GROUP 2 (COM)

COIN SORTING AND COUNTING MACHINE

PRESENTED BY: Win Thawdar Aung
Than Thar Min Htet
Thet San Htar
Phone Min Khant
Thant Pyae Sone Htun

BACKGROUND

- Problem: Coin sorting and counting is a very tedious and time-consuming task.
- Solution: A coin-sorting machine to efficiently and accurately sort and count coins.
- Existing machines often sort by size, which can miss counterfeit or foreign coins.
- Our machine uses a weight sensor for additional accuracy.

GROUP 2 (COM)

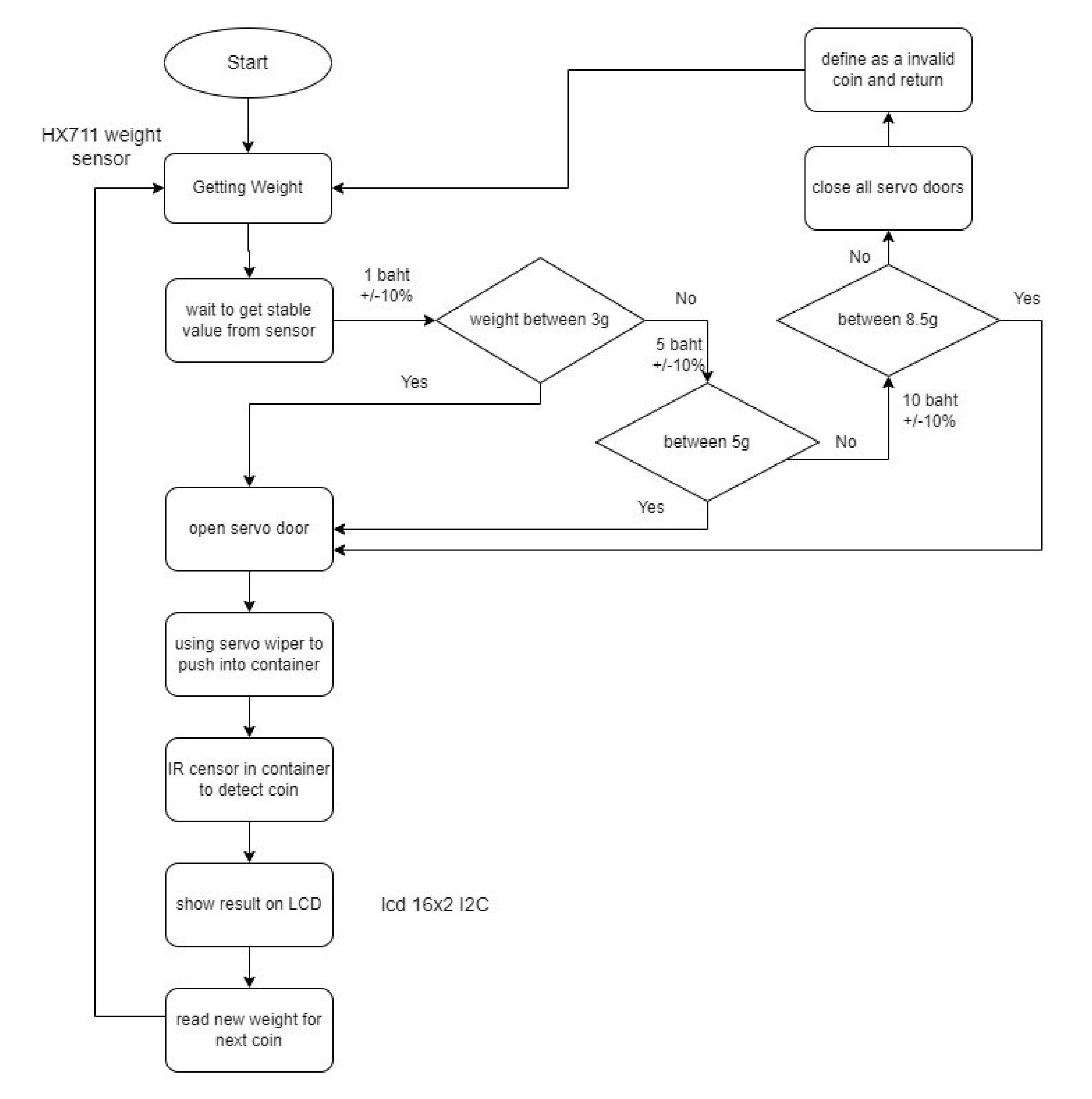
OBJECTIVES

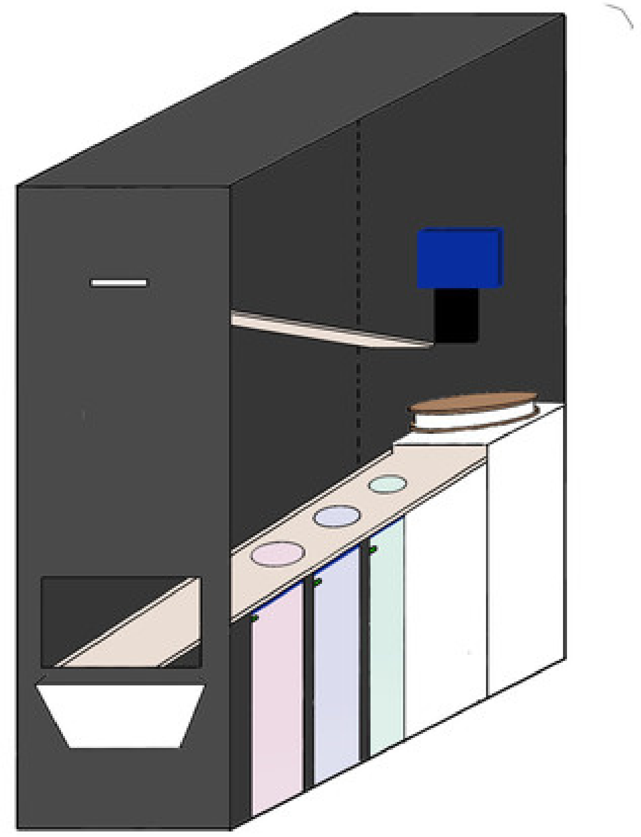
- To develop a machine capable of sorting coins (1 baht, 5baht, 10baht) based on the weight and size , utilizing a weight sensor and an inclined plane with holes matching with their sizes , and equipped with IR sensors to count the coins

Coin Counting and Sorting Machine

OVERALL FLOW

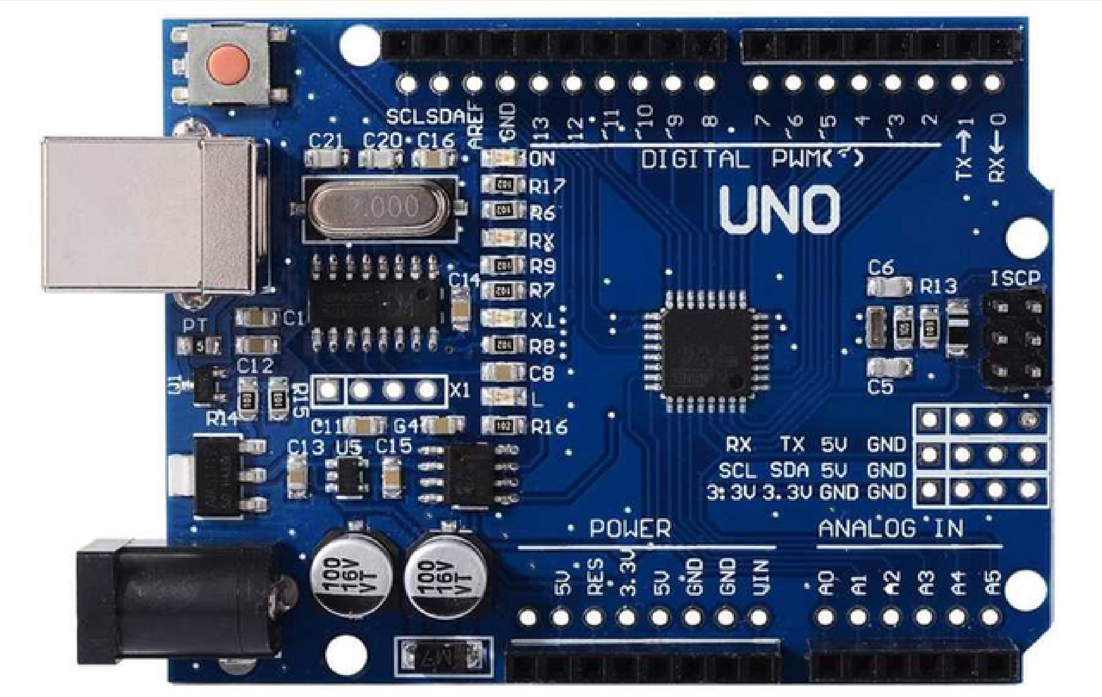
GROUP 2 (COM)





