KING MONGKUT'S UNIVERSITY OF TECHNOLOGY LATKRABANG
FACULTY OF ENGINEERING
ENGINEERING GROUP OF ROBOTICS AND AI

01416304 – FEEDBACK CONTROL

PROJECT TITLE : INVERTED PENDULUM

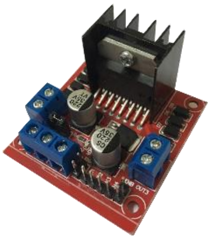
INSTRUCTED BY: DR. PITIKHATE SOORAKSA

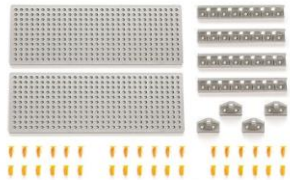
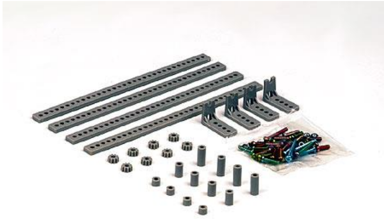
NAME : SURASAKUL, THANAPAT
TRI, WARAKORN, WEERAPHAT,
WORAPON, YODKWAN
STUDENT ID : 62011256, 62011272
62011285, 62011288, 62011293,
62011296, 62011297
DATE OF SUB. : 16/12/2020

