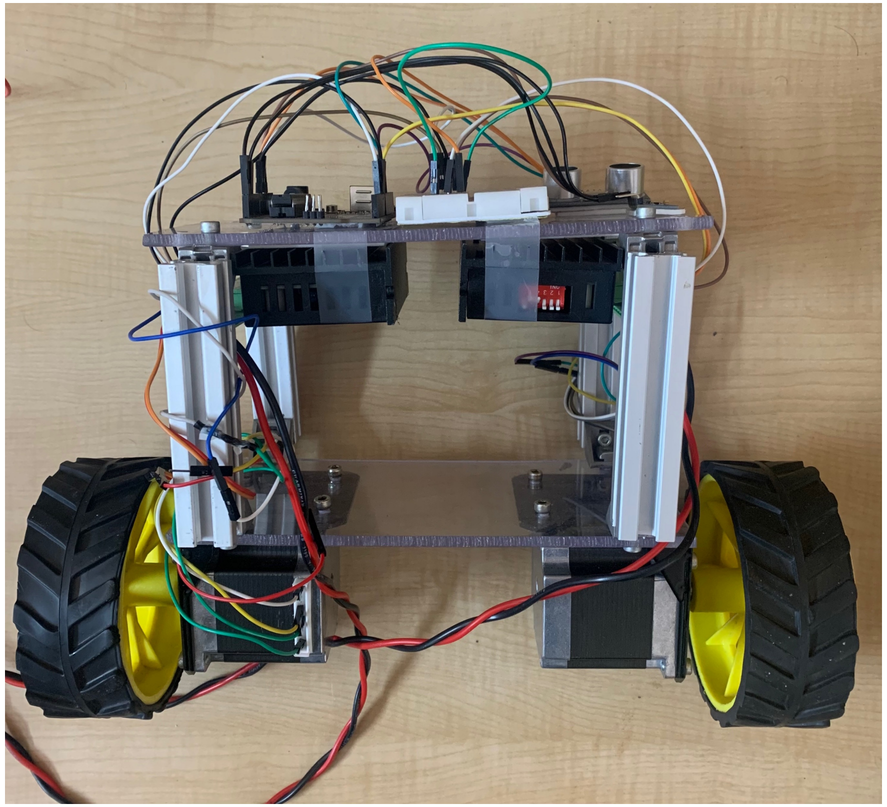
BALANCE-BOT

REUBEN MATTHEW

INTRODUCTION

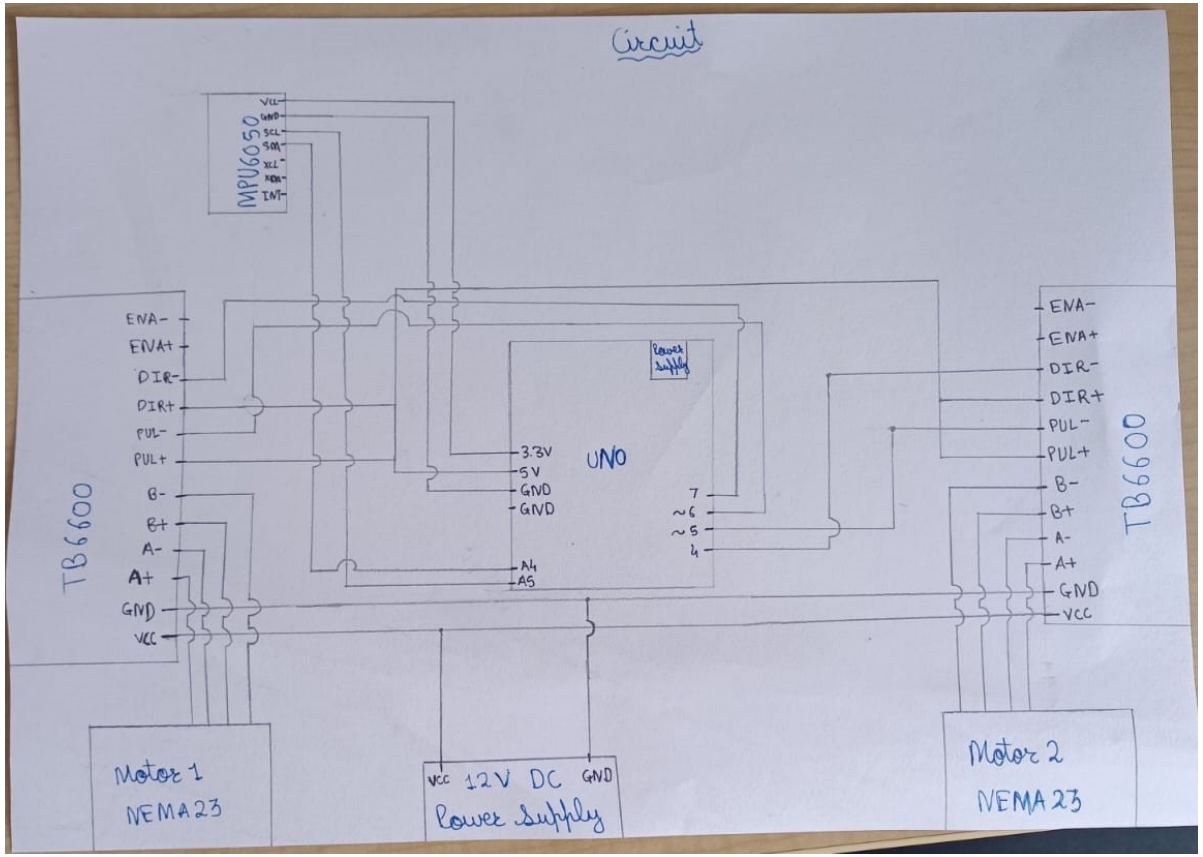
- The self balancing bot is an inverted pendulum on two wheels.
- It is an example of the "cart and pole" system.
- Such a system is unstable and requires a control mechanism to maintain equilibrium.



