***Abstract*— Document is about a Segway robot. The report focuses on the design and development of a control system in order to balance a two-wheeled Segway styled robot. However, the physical robot (hardware) on which the control system was to be tested was not possible due the current pandemic so we can’t properly finished hardware for it but simulation was used instead. The aim behind it was to prepare a 2-D model and improvise software in a way that it allows balancing stability of the robot to be achieved for it to be able to maintain an upright position while disturbances are introduced in the system. Electrical and kinematic parameters are determined theoretically, PID or LQR controller designs are also performed on the linearized equations of motion, where results show that self-balancing can be achieved using the PI or PD controller in transfer function or LQR in stat-space model for vicinity of the upright position.**

***Keywords: Segway; linearized; stability; PID controller; LQR; state-space model***

**I. INTRODUCTION**

In this world of advanced technology, Robotics has been performing an integral part of the human psyche. The dream of creating a machine that replicates human thought and physical characteristics extends throughout the existence of mankind. Mobile robots have stepped out the industrial settings, and have entered civilian and personal spaces such as hospitals, universities as well as in households. Two wheeled balancing robots are an area of research that may well provide the future locomotion for everyday robots. Effective and efficient control system designs provide the robot with the ability to control itself and operate autonomously. Perhaps because of the increasing hype of the Segway, two-wheeled balancing robots have become a prominent project for a lot of engineering students. It is certainly a topic of interest to anyone involved in the study of control systems and their application to robotics.

Segway robot is an example of inverted pendulum on two wheels capable of balancing itself and movement.

**II. LITERATURE REVIEW**

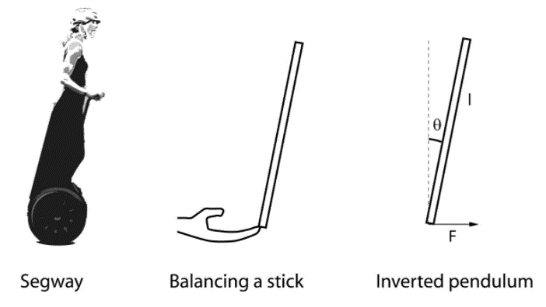
**Two wheeled balancing Robots:** A robot that is able to balance upright on its two wheels is known as a two wheeled balancing robot. The two wheels are arranged underneath the base and permit the robot chassis to maintain an upright position by mobility within the course of tilt, either forward or in reverse, in an attempt to keep the middle of the mass over the wheel axles. The wheels also provide the locomotion thus allowing the robot to transverse across various tracks. Segway is one of these two wheeled robots that has become commercially successful and has proven a comfortable mobility opportunity. The theory used behind the stability of these robots is based on the inverted pendulum.

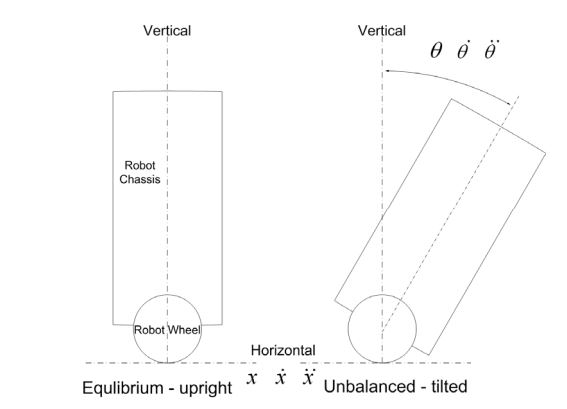
A similar and commercially available system, ‘SEGWAY HT’ has been invented by DEAN KAMEN, who holds more than 150 U.S. and foreign patents related to medical devices, climate control systems, and helicopter design. The ‘SEGWAY HT’ is able to balance a human standing on its platform while the user traverses the terrain with it. This innovation uses five gyroscopes and a collection of other tilt sensors to keep itself upright. Only three gyroscopes are needed for the whole system, the additional sensors are included as a safety precaution

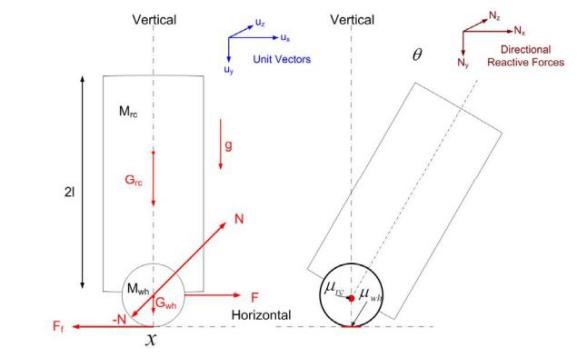


**The Inverted Pendulum:** The inverted pendulum is a classical problem that is frequently analyzed in control systems, and to explore the unstable dynamics, different platforms have been developed.

It is commonly described as a pendulum, or a rigid rod with a bob on the end, mounted by a hinge to a base, which can be translated along a track by an input force. So, this inverted pendulum theory is applied to develop a reliable and capable control system for a two wheeled balancing robot, as understanding of the parameters within the system is essential.