TABLE OF CONTENTS

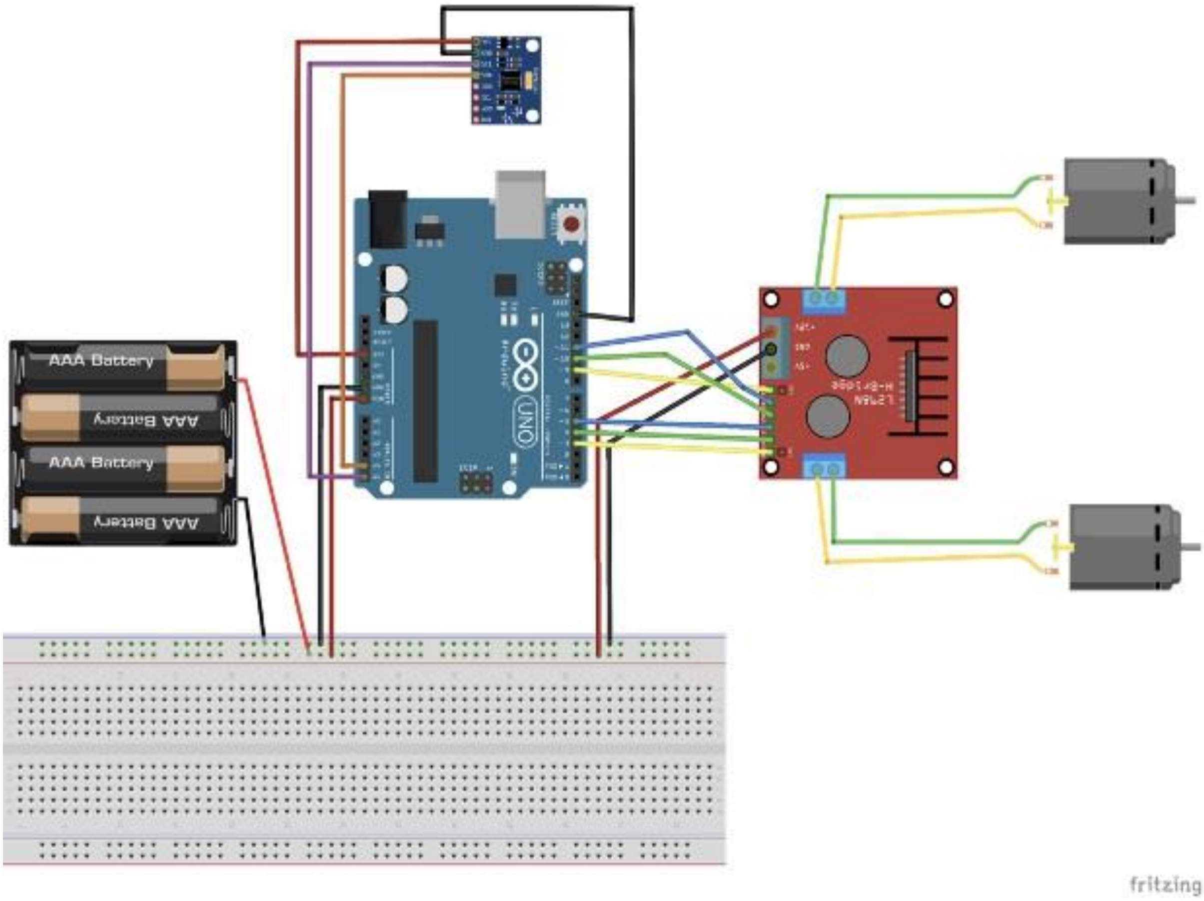
Physical Model

Equipment

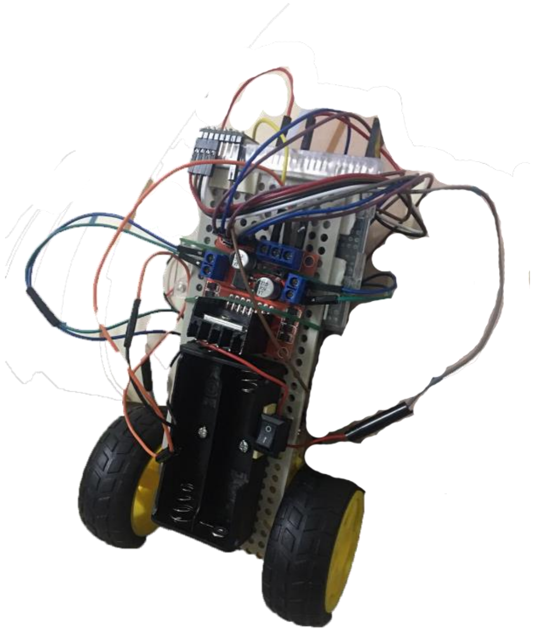
Gyroscope Sensor (GY-521)	
Motor Drive (L298N)	
Arduino Uno	
Slim Tire Set (55 mm Dia.)	
Double Gearbox	

Universal Plate Set	
Universal Arm Set	
Jumper Wires	
Battery Snap	
Battery (8×AA, 1×9V)	
4AA Battery holder (× 2)	

Circuit Diagram

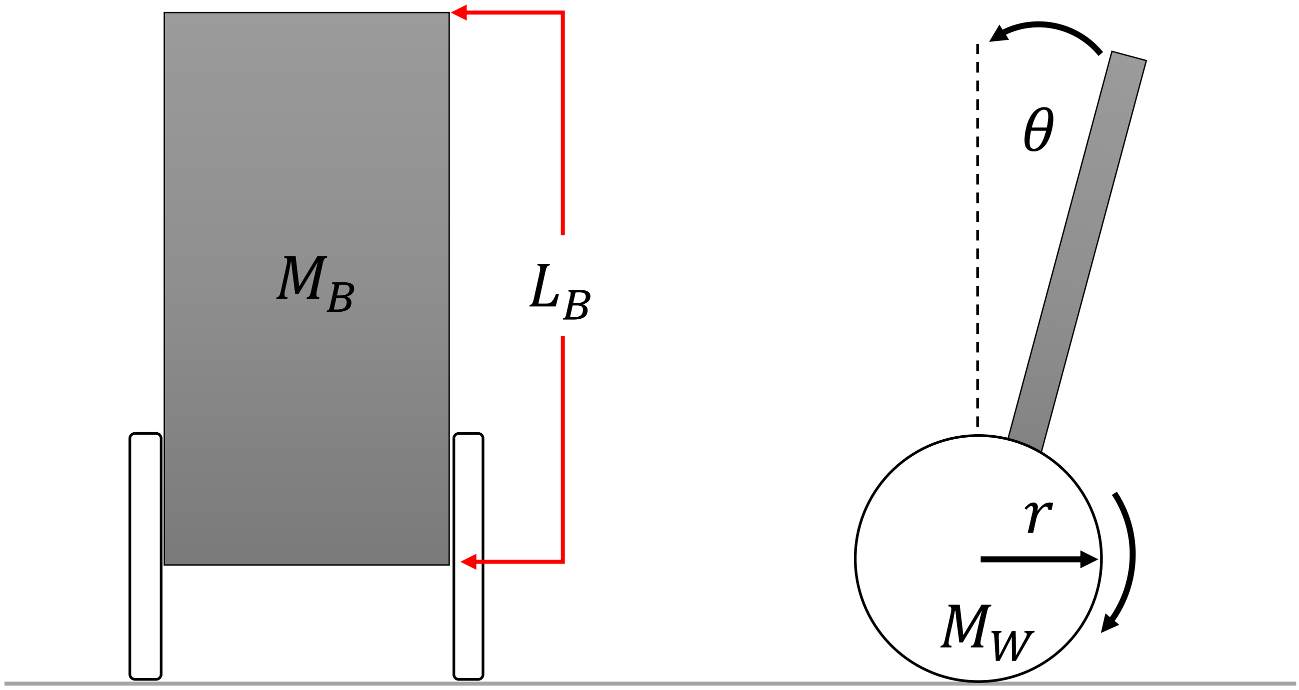


Physical Model



Mathematical Model

1. Free Body Diagram



$$I_B = \frac{1}{3} M_B L_B^2$$

$$I_W = 2\left(\frac{1}{2}\right) M_W r^2 = M_W r^2$$

Given : I_B is the moment of inertia of the body

I_W is the moment of inertia of the wheels

M_B is the mass of the body

M_W is the mass of the wheel

L_B is the length of the body

r is the radius of the wheel

θ is the angle of the body

2. Kinetic and Potential Energy

2.1 Kinetic Energy

$$T = T_W + T_B$$

Given : T is kinetic energy

T_W is kinetic energy of wheels

T_B is kinetic energy of body

$$T = T_W + T_B$$

$$T = \underbrace{\frac{1}{2}(2M_W)v_W^2 + \frac{1}{2}I_W\omega_W^2}_{\text{Wheels}} + \underbrace{\frac{1}{2}M_B v_B^2 + \frac{1}{2}I_B\omega_B^2}_{\text{Body}}$$

$$\text{Given : } v_W = \dot{x},\ \ \omega_w = \frac{\dot{x}}{r} \text{ and } \omega_B = \dot{\theta}$$

