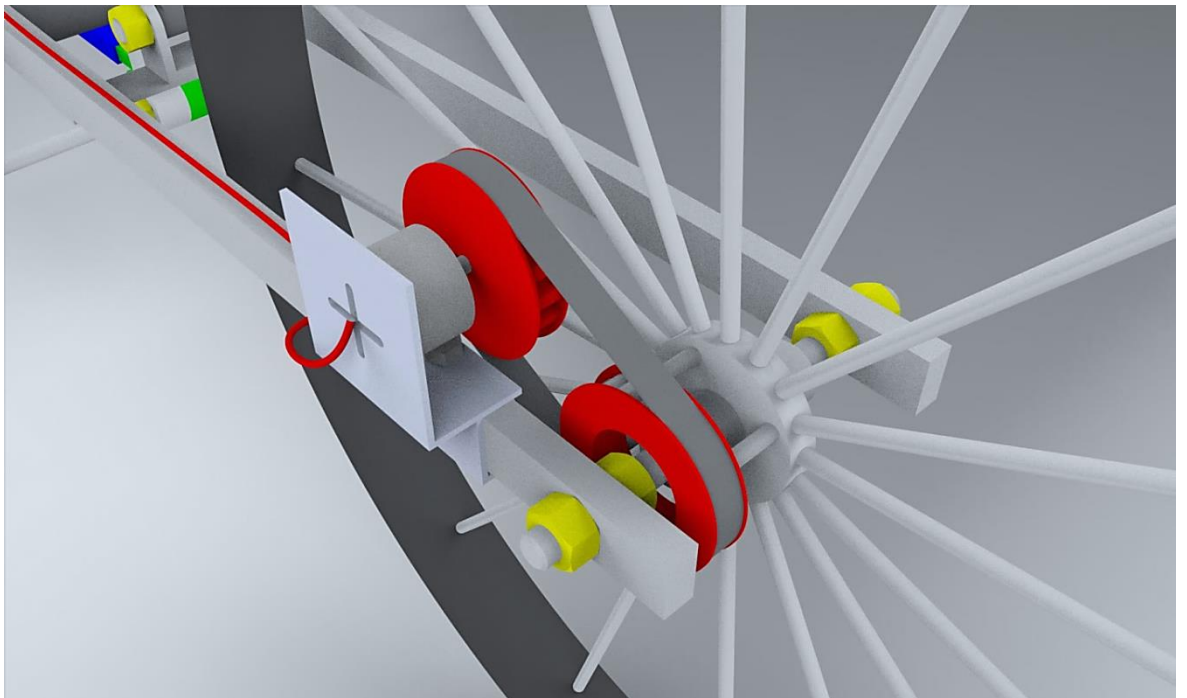


III.1.3 Bánh xe và khung cứng.

Để thuận lợi cho việc tháo lắp trên nhiều loại xe, ta sẽ sử dụng kết cấu dạng tháo lắp để chuyển đổi các thành phần. Bộ thiết bị có thể chuyển đổi thành hai dạng khác nhau, phù hợp lắp đặt trên hầu hết các loại xe đang lưu thông.