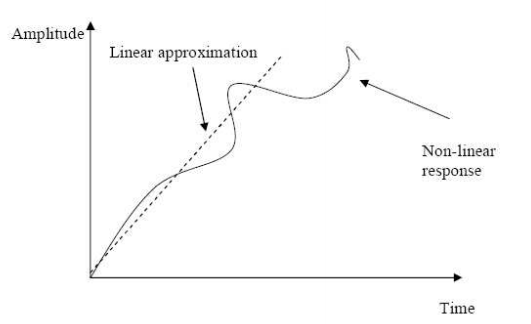


**System Dynamics:** The system dynamics that will be further utilized in the mathematical problem of this two wheel balancing robots stability control (inverted pendulum approach) are below:

* x Displacement (Horizontal) (m)
* x’ Velocity (Horizontal) (ms-1)
* x’’ Acceleration (Horizontal) (ms-2)
* θ Angular displacement (Vertical) (rad s)
* θ’ Angular velocity (Vertical) (rad s-1)
* θ’’ Angular acceleration (Vertical) (rad s-2)
* mw Mass of the wheels and drive shafts (Kg)
* mc Mass of the robot chassis (Kg)
* h Height of the robot chassis (m)

**Control Systems:** Control systems are classified as either Linear or Non-Linear control systems. Since, Non- linear systems have a higher complexity and hard to use in application, linear systems are generally preferred. Linear systems include State Space control, Proportional Integral and Derivative (PID) controllers, Linear Quadratic Regulator (LQR) and pole placement controllers. A microcontroller provides the computational power to allow the robot to balance itself, based on the sensor input information. To maintain the robot upright, the most commonly used controllers are Proportional Integral Derivative (PID) and the Linear Quadratic Regulator.

The modeling of a system requires consideration and application of a lot of parameters which makes the system a bit complex. This is why linear approximation is preferred for the model, which is simpler and is more effective in some instances. However Linear-control theory is not suitable for real life implementations, which mostly exhibit non-linear response. For better performance some non-linear approximation can be applied. Figure below shows how a non-linear response can be approximated to a linear response.



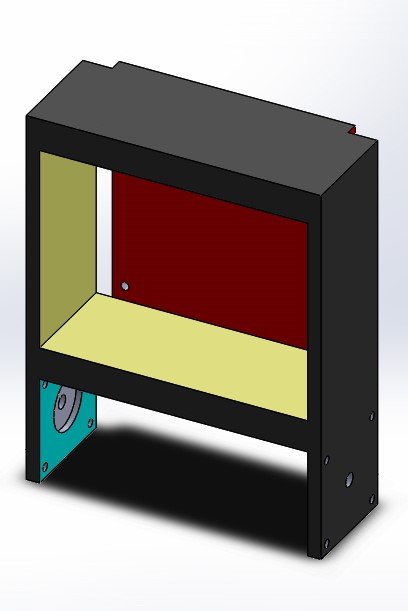
In the research titled ‘Comparative Study of Control Methods of Single – Rotational Inverted Pendulum’ conducted by XU & DUAN (1997) proved that the LQR controllers are far better than the pole placement controller for balancing an inverted pendulum mounted on a rotation arm because the LQR controller offers an optimal control over the system’s input by taking into account the states of the system and the control input. The arbitrary placement of control poles for Pole-placement controllers might cause the poles to be placed too far into the left-hand plane and cause the system susceptible to disturbances.

**Tilt angle Estimation:** Knowing the tilt angle is imperative in order to maintain the robot upright. There are a wide array of sensors that can be used, such as inclinometers, light sensors, accelerometer or gyroscopes. However, all of these sensors have their own shortcomings, the inclinometer takes a long time to converge to the angle it is currently at, light sensors are highly susceptible ambient light and the reflective index of the surface it is operating in and other background noises, gyroscopes have a bias and accelerometers are relatively noisy. Most often, a combination of a gyroscope and an accelerometer are used.

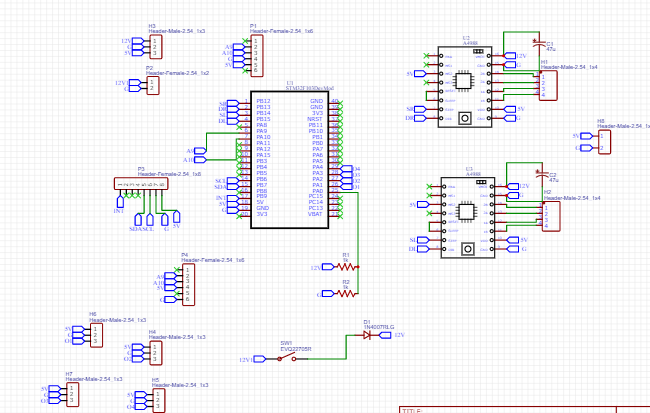
**Components & Materials Required:**

* STM32F103
* A4988
* Headers\Connecters\Diode\switch
* Capacitors
* Resistors
* Gyroscope sensor
* TTL programmer
* Stepper motors