$$v_B^2 = \left(\dot{x} - \frac{L_B}{2}\dot{\theta}\cos(\theta)\right)^2 + \left(\frac{L_B}{2}\dot{\theta}\sin(\theta)\right)^2$$

$$T = M_W\dot{x}^2 + \frac{1}{2}(M_W r^2)\left(\frac{\dot{x}}{r}\right)^2 + \frac{1}{2}M_B\left[\left(\dot{x} - \frac{L_B}{2}\dot{\theta}\cos(\theta)\right)^2 + \left(\frac{L_B}{2}\dot{\theta}\sin(\theta)\right)^2\right] + \frac{1}{2}\left(\frac{1}{3}M_B L_B^2\dot{\theta}^2\right)$$

$$T = \frac{3}{2}M_W\dot{x}^2 + \frac{1}{2}M_B\left(\dot{x} - \frac{L_B}{2}\dot{\theta}\cos(\theta)\right)^2 + \frac{1}{2}M_B\left(\frac{L_B}{2}\dot{\theta}\sin(\theta)\right)^2 + \frac{1}{6}M_B L_B^2\dot{\theta}^2$$

2.2 Potential Energy

$$P = P_W + P_B$$

$$P = \underbrace{2M_W g r}_{\text{Wheels}} + \underbrace{M_B g\left(r + \frac{L_B}{2}\cos(\theta)\right)}_{\text{Body}}$$

$$P = (2M_W g + M_B g)r + M_B g\frac{L_B}{2}\cos(\theta)$$

Given : P is potential energy

P_W is potential energy of wheels

P_B is potential energy of body

3. Find L

$$L = T - P$$

$$L = \frac{3}{2}M_W\dot{x}^2 + \frac{1}{2}M_B\left(\dot{x} - \frac{L_B}{2}\dot{\theta}\cos(\theta)\right)^2 + \frac{1}{2}M_B\left(\frac{L_B}{2}\dot{\theta}\sin(\theta)\right)^2 + \frac{1}{6}M_BL_B^2\dot{\theta}^2 - (2M_Wg + M_Bg)r - M_Bg\frac{L_B}{2}\cos(\theta)$$

4. Lagrange Equation in x coordinate

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) - \left(\frac{\partial L}{\partial x}\right) = 0$$

4.1 Find $\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right)$

$$\frac{\partial L}{\partial \dot{x}} = 3M_W\dot{x} + M_B\left(\dot{x} - \frac{L_B}{2}\dot{\theta}\,cos\,(\theta)\right)$$

$$\frac{\partial L}{\partial \dot{x}} = (3M_W + M_B)\dot{x} - \frac{M_BL_B}{2}\dot{\theta}\,cos\,(\theta)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) = (3M_W + M_B)\ddot{x} - \frac{M_BL_B}{2}\ddot{\theta}\,cos\,(\theta) + \frac{M_BL_B}{2}\dot{\theta}^2\sin(\theta)$$

4.2 Find $\left(\frac{\partial L}{\partial x}\right)$

$$\left(\frac{\partial L}{\partial x}\right) = 0$$

4.3 Substitute in Lagrange Equation

$$(3M_W + M_B)\ddot{x} - \frac{M_BL_B}{2}\ddot{\theta}\,cos\,(\theta) + \frac{M_BL_B}{2}\dot{\theta}^2\sin(\theta) = 0 \quad \blacksquare \text{ (Equation 1)}$$

5. Lagrange Equation in θ coordinate

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \left(\frac{\partial L}{\partial \theta}\right) = 0$$

5.1 Find $\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right)$

$$\frac{\partial L}{\partial \dot{\theta}} = M_B\left(\dot{x} - \frac{L_B}{2}\dot{\theta}\cos(\theta)\right)\left(-\frac{L_B}{2}\cos(\theta)\right) + M_B\left(\frac{L_B}{2}\dot{\theta}\sin(\theta)\right)\left(\frac{L_B}{2}\sin(\theta)\right) + \frac{1}{3}M_BL_B^2\dot{\theta}$$

$$\frac{\partial L}{\partial \dot{\theta}} = -\frac{M_BL_B}{2}\dot{x}\cos(\theta) + \frac{M_BL_B^2}{4}\dot{\theta}\cos^2(\theta) + \frac{M_BL_B^2}{4}\dot{\theta}\sin^2(\theta) + \frac{M_BL_B^2}{3}\dot{\theta}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) = -\frac{M_B L_B}{2}\ddot{x}\cos(\theta) + \frac{M_B L_B}{2}\dot{x}\dot{\theta}\sin(\theta) + \frac{M_B L_B^2}{4}\ddot{\theta}\cos^2(\theta) - \frac{M_B L_B^2}{2}\dot{\theta}^2\sin(\theta)\cos(\theta) + \frac{M_B L_B^2}{4}\ddot{\theta}\sin^2(\theta) + \frac{M_B L_B^2}{2}\dot{\theta}^2\sin(\theta)\cos(\theta) + \frac{M_B L_B^2}{3}\ddot{\theta}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) = -\frac{M_B L_B}{2}\ddot{x}\cos(\theta) + \frac{M_B L_B}{2}\dot{x}\dot{\theta}\sin(\theta) + \frac{7 M_B L_B^2}{12}\ddot{\theta}$$

