University of Science and Technology in Zewail City

CIE 318

Air levitation Control System

June, 2022

Final Report

Jihad Mahmoud - 201902204
Noureldin Mohamed - 201900311
Omar El-Sakka - 201900773
Shireen Dawood - 201800659
Youssef Elshabrawy - 201900667

Contents

1 Introduction

Air levitation is the process by which an object is lifted without mechanical support in a stable position, by providing an upward force that counteracts the gravitational force exerted on the object.
Our motivation for choosing this project was to implement a controlled, simple system physically, but inclusive so that we can apply the studied concepts to it easily, hence it can be easily analyzed.
This report will cover a survey of a similar project procedures that was done before, then a detailed functional and schematic diagrams, after that a list of all the components and their prices will be covered.

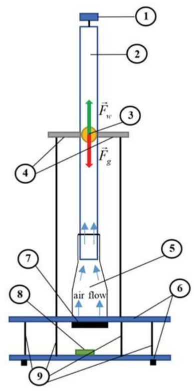

Figure 1: Air levitation

2 Survey

Air levitating control system is mainly a system that controls the position of a ball vertically inside a pipe by increasing or decreasing the speed of a fan just below the pipe, so we can add an upward force that is equal to the gravitational force in amount but in the opposite direction. Thus, the total applied force is zero, and the ball would stay in its position. Although we could levitate balls using several phenomena such as magnetic repulsion, sound waves, and viscous liquids, we would use the air stream to levitate the ball because it is a suitable method for our experiment. In such an experiment, we are in a deep need for:

1. Ultrasonic sensor that could measure the distance.
2. Controller (Arduino Uno)
3. Fan and motor
4. Power supply

3 Functional Block Diagram

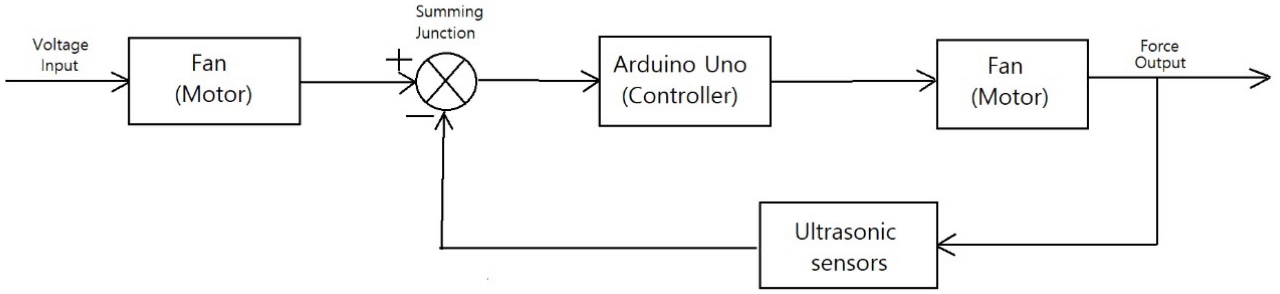

Figure 2: Functional Block Diagram

4 Schematic Diagram

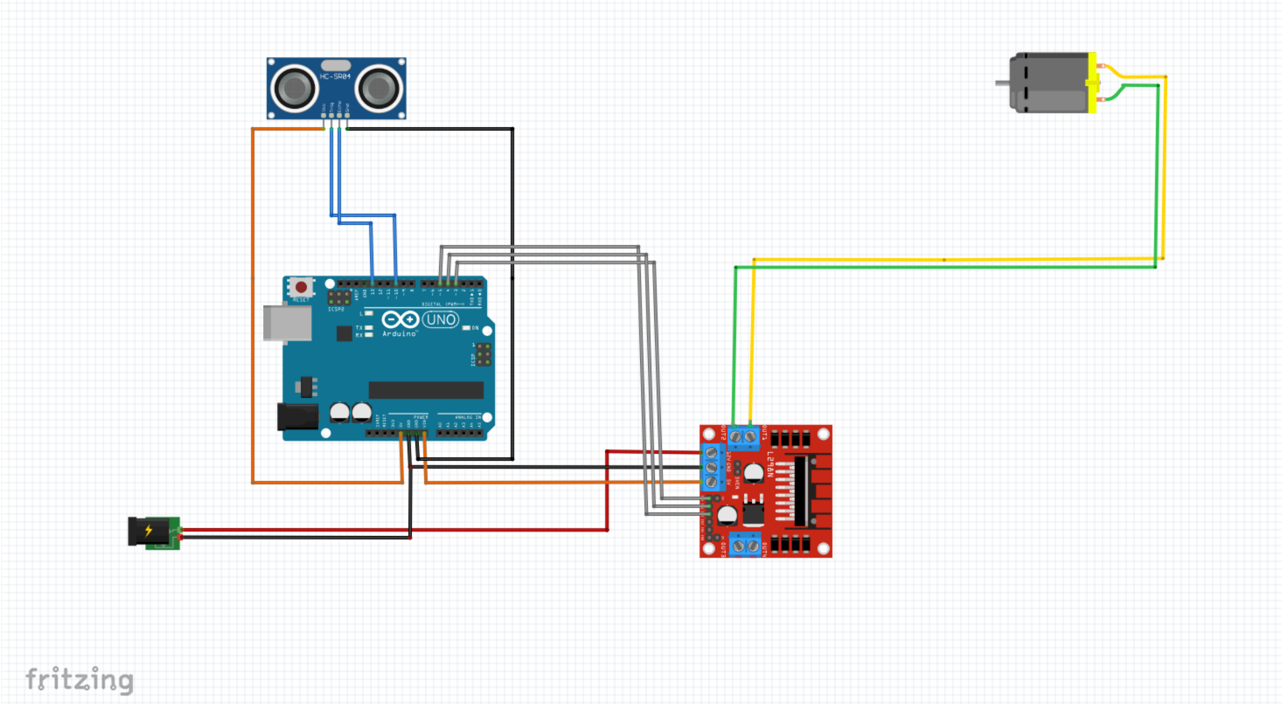

Figure 3: Schematic Diagram without LCD

5 Components and Prices

In this section we will introduce a detailed list of the used components, their usage, and their price.

5.1 Breadboard

Will be used to gather all the components needed in one place without the need to use a soldering iron.
Price: 30 EGP [1]

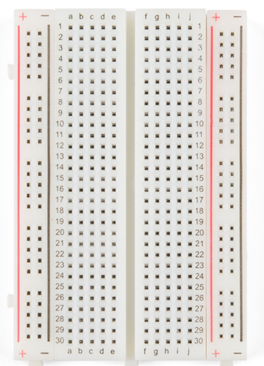

Figure 4: Breadboard

5.2 24V DC Fan size 8x8 cm

The schematic representation of the motor carrying the fan -or the fan itself- used to generate the force affecting the ball, and its speed -the force/air it is producing- is the manipulated variable.

Price: 50 EGP [2]

Figure 5: DC Fan

5.3 H-bridge Module

Switches the polarity of a voltage applied to protect the DC Motor"Fan".
Price: 30 EGP [3]

Figure 6: Relay Module

5.4 Ultrasonic Sensor

The one responsible for the feedback, And it was preferred over the infrared sensor as it is more practical to keep tracing of the ball height that just knowing if it is in the targeted height or not, As the direction of the action (to decrease or to increase) will be critical along with the amount of it.Will be used to measure the ball's distance measurement and based on it a decision on the speed of the fan to maintain the ball at a certain level.
Price: 38 EGP [4]

Figure 7: Ultrasonic Sensor

5.5 Arduino Board UNO

The controller of the whole system which gets the feedback from the sensor (The Ultrasonic Sensor), and then decides what action to make and what orders to send. **Price:** 350 EGP [5]

Figure 8: Arduino UNO

5.6 9 volt battery

To power the components on, it will work as our main power supply.
Price: 30 EGP [6]

Figure 9: Battery

5.7 Connecting jumpers

To connect the components on the breadboard.
Price: 35 EGP [7]

Figure 10: Connecting jumpers

5.8 Blue Cardboard

To support the tube, and enclose the circuit components.
Price: 70 EGP [8]