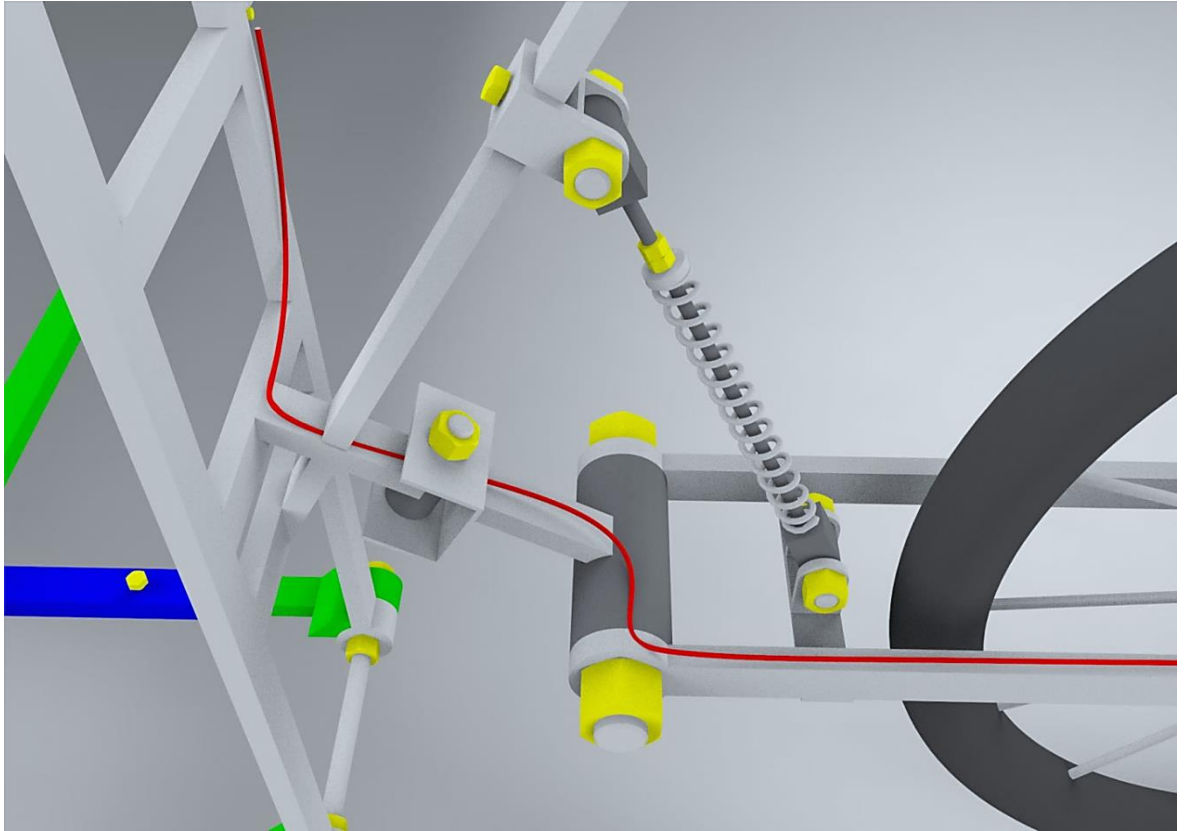


Hình 8. Bánh xe và khung cứng.



Hình 9. Encoder ăn khớp với bánh xe.

Encoder ăn khớp với bánh xe thông qua bộ truyền đai có tỷ số truyền $u = 1$.



Hình 10. Kết cấu lò xo thép

Để đảm bảo bánh xe luôn bám sát mặt đường, không xảy ra hiện tượng trượt lết, kết cấu khung có thêm một lò xo ép.

IV. Thí nghiệm

Xe thử nghiệm: xe máy

IV.1 Lắp đặt thiết bị

- Lắp đặt bộ khung lên xe

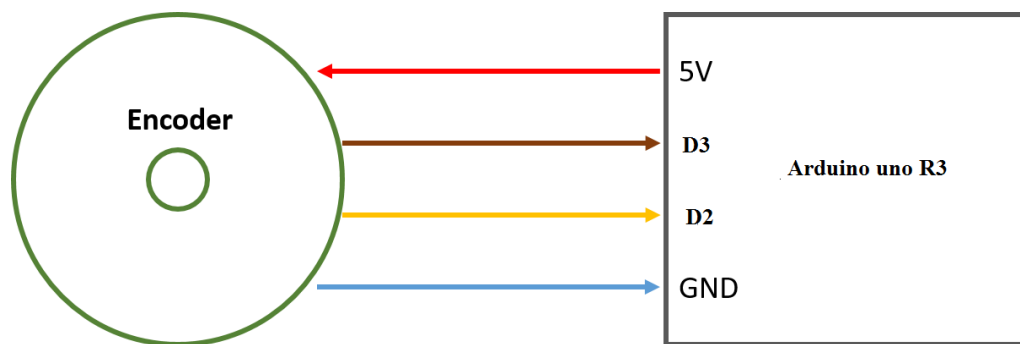


- Lắp đặt các cảm biến lên khung.





- Kết nối các dây cảm biến với bộ thu tín hiệu theo sơ đồ.



IV.2 Cài đặt các thông số ban đầu

Các thông số cần cài đặt ban đầu;

- Bán kính bánh xe số 5: $R = 242.87 \text{ mm}$
- Số xung trong một vòng quay của encoder: 100
- Thời gian lấy mẫu 100ms

IV.3 Quy trình thí nghiệm

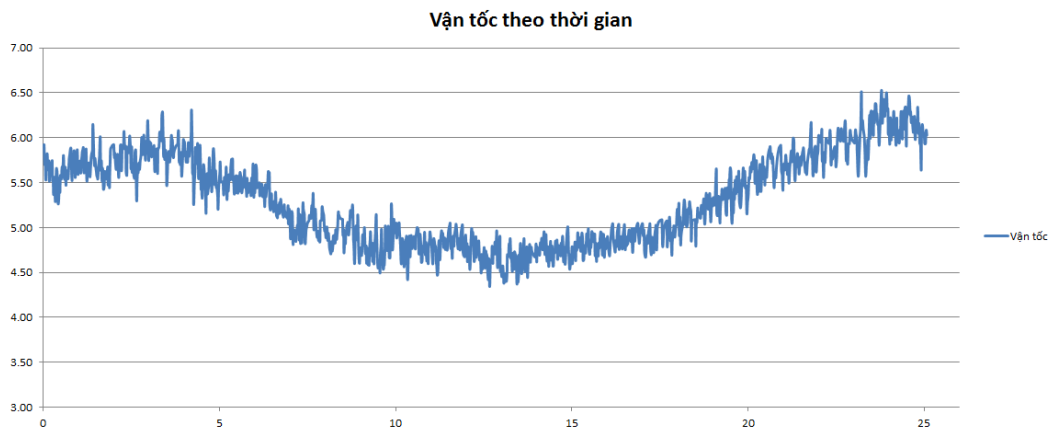
- Người ngồi trước cho xe chạy đều với vận tốc 20 km/h, người ngồi sau sử dụng máy tính để theo dõi giá trị vận tốc thu được khi xe chạy ổn định thì báo với người lái tiến hành phanh xe cho tới khi dừng hẳn