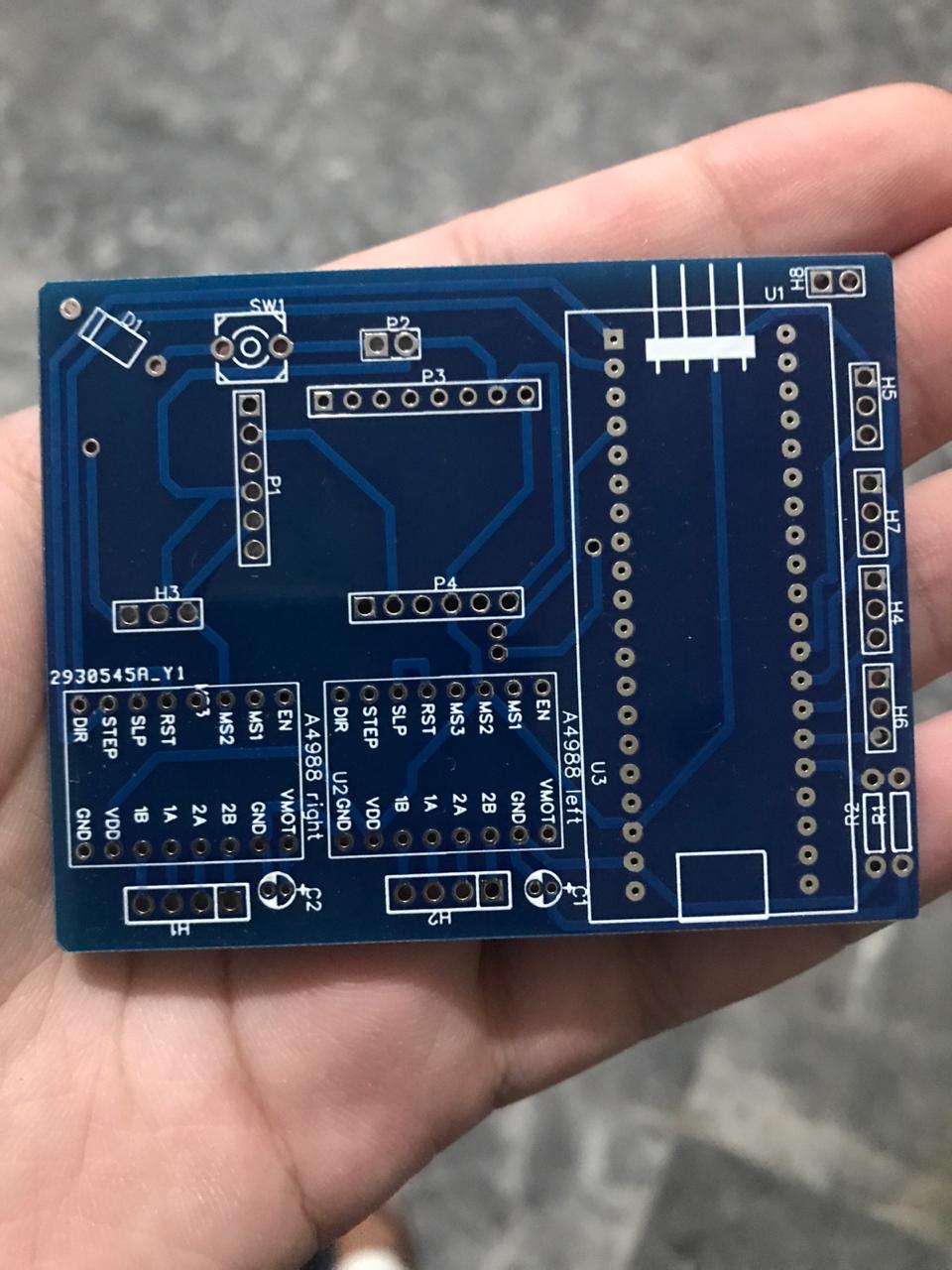
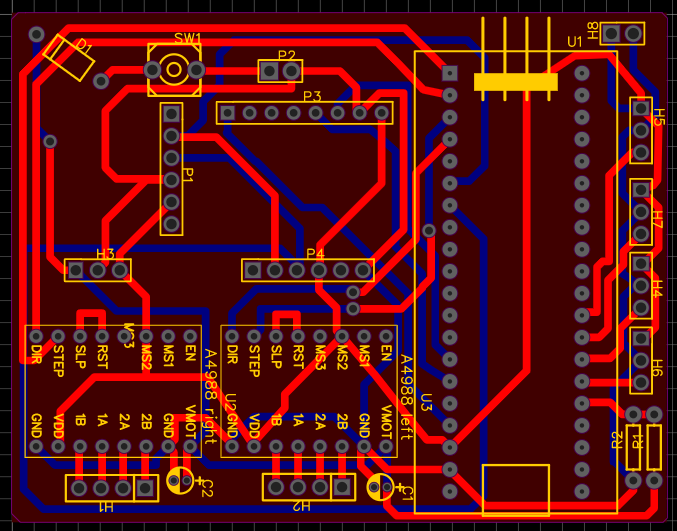
**Body:**



**Circuit:**

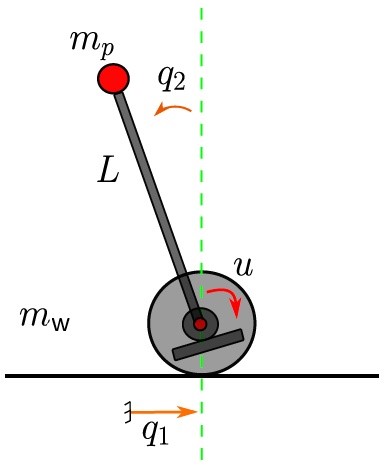


**PCB:**



**Conclusion:** Two wheeled balancing robots consist of a robot chassis, two wheels and a stability control system. Additional components and attachments may be fitted depending on the task required to be performed. There strengths lie in the ability to turn on the spot and easily maneuver in confined areas. Stability is achieved by keeping the wheels beneath the mass of the robot chassis. A non-linear control system for stability allows a more effective and robust control to be implemented. Two wheeled balancing robots offer a revolutionary transportation capability. Now that an understanding and knowledge of two wheeled balancing robots have been achieved, the following units in the report shall begin the design and development section of this project.