Figure 11: Blue Cardboard

5.9 Glass Open Ended Tube

To act as a container for the ball.
We will implement this part

Figure 12: Glass Tube

6 Experimental Setup

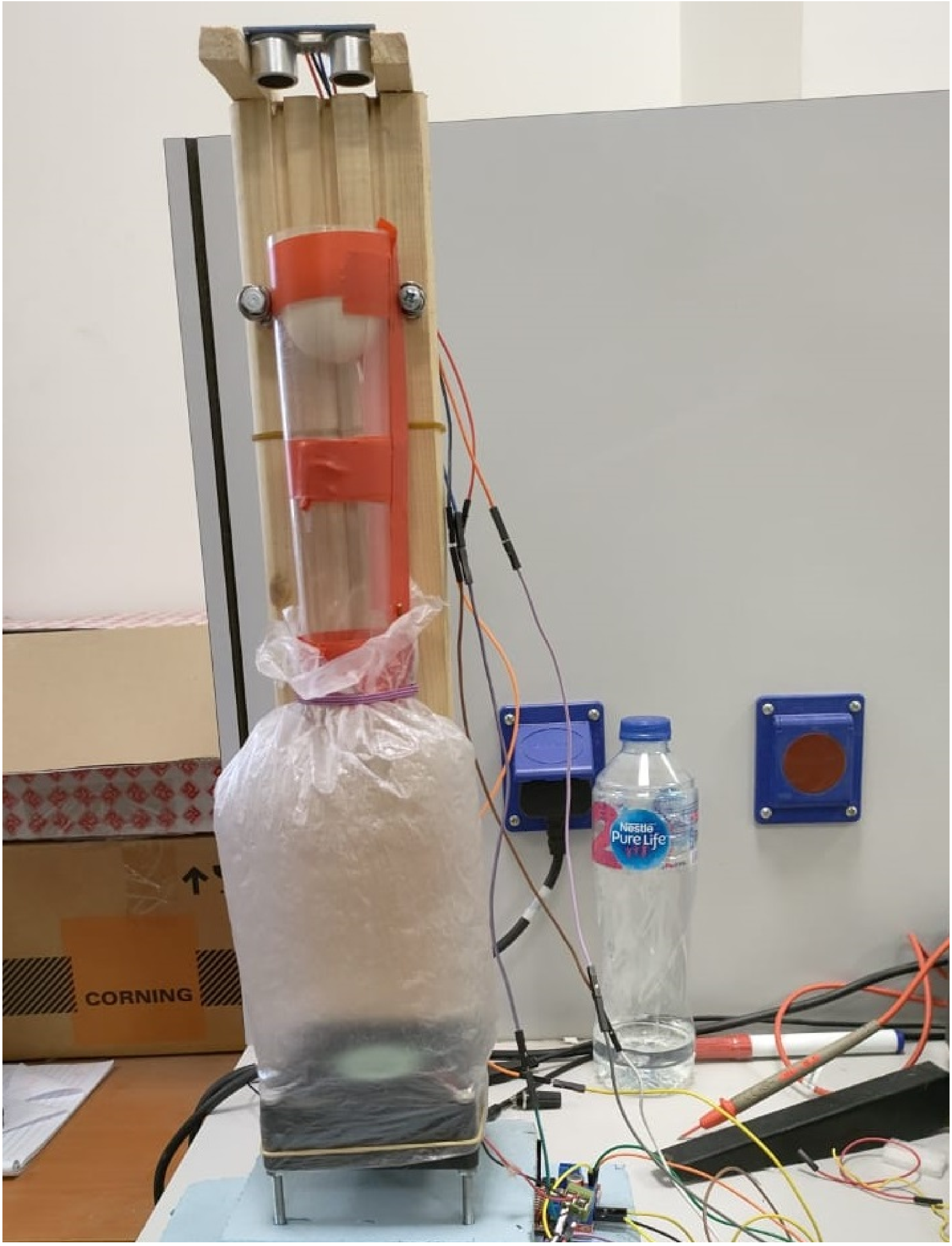

Figure 13: Hardware Implementation

The Figure attached above shows our Air levitation control system design which is built using simple and affordable components. The system consists of an open-ended tube that is fixed vertically. Airflow is forced into the tube from its lower ending by a fan that lifts a lightweight being plastic ball to a controlled height. The system is using airflow for levitating and moving the ball inside the tube by controlling the airflow by changing the speed of the fan. To get a static pressure Airflow able to levitate the plastic being ball a 24 volt

Fan that is fixed to the system is tied with an elastic band to a plastic bag inflated upward that is tied to the opening of the tube to keep the airflow from leaking. The ultrasonic sensor and the tube are fixed to a vertical wooden holder to keep them in a stable position. An H-bridge driver chip is connected to the fan to drive and control it. An ultrasonic sensor is used instead of an infrared sensor to track the ball height. We faced some challenges while working on the setup, and we able to solve them, including: - placing the tube directly on the fan, but the direct airflow was not strong enough to lift the ball. - trying multiple fans different in size, rotation speed and voltage. - we thought that what makes the fan's airflow too weak to levitate the ball is air leakage as the openings in the area between the fan and the tube were not tightly closed, we tried to surround and tie them together using a rubber balloon - The effective solution to this problem was to use a plastic bag and fixing it to the fan and the tube with a rubber band this managed to collect air to make a suitable static pressure airflow forced into the tube to levitate the ball upward. - It was not available to buy a tube with the required system specifications and to fit with the ball diameter, so it is not too wide nor too tight for it to make the ball movement inside it smooth without the ball swaying in its movement or being in need to a stronger fan and airflow to levitate it, so we made a suitable tube using transparent plastic bottles.- we chose a lightweight ball suitable in size to be levitated and controlled easily.

