Report 2 - Step Response of the Balanduino

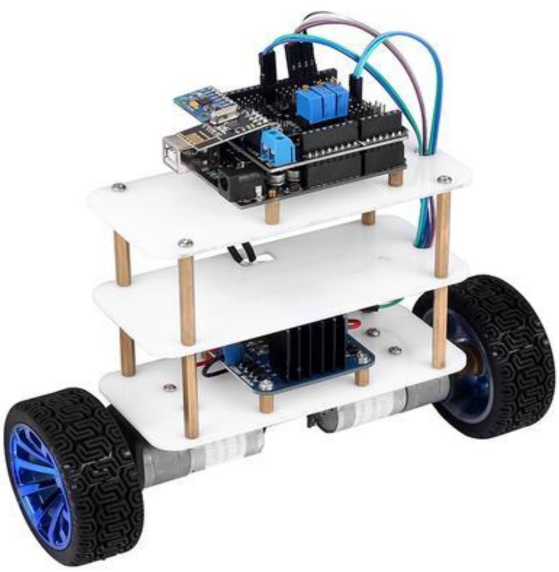
Deepak **Group leader** (B16054)

Mukesh Kumawat (B15222)

Anand Kumar (B16045)

Amit Ghanghas (B15208)

April 27,2018



Contents

List of Figures

Chapter 1

Time Response of the Robot

1.1 objective

To obtain the linearized model for Balanduino robot and calculate its time response for a step input.

1.2 Assumptions

While considering the mass of the pendulum , mass of the frame was neglected.

1.3 Theory

Method to obtain the time response of the system for a unit step input-
The non-linear model was linearized about the vertical state of the robot and the state space equations were obtained in the form:

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

Taking Laplace Transform of the above equations , we get

$$sX(s) = AX(s) + BU(s)$$
$$(sI-A)X(s)=BU(s)$$
$$X(s)= (sI-A)^{-1}\,BU(s)$$
$$Y(s) = CX(s) + DU(s)$$
$$Y(s) = C(sI-A)^{-1}\,BU(s)+ DU(s)$$

From the above two equations we can get the transfer function of the model. Since the input voltage is given to be a step input we substitute U(s) = 1/s to get the step response in frequency domain. Taking inverse Laplace transform of Y(s) we get the time response y(t) to a step input.

y(t) can then be plotted. In MATLAB, the time response to a step input can be obtained directly using the command step(system). Obtaining the constants of the state space model-

- mp = mass of Arduino +Mass of Lipo battery + 2*Mass of Motor
  =) mp = (37 + 245 + 2 * 140) * 10^{2} = 0:562kg

- mw = 0:02kg

- r = 3:015cm

- Jp = Moment of Inertia of Battery + Moment of Inertia of Arduino + Moment of Inertia of Motors
  =) Jp = (9.36 + 0.96 + 2 * 0.4536) * 10^{-4} = 10.663) *10^{-4}kgm^{2}