**III. Mathematical Model**



**Parameters:**

Mass of the wheel base

Mass of the pendulum body

Gravity

Length of pendulum body

Damping of wheel base displacement

Damping in the pendulum body and wheel base joint

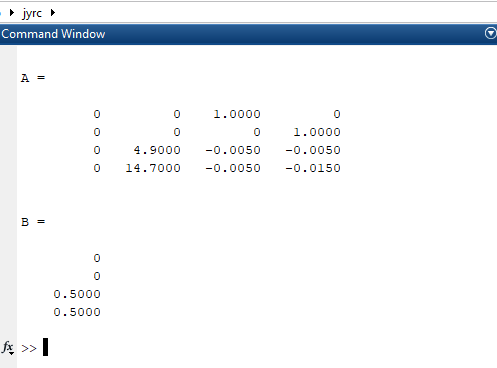
**Modeling:**

The equations of motion for the described system can be derived using Lagrange-Formalism or Newton-Euler. The result is in a nonlinear system.

Where d1 and d2 are the damping factors representing friction in the wheel base displacement and the joint between it and pendulum body as mentioned in parameters.

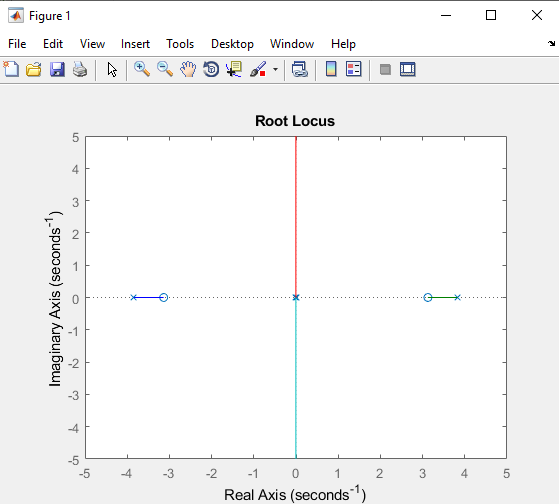
A state-space representation can be derived using the equations above defining

Linearizing about the dq1 = dq2 = 0 yields the linear state-space equation of system