7 Mathematical Model [9]

When high-speed air collides with the bottom of the ball, a high-pressure zone forms beneath it. A low-pressure zone is created when high-pressure air rushes over the curving surface of the ball. Because of its comparatively high pressure, the atmospheric air around this low-pressure zone forces the ball back to center of the stream if it tends to wander away from it. As a result, the ball's lateral motion becomes stable in the middle of the flow. Because gravity and air drag are balanced, the vertical motion of the ball around this equilibrium point is also stable.
The system dynamic equations are nonlinear. This equation makes a crucial prediction about the relationship between the pressure and the velocity of a moving ideal fluid:

$$P_1 + \frac{1}{2}\rho v_1^2 + \rho g y_1 = P_2 + \frac{1}{2}\rho v_2^2 + \rho g y_2 \tag{1}$$

where, $\frac{1}{2}\rho v^2$ is kinetic energy and $\rho g y$ is gravitational potential energy, P_1 and P_2 are the static pressures of air at the cross-section, y_1 and y_2 are the different distances between the ball and the bottom of the pipe.
From Newton's second law:

$$m\frac{d^2 y}{dt} = F_b + F_d - F_g \tag{2}$$

$$F_b = \rho g V_b \tag{3}$$

$$F_d = \frac{1}{2}\rho C_d A(v_f - \dot{y})^2 \tag{4}$$

$$F_g = mg \tag{5}$$

V_b is the ball's volume and v_f is the velocity of the air inside the tube, A is the ball's area, C_d is the so-called drag coefficient, and y is the position of the ball in the tube.
By summarizing and reforming the relations mentioned above, the system's dynamic equations can be obtained by exploiting net force between the airflow force and gravitational force as

$$m\Delta\ddot{y} = -mg + \frac{1}{2}\rho C_d A(v_f - \dot{y})^2 + \rho g V_b \tag{6}$$

After approximations like C_d is constant due to the small velocity of flow, the levitating ball will be in a steady state when it does not move ($\ddot{y} = \dot{y} = 0$). We can conclude that the transfer function of the entire system defines as follows:

$$\boxed{G(s) = \frac{y(s)}{u(s)} = \frac{bk_v(1 - T_d s)}{s(s+b)(\tau s+1)(T_d s+1)}} \tag{7}$$