5.2 Find $\left(\frac{\partial L}{\partial \theta}\right)$

$$\left(\frac{\partial L}{\partial \theta}\right) = M_B\left(\dot{x} - \frac{L_B}{2}\dot{\theta}\cos(\theta)\right)\left(\frac{L_B}{2}\sin(\theta)\right) + M_B\left(\frac{L_B}{2}\dot{\theta}\sin(\theta)\right)\left(\frac{L_B}{2}\cos(\theta)\right) + M_B g\frac{L_B}{2}\sin(\theta)$$

$$\left(\frac{\partial L}{\partial \theta}\right) = \frac{M_B L_B}{2}\dot{x}\dot{\theta}\sin(\theta) - \frac{M_B L_B^2}{4}\dot{\theta}^2\sin(\theta)\cos(\theta) + \frac{M_B L_B^2}{4}\dot{\theta}^2\sin(\theta)\cos(\theta) + M_B g\frac{L_B}{2}\sin(\theta)$$

$$\left(\frac{\partial L}{\partial \theta}\right) = \frac{M_B L_B}{2}\dot{x}\dot{\theta}\sin(\theta) + M_B g\frac{L_B}{2}\sin(\theta)$$

5.3 Substitute in Lagrange Equation

$$-\frac{M_B L_B}{2}\ddot{x}\cos(\theta) + \frac{M_B L_B}{2}\dot{x}\dot{\theta}\sin(\theta) + \frac{7 M_B L_B^2}{12}\ddot{\theta} - \frac{M_B L_B}{2}\dot{x}\dot{\theta}\sin(\theta) - M_B g\frac{L_B}{2}\sin(\theta) = 0$$

$$-\ddot{x}\cos(\theta) + \frac{7L_B}{6}\ddot{\theta} + g\sin(\theta) = 0 \quad \blacksquare \quad \text{(Equation 2)}$$

6. Eliminate $\ddot{x}$

$$(3M_W + M_B)\ddot{x} - \frac{M_B L_B}{2}\ddot{\theta}\,cos\,(\theta) + \frac{M_B L_B}{2}\dot{\theta}^2\sin(\theta) = 0 \quad \blacksquare \quad \text{(Equation 1)}$$

$$-\ddot{x}\cos(\theta) + \frac{7L_B}{6}\ddot{\theta} + g\sin(\theta) = 0 \quad \blacksquare \quad \text{(Equation 2)}$$

$$\text{(Equation 1)} \times \left(\frac{\cos(\theta)}{3M_W + M_B}\right) = \text{(Equation 3)}$$

$$\ddot{x}\cos(\theta) - \frac{M_B L_B}{2(3M_W + M_B)}\ddot{\theta}\cos^2(\theta) + \frac{M_B L_B}{2(3M_W + M_B)}\dot{\theta}^2\sin(\theta)\cos(\theta) = 0 \quad \blacksquare \quad \text{(Equation 3)}$$

(Equation 2) + (Equation 3) = (Equation 4)

$$\frac{7}{3}\ddot{\theta} + g\sin(\theta) - \frac{M_B L_B}{(3M_W + M_B)}\ddot{\theta}\cos^2(\theta) + \frac{M_B L_B}{(3M_W + M_B)}\dot{\theta}^2\sin(\theta)\cos(\theta) = 0$$

$$\frac{7}{3}\ddot{\theta} + \frac{g}{L_B}\sin(\theta) - \frac{M_B}{(3M_W + M_B)}\ddot{\theta}\cos^2(\theta) + \frac{M_B}{(3M_W + M_B)}\dot{\theta}^2\sin(\theta)\cos(\theta) = 0 \quad \blacksquare \text{ (Equation 4)}$$

7. Linearization $\left[\sin(\theta) \approx \theta,\ \cos(\theta) \approx 1,\ \dot{\theta} \approx 0\right] \rightarrow$ (Equation 4)

$$\frac{7}{3}\ddot{\theta} + \frac{g}{L_B}\theta - \frac{M_B}{(3M_W + M_B)}\ddot{\theta} = 0$$

$$\left(\frac{7}{3} - \frac{M_B}{(3M_W + M_B)}\right)\ddot{\theta} + \frac{g}{L_B}\theta = 0$$

$$\ddot{\theta} = -\frac{g}{L_B}\frac{(3M_W + M_B)}{(7M_W + 2M_B)}\theta \quad \blacksquare$$

8. State Space Equation

8.1 Input

$$x = \begin{bmatrix}\theta \\ \dot{\theta}\end{bmatrix}$$

$$\dot{x} = \begin{bmatrix}\dot{\theta} \\ \ddot{\theta}\end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L_B}\frac{(3M_W + M_B)}{(7M_W + 2M_B)} & 0 \end{bmatrix} x + \begin{bmatrix}0 \\ 1\end{bmatrix} u$$

8.2 Output

$$y = \begin{bmatrix}1 & 0\end{bmatrix} x$$

8.3 Coefficient Matrix

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L_B}\frac{(3M_W + M_B)}{(7M_W + 2M_B)} & 0 \end{bmatrix}, \qquad B = \begin{bmatrix}0 \\ 1\end{bmatrix}, \qquad C = \begin{bmatrix}1 & 0\end{bmatrix}$$

8.4 Controllability

$$CM = \begin{bmatrix}B & AB\end{bmatrix}$$

$$CM = \begin{bmatrix}0 & 1 \\ 1 & 0\end{bmatrix}$$

$$\text{rank}(CM) = 2$$

8.5 Observability

$$OM = \begin{bmatrix} C \\ CA \end{bmatrix}$$

$$OM = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\text{rank}(OM) = 2$$

8.6 Conclusion

A system has the full-row rank and full-column rank, it means that the system is controllable and observable.