COMPONENTS REQUIRED

- ARDUINO UNO R3
- MPU6050 gyroscope and accelerometer
- TB6600 stepper motor driver x2
- Stepper motors x2
- Ultrasonic distance sensor
- 100 mm wheels x2
- Breadboard, Jumper wires
- Aluminum extrusions, Acrylic plates.

CIRCUIT DIAGRAM

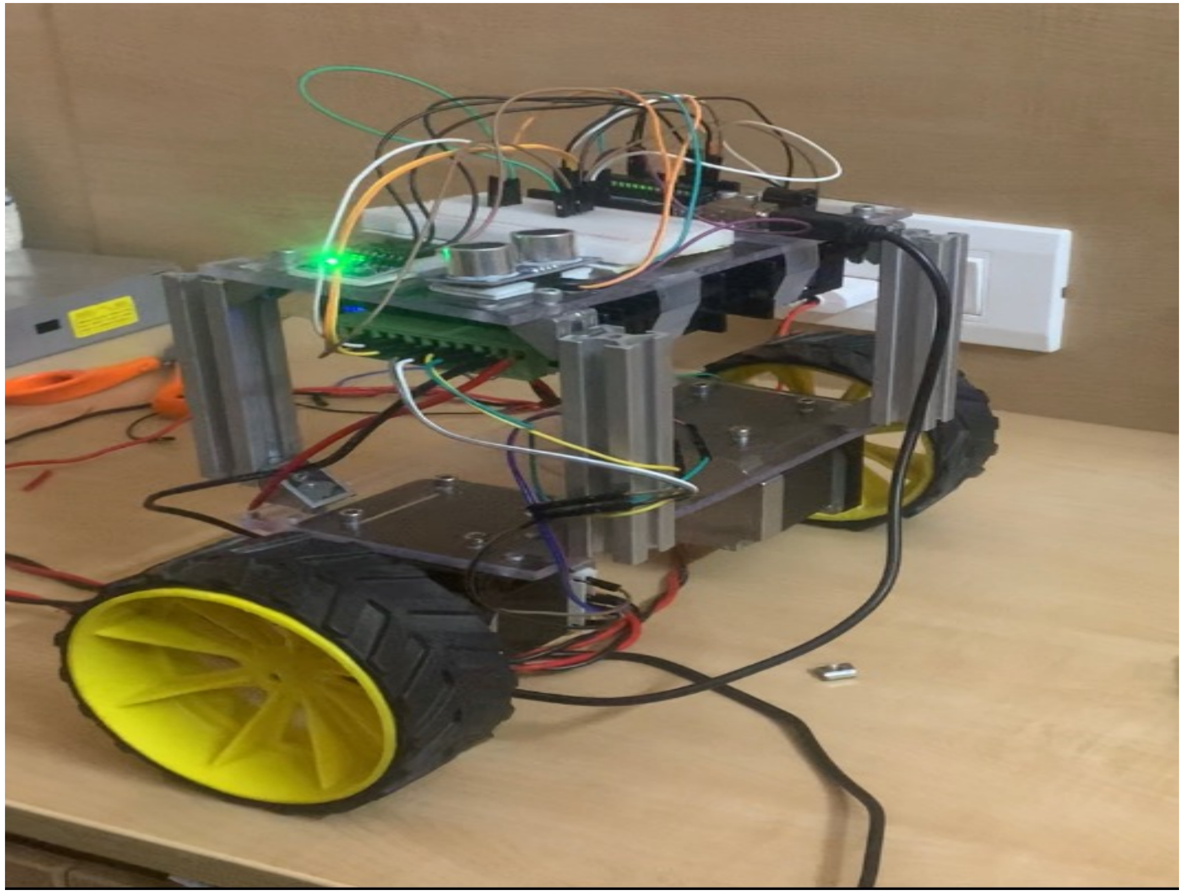


CALCULATIONS

Motor Torque :

- T = m*g*l*sin(t)
- = 2.9*9.8*0.1*sin(5)
- = 0.25 Nm or 2.5 Kg-cm

ASSEMBLY



WORKING PRINCIPLES

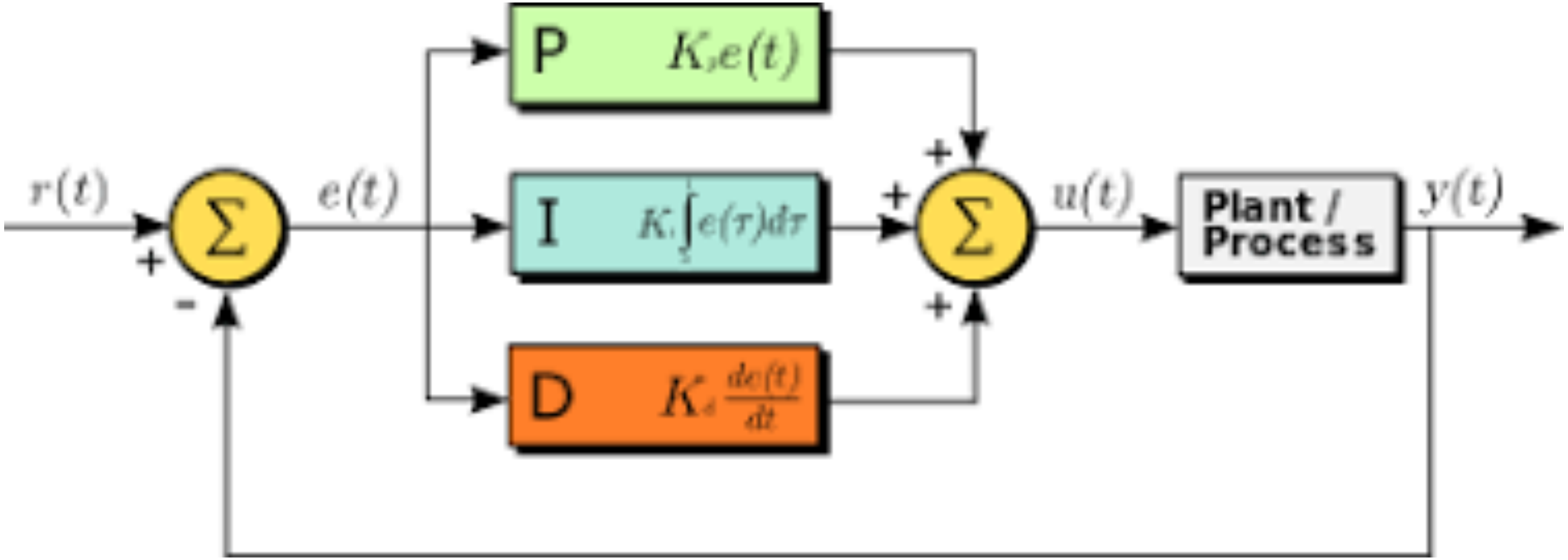
- The Arduino polls for MPU data.
- The angle is calculated individually using the accelerometer and gyroscope data.
- The angles are combined using a complimentary filter in a process called sensor fusion.
- A timer interrupt is set up to execute an ISR every 5 milliseconds.
- The ISR includes calculating the output signal using the PID control algorithm.
- The main program loop executes a function to actuate the stepper motors using the calculated output signal.