8 System Analysis using MATLAB

8.1 System Identification

Here are the result of system identification using MATLAB

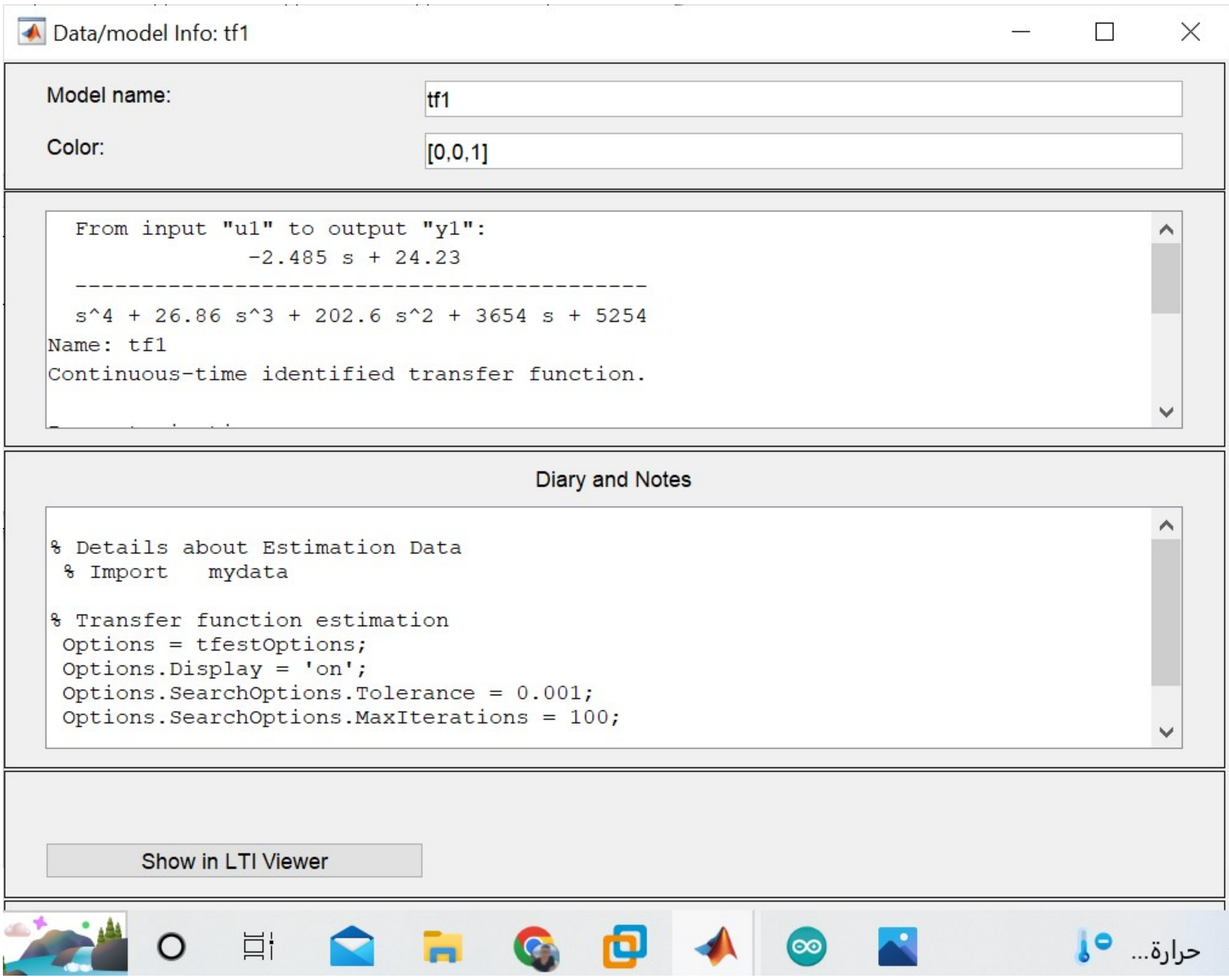

Figure 14: System Identification using MATLAB

8.2 Stability and MSE

That gives a good MSE value and stability, given the number of poles are 4 and number of zeros is 1. And here is figure 6 the figure of system response.