9. State space Equation in x and θ coordinate

Given : x_0 is the distance

θ_0 is the angle

From

$$(3M_W + M_B)\ddot{x}_0 - \frac{M_B L_B}{2}\ddot{\theta}_0 \, cos\,(\theta_0) + \frac{M_B L_B}{2}\dot{\theta}_0{}^2 \sin(\theta_0) = 0 \quad \text{(Equation 1)}$$

$$-\ddot{x}\cos(\theta_0) + \frac{7L_B}{6}\ddot{\theta}_0 + g\sin\,(\theta_0) = 0 \quad \text{(Equation 2)}$$

9.1 Equivalent Equation

$$D(\theta)\ddot{\theta} + C(\dot{\theta},\theta)\dot{\theta} + G(\theta) = Hu$$

Where, $\ddot{\theta} = \begin{bmatrix} \ddot{x}_0 \\ \ddot{\theta}_0 \end{bmatrix}$, $\quad \dot{\theta} = \begin{bmatrix} \dot{x}_0 \\ \dot{\theta}_0 \end{bmatrix}$, $\quad \theta = \begin{bmatrix} x_0 \\ \theta_0 \end{bmatrix}$

$$D(\theta) = \begin{bmatrix} 3M_W + M_B & \frac{M_B L_B}{2}\,cos\,(\theta_0) \\ -\cos(\theta_0) & \frac{7L_B}{6} \end{bmatrix}, \qquad C(\dot{\theta},\theta) = \begin{bmatrix} 0 & \frac{M_B L_B}{2}\sin(\theta_0) \\ 0 & 0 \end{bmatrix}$$

$$G(\theta) = \begin{bmatrix} 0 \\ g\sin(\theta_0) \end{bmatrix}, \qquad H = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

9.2 Linearization $\left[\sin(\theta) \approx \theta,\ \cos(\theta) \approx 1,\ \dot{\theta}\ \approx 0\,\right]$

$$D(\theta)\ddot{\theta} + C\big(\dot{\theta},\theta\big)\dot{\theta} + G(\theta) = Hu$$

$$\begin{bmatrix} 3M_W + M_B & \dfrac{M_B L_B}{2} \\ -1 & \dfrac{7L_B}{6} \end{bmatrix}\ddot{\theta} + \begin{bmatrix} 0 & 0 \\ 0 & g \end{bmatrix}\theta = \begin{bmatrix} 1 \\ 0 \end{bmatrix}u$$

$$\ddot{\theta} = -\begin{bmatrix} 3M_W + M_B & \dfrac{M_B L_B}{2} \\ -1 & \dfrac{7L_B}{6} \end{bmatrix}^{-1}\begin{bmatrix} 0 & 0 \\ 0 & g \end{bmatrix}\theta + \begin{bmatrix} 3M_W + M_B & \dfrac{M_B L_B}{2} \\ -1 & \dfrac{7L_B}{6} \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix}u$$

$$\ddot{\theta} = \begin{bmatrix} 0 & \dfrac{M_B g}{2\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} \\ 0 & \dfrac{(3M_W + M_B)g}{L_B\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} \end{bmatrix}\theta + \begin{bmatrix} \dfrac{7}{6\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} \\ \dfrac{1}{L_B\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} \end{bmatrix}u$$

9.3 Equivalent State Space From,

$$x = A\dot{x} + Bu$$
$$y = Cx$$

Where,

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{M_B g}{2\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} & 0 & 0 \\ 0 & -\frac{(3M_W + M_B)g}{L_B\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{7}{6\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} \\ \frac{1}{L_B\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\dot{x} = \begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} \dot{x}_0 \\ \dot{\theta}_0 \\ \ddot{x}_0 \\ \ddot{\theta}_0 \end{bmatrix}, \quad x = \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} x_0 \\ \theta_0 \\ \dot{x}_0 \\ \dot{\theta}_0 \end{bmatrix}$$

10. Controllability and Observability

10.1 Controllability

$$CM = [B \quad AB \quad A^2B \quad A^3B]$$

$$CM = \begin{bmatrix} 0 & \frac{7}{6\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} & 0 & -\frac{M_B g}{2L_B\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)^2} \\ 0 & \frac{1}{L_B\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} & 0 & -\frac{(3M_W + M_B)g}{L_B^2\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)^2} \\ \frac{7}{6\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} & 0 & -\frac{M_B g}{2L_B\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)^2} & 0 \\ \frac{1}{L_B\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} & 0 & -\frac{(3M_W + M_B)g}{L_B^2\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)^2} & 0 \end{bmatrix}$$

$$\mathrm{rank}(CM) = 4$$

10.2 Observability

$$OM = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix}$$

$$OM = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{M_B g}{2\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} & 0 & 0 \\ 0 & -\frac{(3M_W + M_B)g}{L_B\left(\frac{7}{2}M_W + \frac{2}{3}M_B\right)} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathrm{rank}(OM) = 4$$

10.3 Conclusion

A system has the full-row rank and full-column rank, it means that the system is controllable and observable.