COMPLIMENTARY FILTER

$$\text{currentAngle} = \underbrace{\alpha.(\text{previousAngle} + \text{gyroAngle})}_{\text{HPF}} + \underbrace{(1-\alpha).(\text{accAngle})}_{\text{LPF}}$$

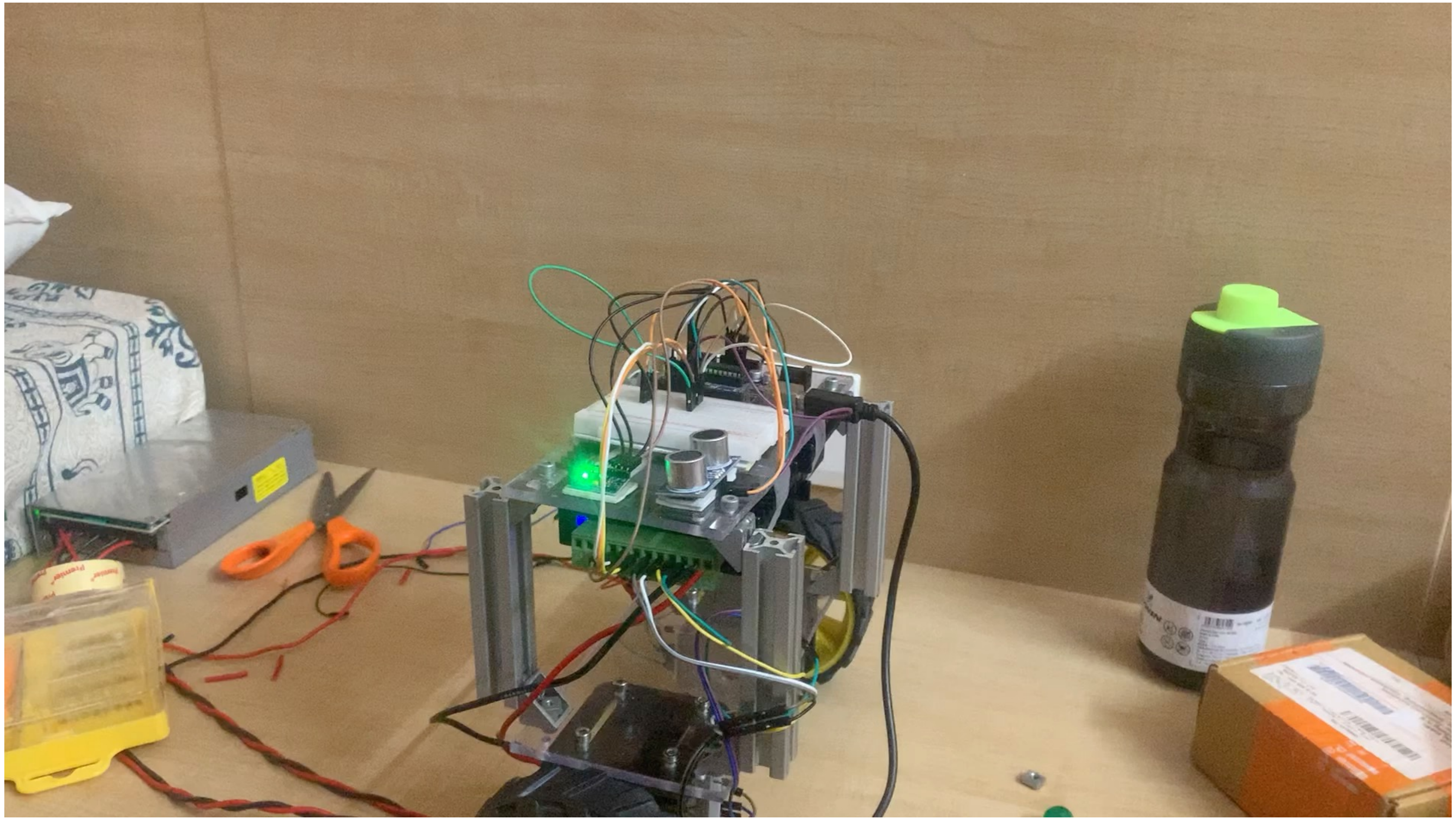
$$\boxed{\alpha = \frac{\tau}{\tau + dt} = \frac{0.75}{0.75 + 0.005} = 0.9934}$$

PID ALGORITHM



DEMONSTRATION

DEMONSTRATION



KEY LEARNINGS

- Interfacing different sensors with the Arduino uno.
- Implementing sensor fusion through the complementary filter.
- Using timer interrupts to drive stepper motors at different speeds.
- Understanding and implementing the PID control algorithm.
- Setting PID gains for different systems.

LINKS

- https://scholar.harvard.edu/files/jgafford/files/finalpaper_final_version.pdf

THANK YOU