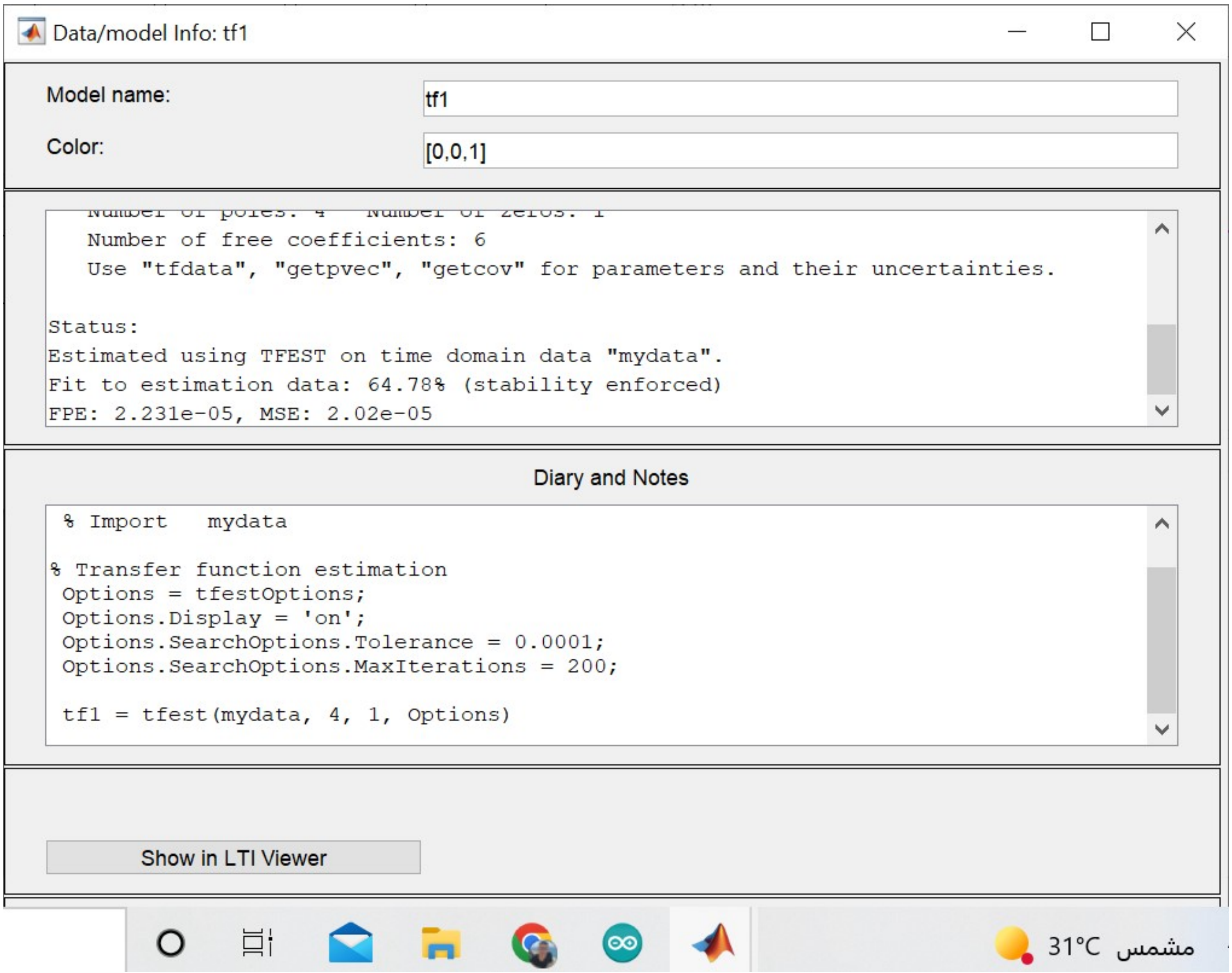

Figure 15: Stability & MSE

8.3 System Response

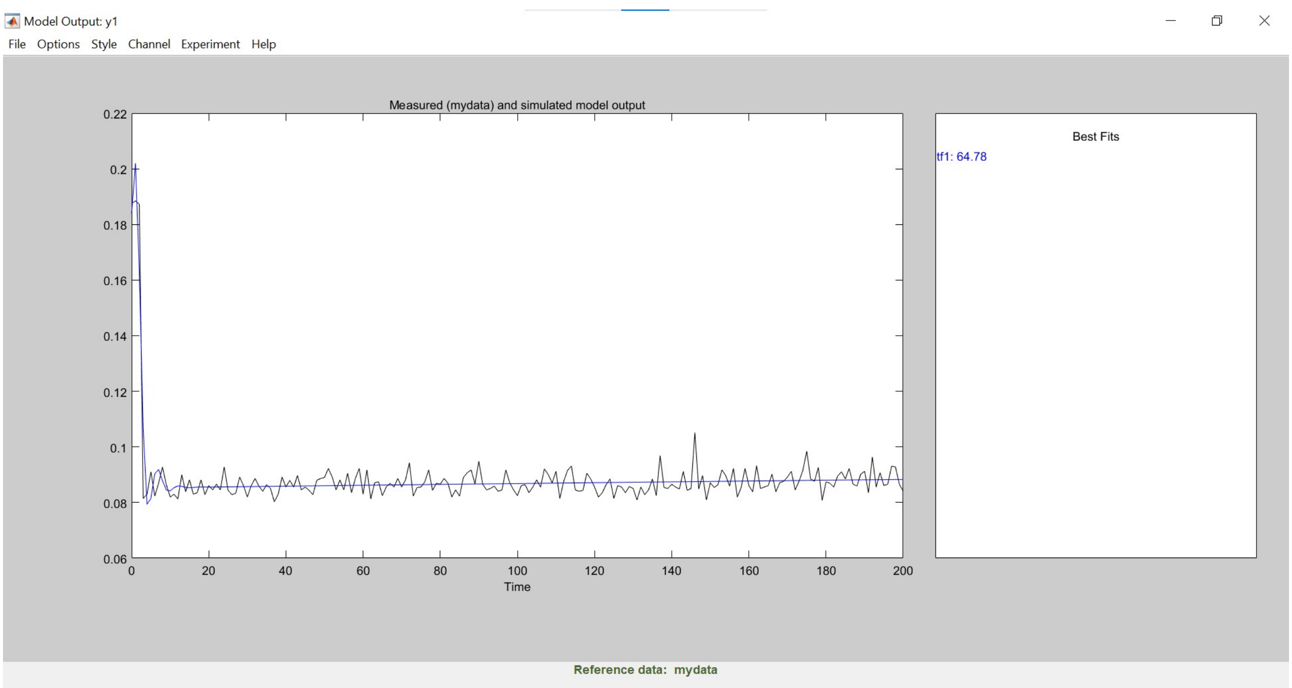

Figure 16: System Response

9 Z-Transform And Tuning using MATLAB

9.1 Tuning using Matlab

9.1.1 Transfer function

```
sys_original =

                -2.485 s + 24.23
  -------------------------------------------
  s^4 + 26.86 s^3 + 202.6 s^2 + 3654 s + 5254
```