PID Control

1. Turn state space equations into transfer function by MATLAB

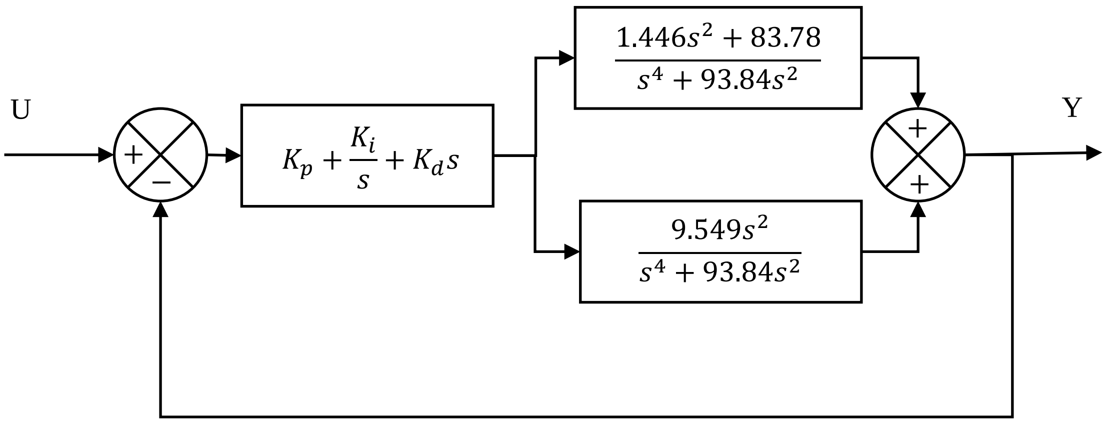
MATLAB Program 0-1

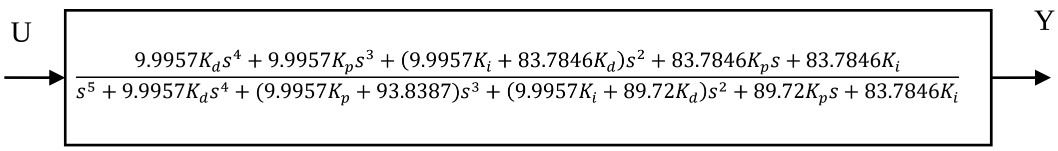
```
%*****Finding transfer function from state space equations******

M = 1.00;
m = 0.04;
g = 9.8;
L = 0.145;

A = [0,0,1,0;
     0,0,0,1;
     0,-M*g/(2*((7*m/2)+(2*M/3))),0,0;
     0,-((3*m)+M)*g/(L*((7*m/2)+(2*M/3))),0,0];

B = [0;0;7/(6*((7*m/2)+(2*M/3)));1/(L*((7*m/2)+(2*M/3)))];

C = [1,0,0,0;0,1,0,0];

D = [0;0];

[num,den] = ss2tf(A,B,C,D)


%*****Result*****

num =

         0         0    1.4463         0   83.7846
         0         0    8.5494         0         0


den =

    1.0000         0   93.8387         0         0
```

MATLAB Program 0-2

```
%*****Showing Two transfer function for two outputs*****

num1 = [0,0,1.4463,0,83.7846];
num2 = [0,0,8.5494,0,0];

sys1 = tf(num1,den)
sys2 = tf(num2,den)


%*****Result*****

sys1 =

          1.446 s^2 + 83.78
          ---------------------
            s^4 + 93.84 s^2

Continuous-time transfer function.


sys2 =

             9.549 s^2
          -----------------
          s^4 + 93.84 s^2

Continuous-time transfer function.
```