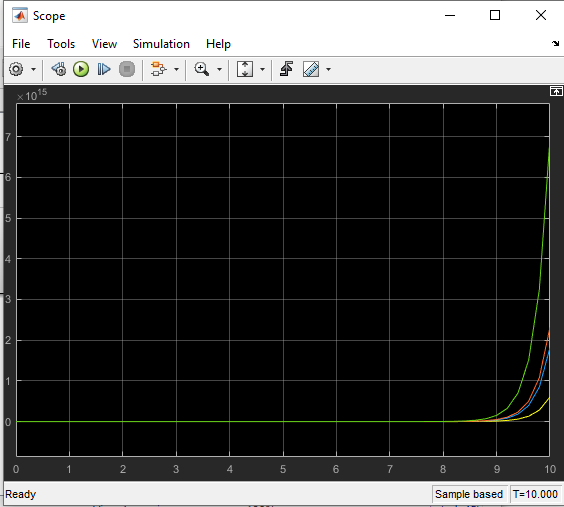
**IV. Pole and zeros map**

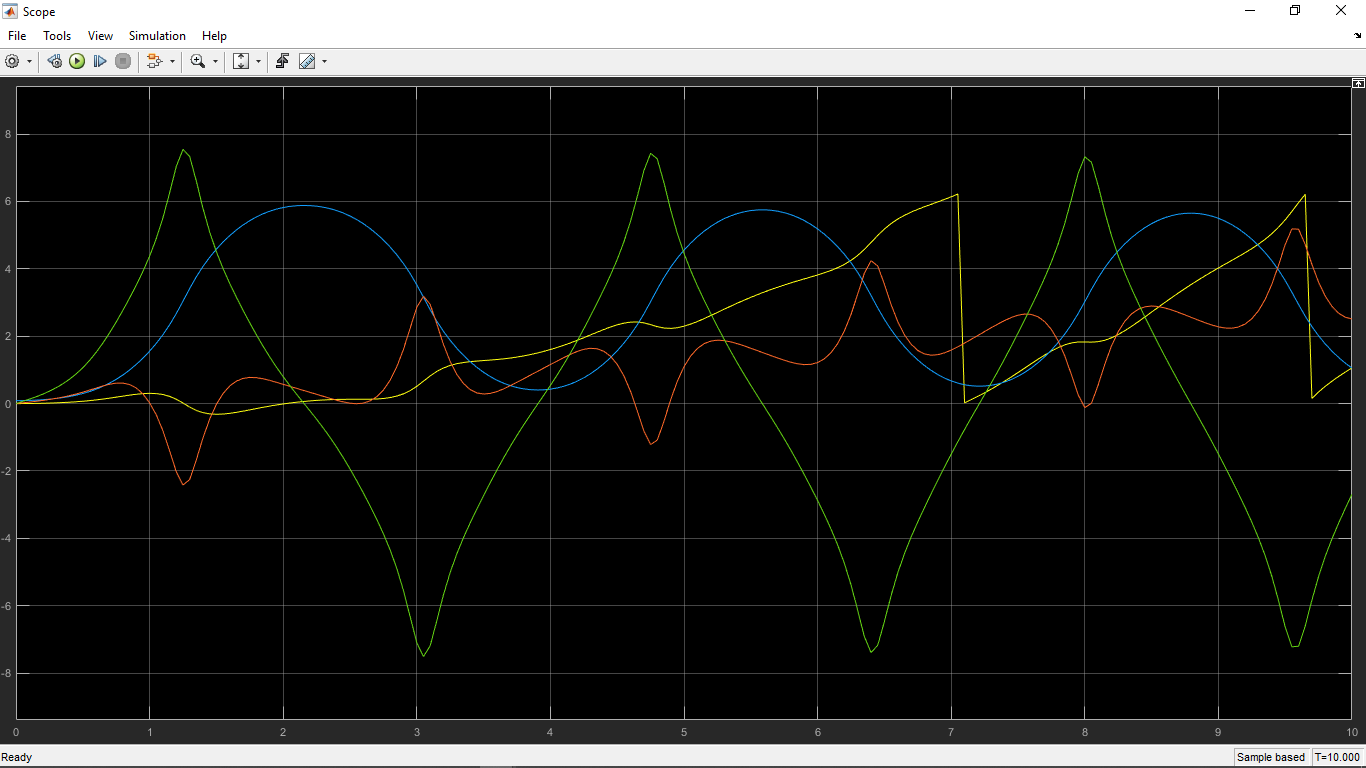




The system is unstable and we have to design a controller in order to stable the system. We are in state-space so we have to design LQR controller (Liner Quadratic Regulator) for optimal control

**V. Step response**System is highly unstable as mentioned previous the step response also show that as the some poles lies on the right side of the p-z map





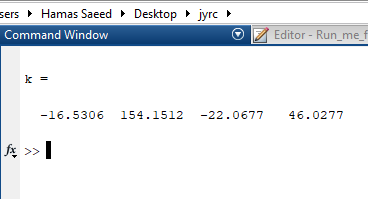
**VI. Controller Design**

We have a problem with the total stability of the system. That was, though the input-output mapping of the system seemed to be stabilized using a proper controller and the angle of the pendulum converted to zero, the position of the wheel base diverged. Two important features of a dynamical system are controllability and observability.

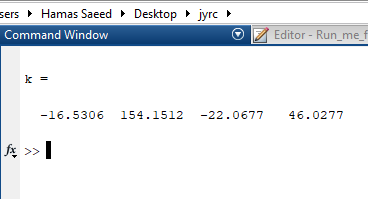
We are going to define the wheel base position q1 as output. And designing a state feedback controller K to shift the poles of the closed loop to proper positions and then create the closed loop using the system matrices and K. Now the desired closed loop poles are:

Now designing an optimal controller using LQR as mentioned previous

By using MATLAB command

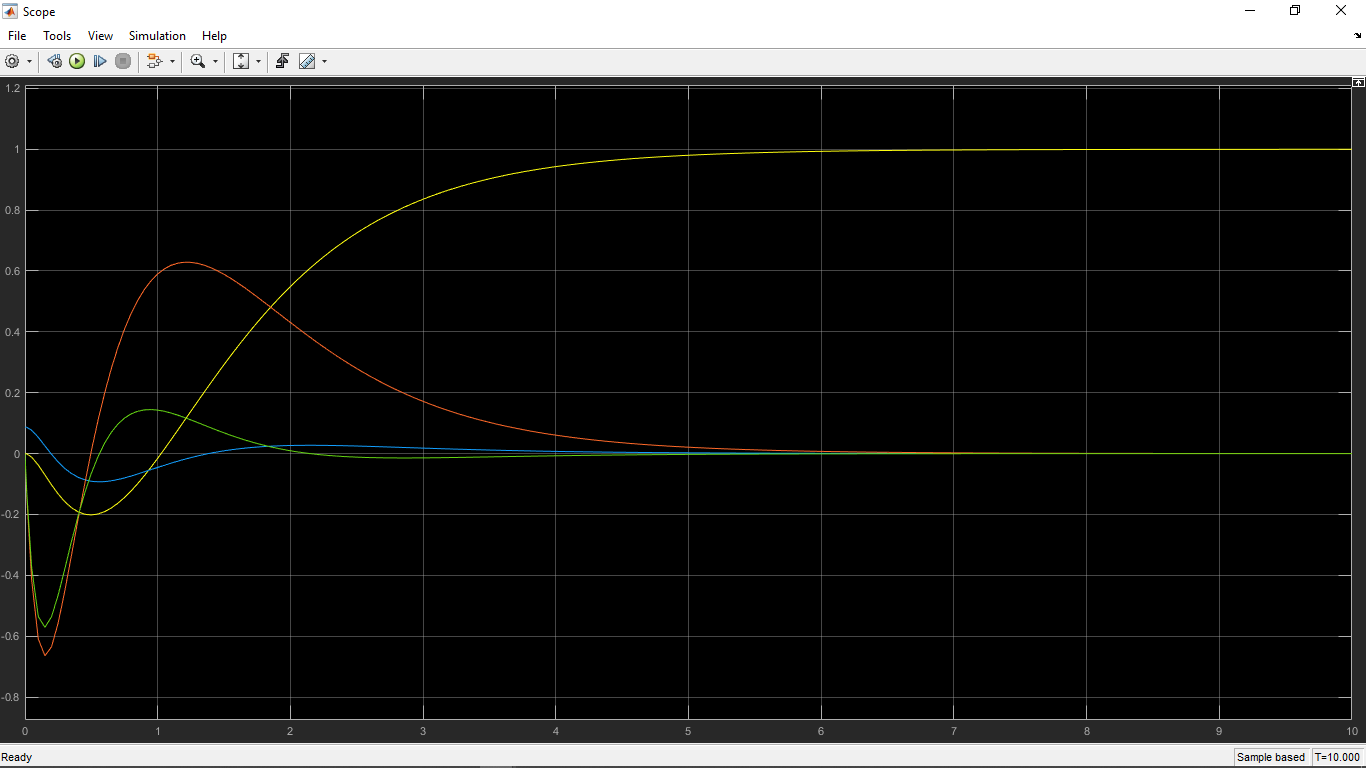


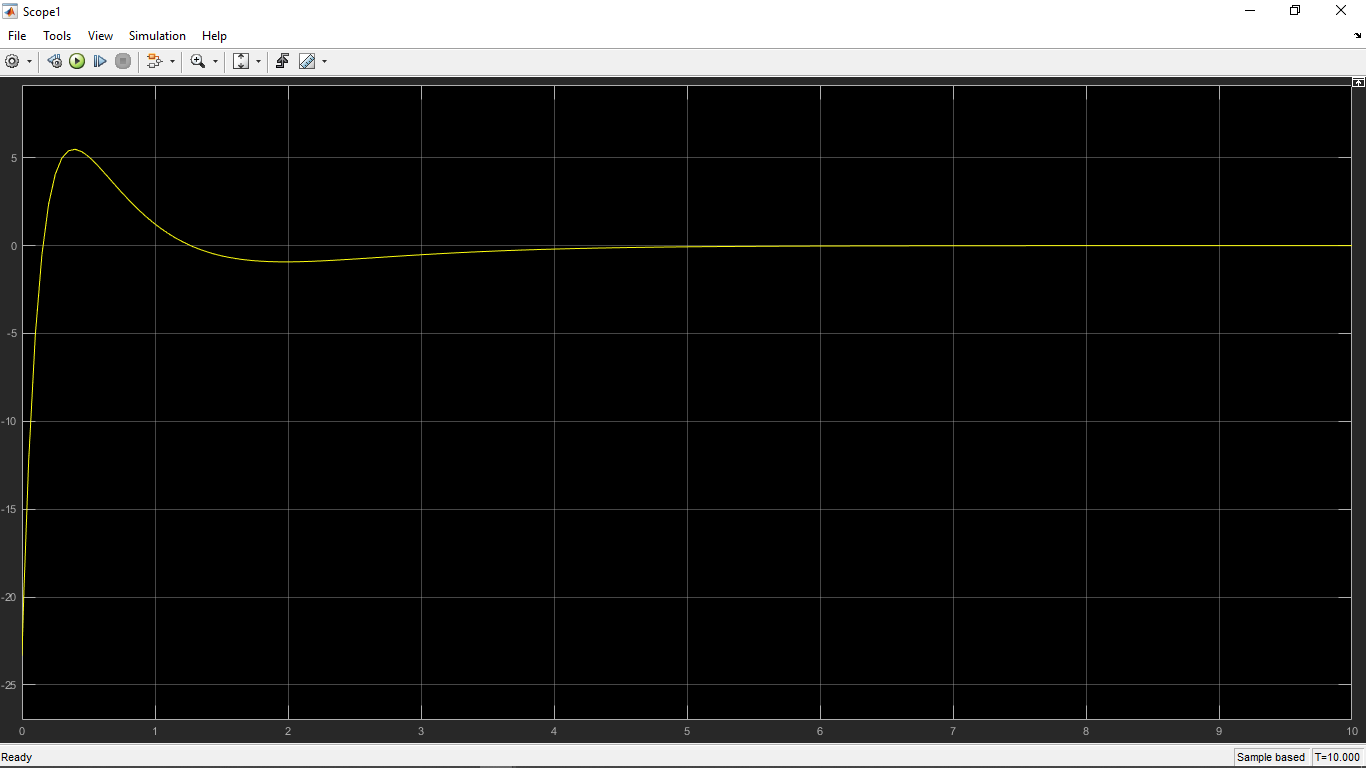
So we now have k but for LQR we have (using MATLAB command):



Now have to give an initial condition of 5 degrees:

**Step response now:**





The poles and zeros are now on the desired place where they should be maintaining zero degree keeping Segway still and only allowing flex of 5 degree as we given initial condition.