Figure 17: PID tune

9.1.2 Continuous to discrete

Once we have a continuous function of the tuned system, the following step is to convert it to discrete using the function c2d.

```
-3.778e-07 z^3 - 9.622e-07 z^2 + 1.205e-06 z + 3.469e-07
--------------------------------------------------------
      z^4 - 3.745 z^3 + 5.258 z^2 - 3.277 z + 0.7644
```

Figure 18: Continuous to discrete

9.1.3 Tuning the Discrete System

Now that we have converted this lets tune one more time to improve our accuracy.

```
g =

             Ts             z-1
  Kp + Ki * ------ + Kd * ------
             z-1             Ts

  with Kp = 90.9, Ki = 314, Kd = 0.447, Ts = 0.01
```

Figure 19: Continuous to discrete pid tune

9.1.4 Standard Form

Lets put it in the standard form through using the function pidstd.

```
             1          1
Kp * (1 + ---- * --- + Td * s)
            Ti          s

with Kp = 90.9, Ti = 0.289, Td = 0.00491
```

Figure 20: PID in Standard form

9.2 Tuning using Simulink

9.2.1 PID Tune

The following step is to use the function pidtune to tune our controller given the transfer function obtained from the system identification. We prefered to take those result instead of the command due to facility of optimizing the response and transient time.

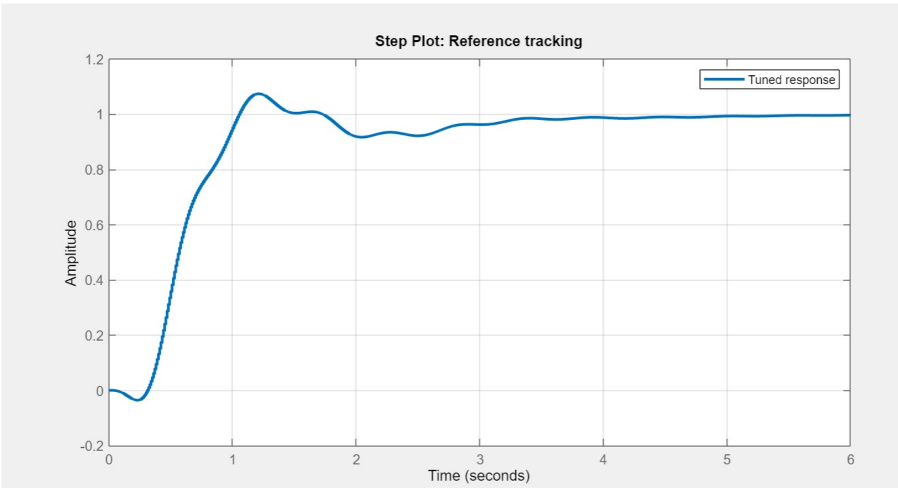

Figure 21: PID Manual Tuning using Simulink

9.2.2 PID Values

This good response can be obtained by those values of P, I, and D.