2. Block Diagram

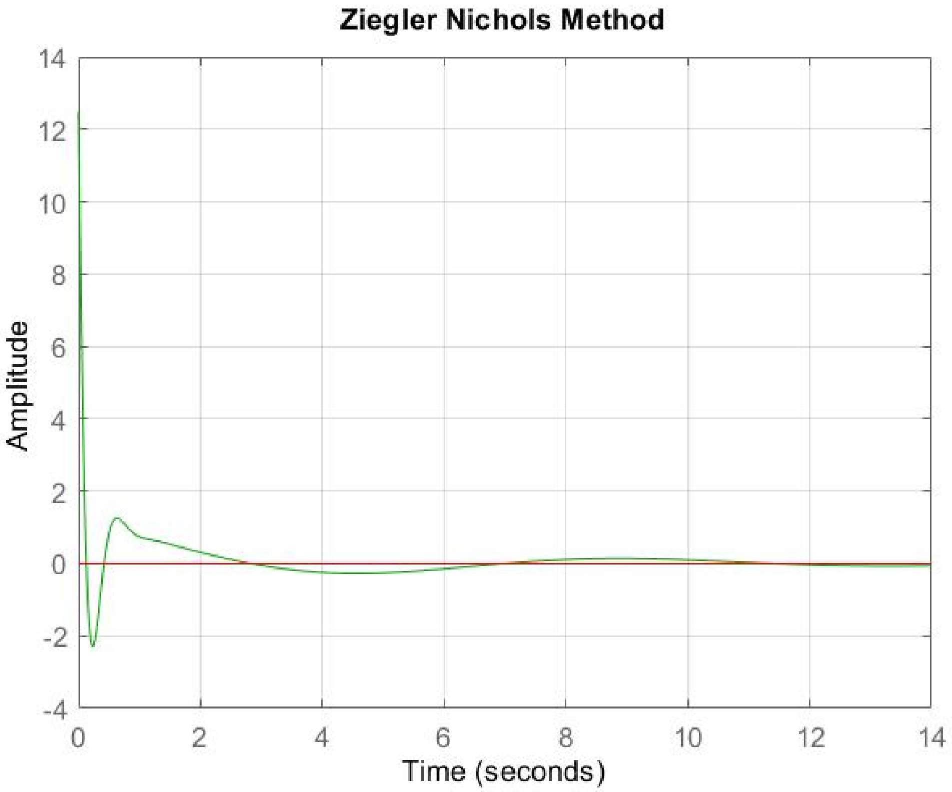




3. Tuning PID parameters

MATLAB Program 0-3

```
%***** Tuning PID by Ziegler Nichols Method *****
a = 9.9957;
b = 83.7846;
c = 93.8387;

subplot(1,1,1);
Kp = 0.9;
Ki = 0.45;
Kd = 1.25;
num = [Kd*a,Kp*a,(Ki*a)+(Kd*b),Kp*b,Ki*b];
den = [1,Kd*a,(Kp*a)+c,(Ki*a)+(Kd*b),Kp*b,Ki*b];
sys = tf(num,den);
impulse(sys,'g');

grid on

title('Ziegler Nichols Method')
```



Arduino Code

Arduino Code 0-1

```
#include "I2Cdev.h"
#include "MPU6050.h"
#include "Wire.h"
MPU6050 mpu;
int16_t ax, ay, az;
int16_t gx, gy, gz;

int angle;
int speedMotor;

unsigned long currentTime, previousTime;
double elapsedTime, error, setPoint = 170, input, lastError, output;
double cumError, rateError, Kp = 0.5, Ki = 0, Kd = 0.5;

void setup() {
  Serial.begin(9600);
  pinMode(3, OUTPUT); //ENB
  pinMode(4, OUTPUT);
  pinMode(5, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(7, OUTPUT);
  pinMode(6, OUTPUT); //ENA
  Wire.begin();
  mpu.initialize();
 // calibrate(); //In 20 sec, adjust min & max angle
}

void loop() {
  mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
  angle = map(az, -17000, 17000, 0, 179);
  input = angle;
  output = computePID(input);
  delay(10);

  speedMotor = map(output, -1, 1, -255, 255);
  speedMotor = abs(speedMotor);
  speedMotor = map(speedMotor, 0, 255, 75, 100);
  speedMotor = constrain(speedMotor, 0, 75);

  if (output >= 0)
  {
    digitalWrite(5,LOW);
    digitalWrite(4,HIGH);
    analogWrite(3,speedMotor);
    digitalWrite(8,LOW);
    digitalWrite(7,HIGH);
    analogWrite(6,speedMotor);

  }
```

```
else
  {
    digitalWrite(5,HIGH);
    digitalWrite(4,LOW);
    analogWrite(3,speedMotor);
    digitalWrite(8,HIGH);
    digitalWrite(7,LOW);
    analogWrite(6,speedMotor);
  }


  Serial.print("output= ");
  Serial.print(output);
  Serial.print("    ");
  Serial.print("speed= ");
  Serial.println(speedMotor);

}

double computePID(double inp)
{
  currentTime = millis();
  elapsedTime = (double)(currentTime - previousTime);

  error = setPoint - inp;
  cumError += error * elapsedTime;
  rateError = (error - lastError)/elapsedTime;

  double out = Kp*error + Ki*cumError + Kd*rateError;

  lastError = error;
  previousTime = currentTime;

  return out;
}
```