- tiến hành lưu lại dữ liệu

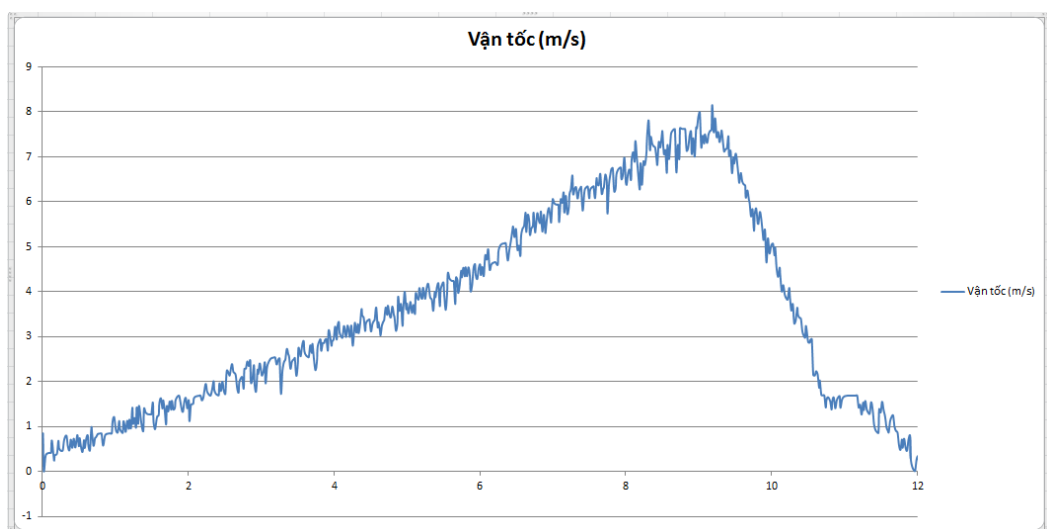
- tiến hành thí nghiệm 3 lần ở các dải vận tốc 20 km/h, 30 km/h, 40 km/h

V. Kết quả và đánh giá

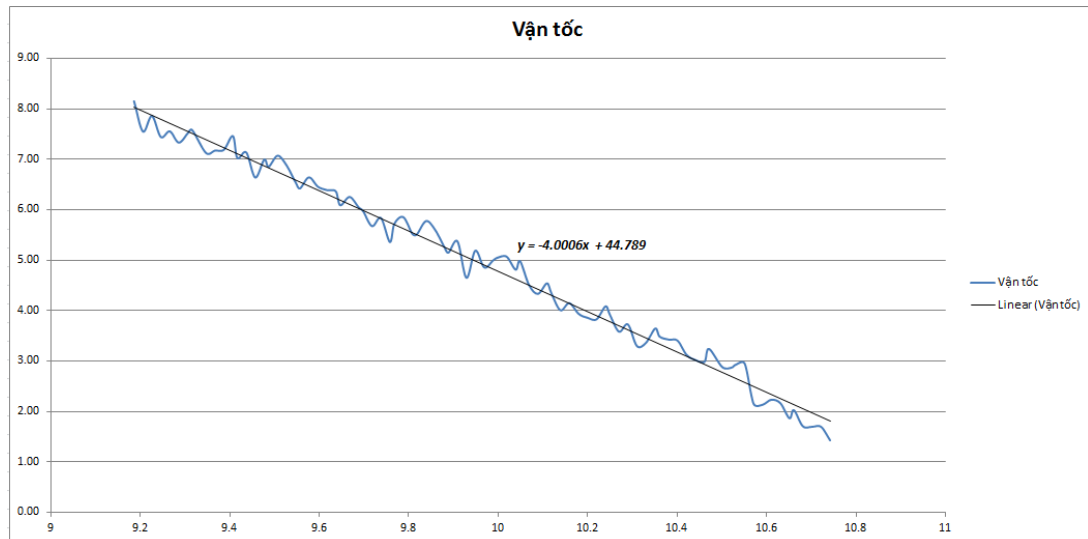
V.1 Kết quả



Chạy thử xe ở vận tốc quanh 20km/h



Phanh xe ở vận tốc 30km/h



Quá trình phanh xe

Theo đồ thị quá trình phanh xe ta nhận được gia tốc phanh của xe là -4 m/s^2 phù hợp với tiêu chuẩn gia tốc phanh của xe máy là lớn hơn 3.6 m/s^2 .

V.2 Đánh giá

Qua thời gian nghiên cứu, đề tài đã đạt được một số kết quả sau:

- Thiết kế và chế tạo thành công bộ bánh xe độc lập đo vận tốc ô tô, gia tốc ô tô và quãng đường đi được.
- Bộ thiết bị có thể lắp đặt trên nhiều loại xe khác nhau.
- Điều khiển và kết nối với máy tính, thu thập được dữ liệu để sử dụng cho các mục đích khác.
- Thí nghiệm thành công trên xe máy honda rsx