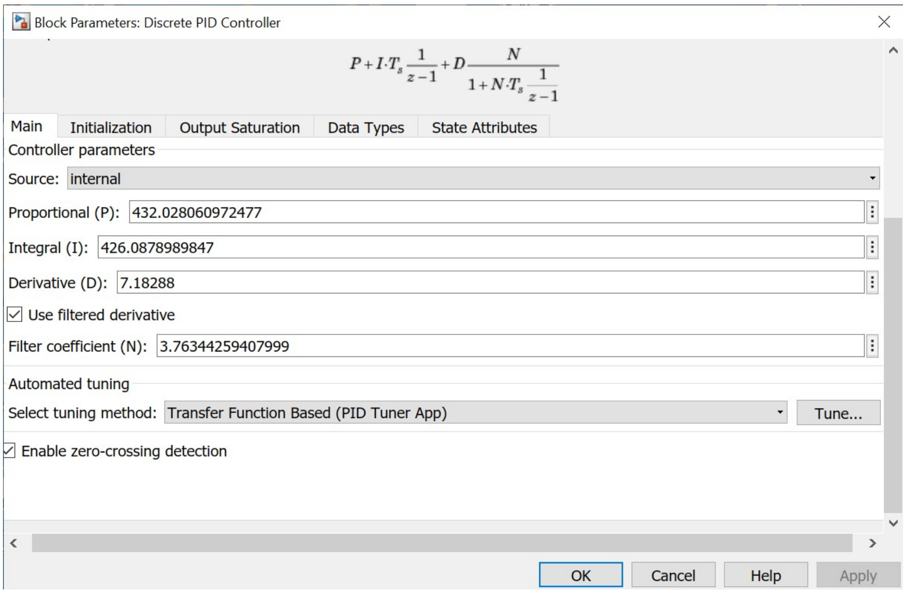

Figure 22: PID Values

These numbers will be used in the Controller code on Arduino.

9.3 Obtaining the controller output functions

We now can obtain the values for K_P, T_I, T_D and we will be using a sampling time of 0.01s.//

Those values would be used by the controller to compute the output values to produce from the errors. Lets begin by computing the proportional section output from the error obtained at the moment.

$$w_k = K_p * e_k \tag{8}$$

For the integral section we will use the current error, in addition to the previous value of the proportional output which is initially set to zero.

$$w_k = \alpha e_k + p_{k-1} \tag{9}$$

Alpha is a ratio computed to take a value of the current error and it to the output.

$$\alpha = K_p * \frac{T_s}{T_i} \tag{10}$$

Now lets jump to the derivative section which totally depends on the errors, one is the current error, and one is the previous error, which is initially set to 0.

$$q_k = \beta * (e_k - e_{k-1}) \tag{11}$$

β is a ratio taken which depends on vales obtained from the controller.

$$\beta = K_p * \frac{T_D}{T_s} \tag{12}$$

- $K_P = 432.028060972477$
- $\alpha = 4.260878989847$
- $\beta = 718.228$

10 ISR

Now that we designed out controller and obtained our values of k_p,$\alpha$and β to start writing down our ISR.

We will begin by setting our reference value first, followed by computing the error, which is simply the desired value - our actual value produced from the ultrasonic sensor. We will calculate w_k,q_k,and p_k values and add them all together to get the controller output. Once we have gathered our output in voltage, it is time to convert it into a digital signal through pulse width modulation in order to send it to the enable to control the fan. We limit the val output to 255 which corresponds to the maximum volt, 24 V. Finally we set the previous error to the current error, and the previous differential output to the our current differential output, and start once again.

11 Conclusion

Finally the values are not close to the values obtained on Matlab so we performed manual tuning, the tube is not perfect and is causing slight errors, the Fan is not powerful enough to cover large distances; therefore, we created a small tube. The controller is not working perfectly since the manual tuning was not perfect, but we can observe the fan trying to control the height of the ball through the voltage sent to it by the H-bridge.

References

[1] *Breadboard 630-Tie Point "BB-01"*. https://ram-e-shop.com/product/bb01-bread-board/. Accessed: 4/24/2022.

[2] *2V DC Fan Size 5×5 cm2*. https://ram-e-shop.com/product/fan-12vdc-5010/. Accessed: 4/24/2022.

[3] *Relay Module 1 Channel (12Vdc) Pins*. https://free-electronic.com/product/relay-module-1-channel-12vdc-pins/. Accessed: 4/24/2022.

[4] *Ultrasonic Sensor Distance Measurement Module HC-SR04*. https://ram-e-shop.com/product/kit-ultrasonic-txrx/. Accessed: 4/24/2022.

[5] *Arduino Board – Arduino UNO Rev3*. https://ram-e-shop.com/product/kit-arduino-uno/. Accessed: 4/24/2022.

[6] *Battery 9V (PHILIPS) LongLife*. https://ram-e-shop.com/product/battery-9v-philips-longlife/. Accessed: 4/24/2022.

[7] *Connecting Jumper Wires for Breadboard (140 Wire) 22AWG*. https://ram-e-shop.com/product/bb-j100/. Accessed: 4/24/2022.

[8] *Plywood and melamine board prices in Egypt 2022*. https://www.as3arak.com/2019/02/Prices-Panels-Wood-Abelkash.html. Accessed: 4/24/2022.

[9] Amirreza Tootchi, Saeed Amirkhani, and Ali Chaibakhsh. "Modeling and Control of an Air Levitation Ball and Pipe Laboratory Setup". In: *2019 7th International Conference on Robotics and Mechatronics (ICRoM)*. IEEE. 2019, pp. 29–34.