- Jw = mw * r2 = 0.18 *10^{-4}kgm^{2}

- Ra = 12.617ohm

- ke = 0.3651

- n * kt = 2.3

- L = 4.56cm

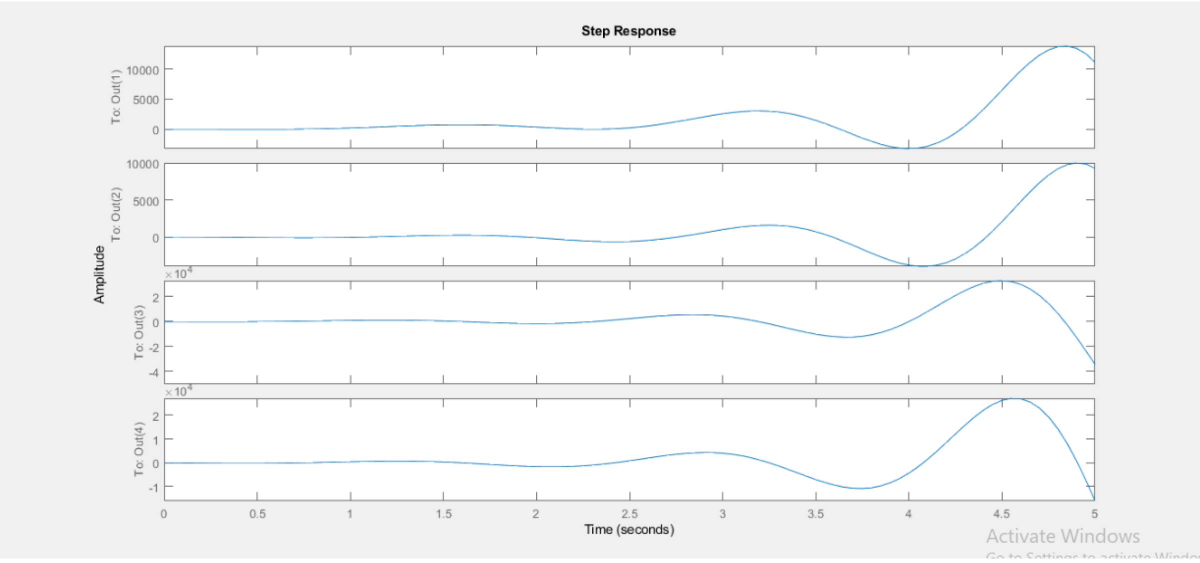
1.4 MATLAB code

```
%physical constants
g = 9.81;n = 1;
mp = 0.562;
mw=0.02;
r=3.015*10e-2;
jp=10.663*10e-4;
jw=0.18*10e-4;
ra=12.617;
ke=0.3651;
kt=2.3;
l=4.56*10e-2;
%matrix terms
b=(-(l^2*r^2*mp)+jp*(jw+r^2*(mp+mw)));
a32=(g*(l^2)*r*mp^2)/b;
a33=((-2*n*jp*ke*kt)-(2*l*n*r*ke*kt*mp))/(b*ra);
a42=(g*l*jw*mp*ra+l*r^2*mp*(g*mp+g*mw)*ra)/(b*ra);
a43=(-2*n*jw*ke*kt-2*n*r*ke*kt*((l+r)*mp)+r*mw)/(b*ra);
b31=((2*n*jp*kt)+(2*l*n*r*kt*mp))/b*ra;
b41=(2*n*jw*ke*kt+2*n*r*kt*((l+r)*mp)+r*mw)/b*ra;
%A B C D matrix of state space model
A = [0 0 1 0
    0 0 0 1
    0 a32 a33 0
    0 a42 a43 0];
B = [0
    0
    b31
    b41];
C=[1 0 0 0
    0 1 0 0
    0 0 1 0
    0 0 0 1];
D=0;
%state space model and step response plotting
system = ss(A,B,C,D);
step(system,5);%impulse(system,5)  --impulse response plotting
```

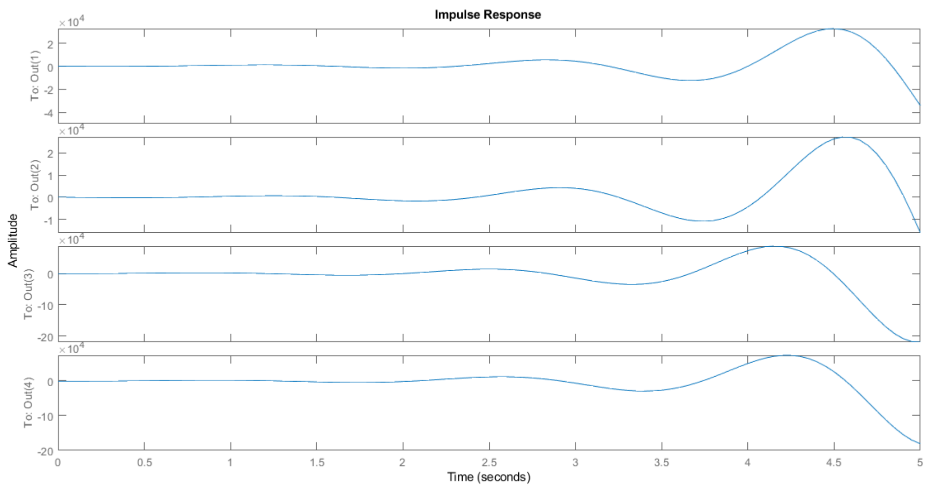
Physical constants & State Space Model

1.5 Observations

By using above values of the constants, the following plots were obtained.



Step Response Plots



Impulse Response Plots

```
A =

         0         0    1.0000         0
         0         0         0    1.0000
         0  -19.3152    1.1638         0
         0  -13.2704    1.6537         0

Trial>> eigs(A)

ans =

   1.1101 + 3.7926i
   1.1101 - 3.7926i
  -1.0565 + 0.0000i
   0.0000 + 0.0000i
```

Matrix A & Eigen values of A

1.6 Conclusion

Clearly, the Balanduino is an asymptotically unstable system for the impulse response is ever increasing oscillations. From the step response it is also clear that the system is not BIBO stable as well. This matches with the prediction based on eigen values that the system is expected to be unstable.

References

_ https://www.robotshop.com/en/ban-90-green-wheel.html