Arduino Code 0-2

```
#include "I2Cdev.h"
#include "MPU6050.h"
#include "Wire.h"
MPU6050 mpu;
int16_t ax, ay, az;
int16_t gx, gy, gz;
int valx , valy , valz;
char rd;
int prevVal;
int pin11 = 11 , pin10 = 10 ;
int val1 , val2 ;
int valgy1 = 0 , valgy2 = 0;
byte countS = 0; //03

int recOmegaI[10]; //05
int omegaI = 0;
int zeromegaI = 0;//06
long thetaI = 0; //07
long sumPower = 0; //08
long sumSumP = 0; //09
const int kAngle = 50; //10
const int kOmega = 500; //11
const long kSpeed = 60; //12
const long kDistance = 20; //13
long powerScale; //14
int power; //15
long vE5 = 0; //16
long xE5 = 0; //17

int ry; //DL2
long R; //DL3

void setup () { //18
 Wire.begin();
Serial.begin(38400);

 pinMode(4, OUTPUT); //20
 pinMode(5, OUTPUT); //20-a
 pinMode(6, OUTPUT); //21
 pinMode(7, OUTPUT);
 pinMode(8, OUTPUT);
 pinMode(9, OUTPUT);
 for ( int i = 0 ; i < 10 ; i++ ) { recOmegaI[i] = 0; } //25 ("int" is added instead of line 2 omitted.)
 pinMode(10, OUTPUT); //DL18 (These 8 lines, DL18-DL25, are added in this version.)
 digitalWrite(10, HIGH); //DL19

Serial.println("Initialize MPU");
mpu.initialize();
Serial.println(mpu.testConnection() ? "Connected" : "Connection failed");
```

```
delay(300); //26
// training(); // (This line is omitted in this version.)
// MsTimer2::set(5, chkAndCtl); // (This line is omitted in ver.2.0 and the later.)
// MsTimer2::start(); // (This line is omitted in ver.2.0 and the later.)
} //30
void loop () { //31
 chkAndCtl(); // NL1 (This line is added in ver.2.0 and the later.)
 if ( power > 0 ) { //32
 analogWrite( 6, power );
 digitalWrite( 5, HIGH );
 digitalWrite( 4, LOW ); //35
 analogWrite( 9, power );
 digitalWrite( 7, HIGH );
 digitalWrite( 8, LOW );
 } else {
 analogWrite( 6, - power ); //40
 digitalWrite( 5, LOW );
 digitalWrite( 4, HIGH );
 analogWrite( 9, - power );
 digitalWrite( 7, LOW );
 digitalWrite( 8, HIGH ); //45

 }
// delayMicroseconds(3600); // NL2 (This is omitted in this version.)
Serial.println(power) ;
}
//void training(){ //48 (These 7 lines, 48-54, are omitted in this version.)
// delay (1000);
// for ( i = 0 ; i < 500 ; i++ ){ //50
// zeroOmegaI = zeroOmegaI + analogRead(A5);
// }
// zeroOmegaI = zeroOmegaI / i;
//} //54
void chkAndCtl() { //55
// omegaI = 0; // NL3 (These 6 lines, NL3-NL8, are omitted in this version.)
// for ( i = 0 ; i < 10 ; i++ ) { //NL4
// omegaI = omegaI + analogRead(A5) - zeroOmegaI; //NL5
// delayMicroseconds(10); //NL6
// } //NL7
// omegaI = omegaI / 10; //NL8
  //DL26 (These 7 lines, DL26-DL32, are added in this version.)

  zeromegaI=0;
 for ( int i = 0 ; i < 45 ; i++ ) { //DL27 ("int" is added instead of line 2 omitted.)
mpu.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
valx = map(ax, -17000, 17000, 0, 179) -70;
 zeromegaI = valx + zeromegaI;

Serial.println(power) ;

 Serial.println(valx) ;
```

```
 delayMicroseconds(90); //DL30
 } //DL31
 omegaI = zeromegaI/45  ; //DL32
// omegaI = analogRead(A5) - zeroOmegaI; //56 (This line is omitted in ver.2.0 and the later.)
 if ( abs( omegaI ) < 8 ) { omegaI = 0; } //57 (The lower bound is less than 2 in this version.)
 recOmegaI[0] = omegaI;
 thetaI = thetaI + omegaI;
 countS = 0; //60
 for ( int i = 0 ; i < 10 ; i++ ) { // ("int" is added instead of line 2 omitted.)
 if ( abs( recOmegaI[i] ) < 4 ) { countS++; } //62 (The lower bound is less than 4 in this version.)
 }
 if ( countS > 9 ) {
 thetaI = 0; //65
 vE5 = 0;
 xE5 = 0;
 sumPower = 0;
 sumSumP = 0;
 } //70
 for ( int i = 9 ; i > 0 ; i-- ) { recOmegaI[ i ] = recOmegaI[ i-1 ]; } // ("int" is added instead of line 2 omitted.)
 powerScale = ( kAngle * thetaI / 100 ) + ( kOmega * omegaI / 100 ) + ( kSpeed * vE5 / 1000 ) + ( kDistance
* xE5 / 1000 ); //72
 power = max ( min ( 95 * powerScale / 100 ,50 ) , -50 );
 if (power>0){power+30;}
  if (power<0){power-30;}

 sumPower = sumPower + power;
 sumSumP = sumSumP + sumPower; //75
vE5 = sumPower; //76a
xE5 = sumSumP / 1000; } //78
// Copyright (C) 2014 ArduinoDeXXX All Rights Reserved. //79
```

Discussion

Timeline

1. We picked up the model from internet.
2. We needed to buy the equipment that the model required.
3. We built the robot.
4. We put the code in the robot.
5. We found that this code can't be used.
6. We tried to find the new code that from the other sources and friends.
7. We designed the 2 new robots.
   7.1 The first one is Choco Pie Bot.
   7.2 The second is the 2 plates Bot.
8. The code that we got; it still can't be used.
9. We decided to read the old code again and tried the code to the old robot.

Problems

1. We did not read comments before picked that model.
2. Some of components did not apply for our project. For example, IC driver could not be used, so I changed it to be motor drive (L298N). The reason why we choose L298N driver is IC driver did not response to the motor.
3. L298N driver divided the unstable signal for each side.
   However, we had checked all devices already. We solved by change the new one.
4. While we used L298N for a while, it suddenly did not work and
   we wasted a lot of time to attempt solved the problem.
   Finally, we found that the jumper wire did not connect properly.
5. Firstly, we used battery which is not suit for model.
   We solved by change to lithium-ion battery.
6. The code that gave from the internet did not work. So, first way we consult with friends and second way we look back to the original code.
7. We have to recharge battery many times.
8. Since the hardware has problems, we could find the PID value.
9. Gyro sensor lost connection many times.

References

1. **Another Easier Inverted Pendulum Robot**

   https://www.instructables.com/Another-Easier-Inverted-Pendulum/

2. **MPU6050**

   https://www.arduino.cc/reference/en/libraries/mpu6050/

3. **Github**

   https://github.com/jrowberg/i2cdevlib