**Discrete time controller:** Now in order to operate our Segway model we going make the continuous into steps of discrete time so we need to also change our state-space system into discrete time to make an observer

Sampling time:

System into discrete time by using MATLAB command:

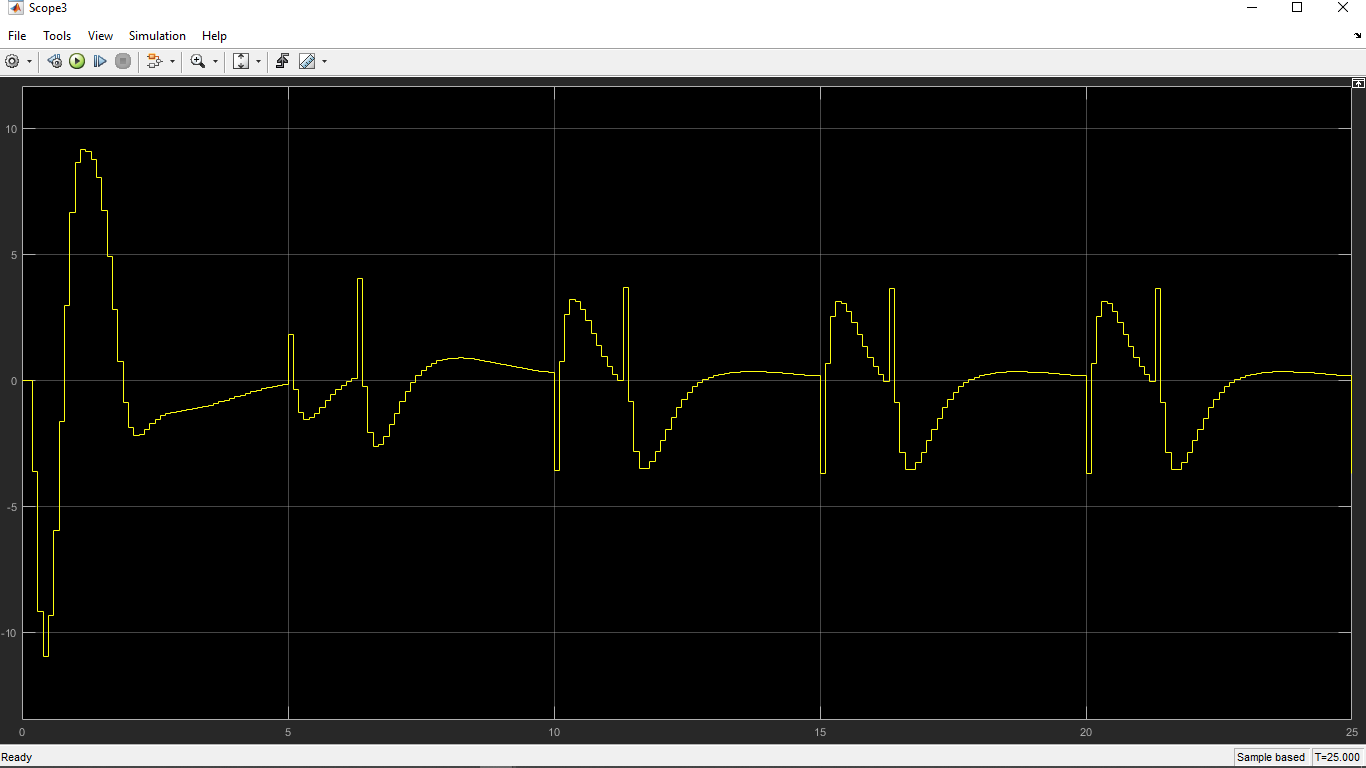
Having LQR values:

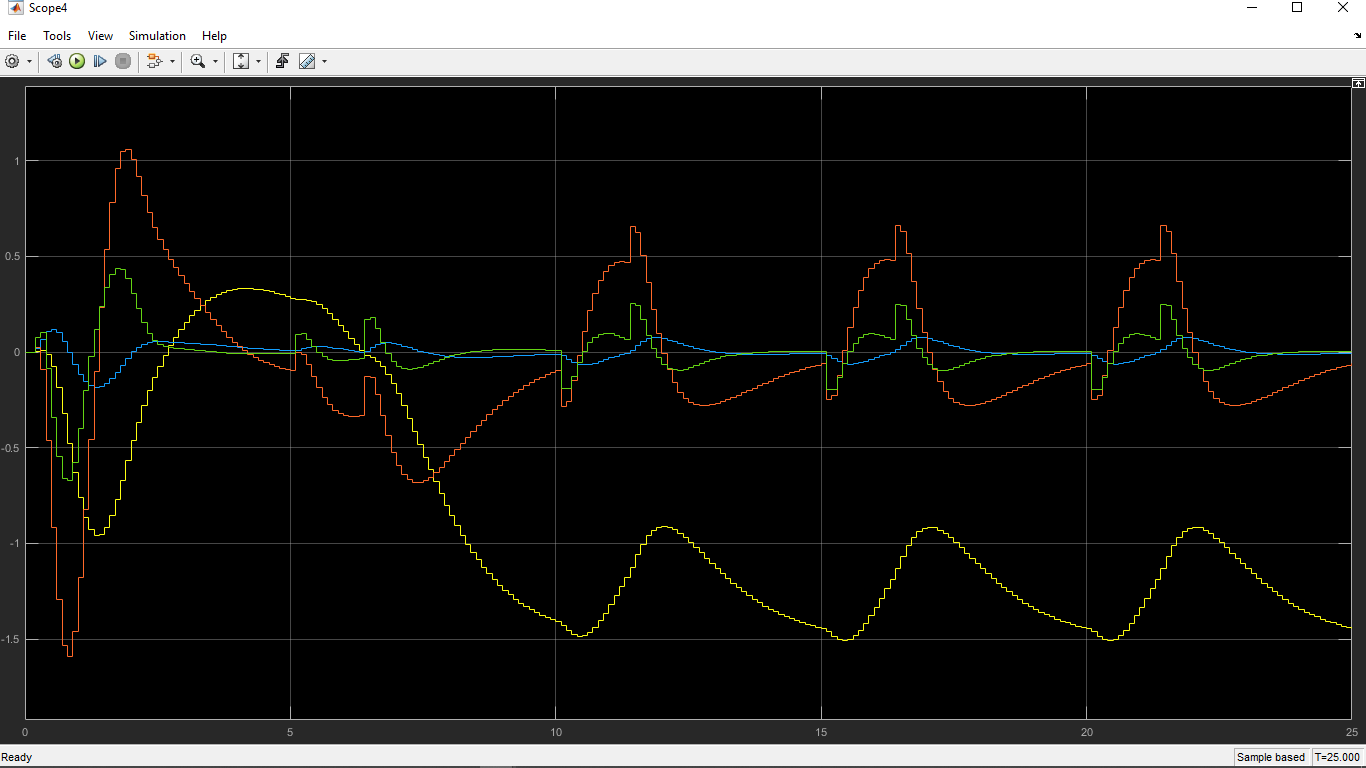
The discrete timeand observer Ob are:

**VII. Step response of the Segway model**

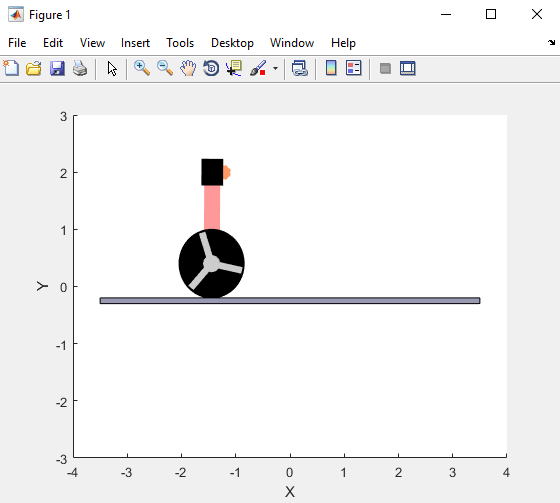
**Discrete time**







You can play this video of simulation





**VIII. CONCLUSION**

Our simulations and analysis were successful enough in achieving its aim to balance a two-wheel robot (Segway) which was completely based on the concept of an inverted pendulum model. However, modelling based on the inverted pendulum shows that this system is unstable without a controller and that there arises the need to implement various controllers in order to control the stability as well as the tilt angle for the controller to stay in the upright condition while moving. The selection of controller was done using the technique of root locus shaping.

As seen in the simulation, software results in the video the designed robot is able to maintain its vertical position by slightly enhancing the wheels but again no doubt further implementation can be done in order to make it perfectly stable with some more better controller and feedback system.

Since, the linear control system developed during our workings, proved to be able to balance the robot under minimal disturbance. But with only software implementation the robustness of the system is not fully tested Therefore, ample amount of time is required in order to give better result and conclusions by performing experiments in order to evaluate the robustness of the system and fine tuning of the control algorithm is required for better performance.

**IX. REFERENCES**

1. [*https://journals.sagepub.com/doi/pdf/10.1177/1729881418770865*](https://journals.sagepub.com/doi/pdf/10.1177/1729881418770865)
2. [*https://www.youtube.com/watch?v=1U6y\_68CjeY*](https://www.youtube.com/watch?v=1U6y_68CjeY)
3. [*https://www.youtube.com/watch?v=u4cnGvygFPI*](https://www.youtube.com/watch?v=u4cnGvygFPI)
4. [*https://cdn.hackaday.io/files/16098688736832/GafarA\_fin al.pdf*](https://cdn.hackaday.io/files/16098688736832/GafarA_fin%20al.pdf)
5. [*https://robotics.ee.uwa.edu.au/theses/2003-Balance-Ooi.pdf*](https://robotics.ee.uwa.edu.au/theses/2003-Balance-Ooi.pdf)
6. [*https://core.ac.uk/download/pdf/11039073.pdf*](https://core.ac.uk/download/pdf/11039073.pdf)
7. [*https://www.youtube.com/watch?v=bJM9jU-P\_H0*](https://www.youtube.com/watch?v=bJM9jU-P_H0)
8. [*https://kth.diva-portal.org/smash/get/diva2:916184/FULLTEXT01.pdf*](https://kth.diva-portal.org/smash/get/diva2:916184/FULLTEXT01.pdf)