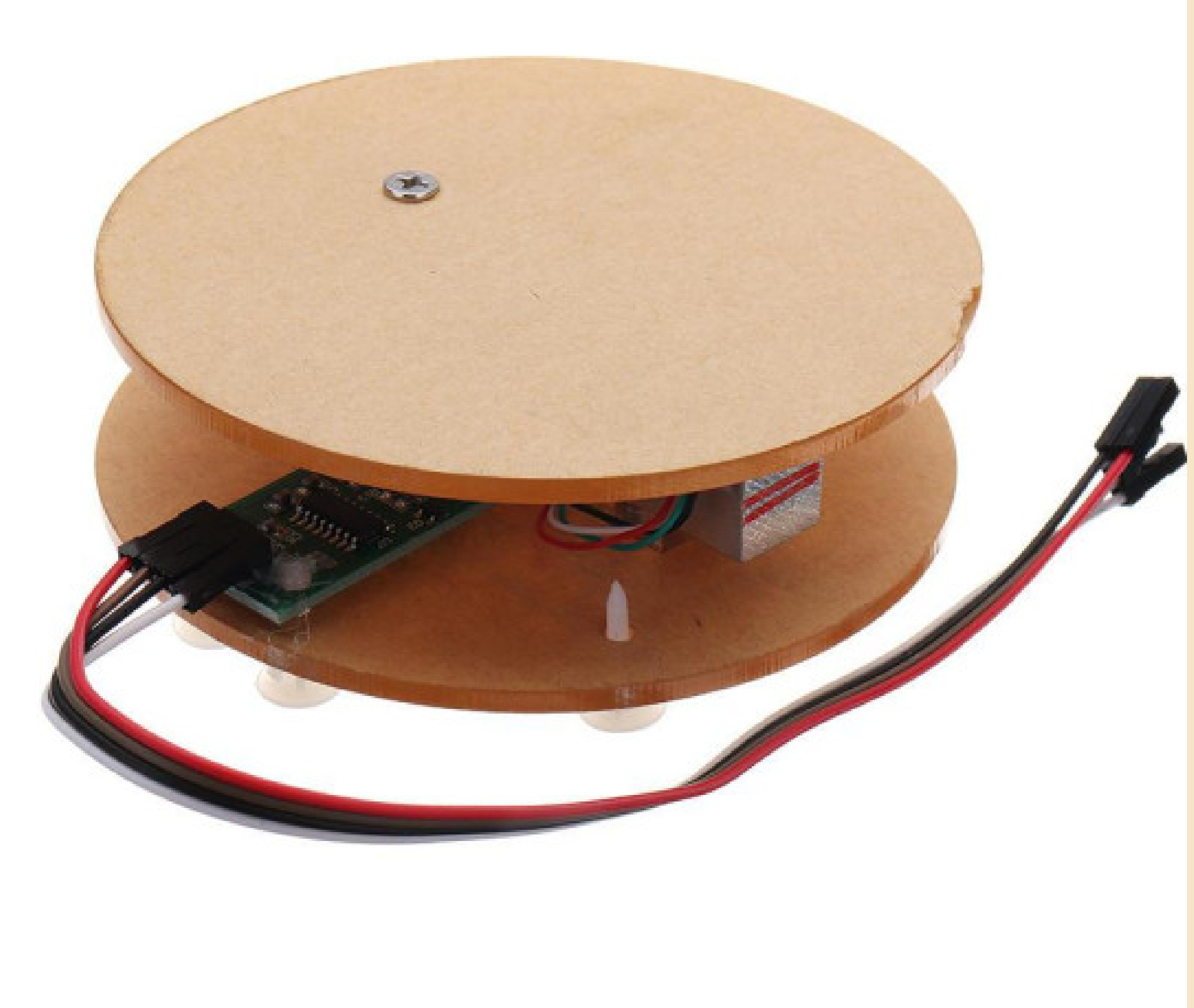
ARDUINO UNO

- This microcontroller acquires data from sensors such as IR and weight sensors and serves as a control system for managing data flow and controlling other modules like servo motors and the LCD.
- Beginner friendly
- Sufficient processing power
- Has more pin inputs compared to Arduino Nano and cost effective.



Pin Inputs

- 8,3,7 - IR SENSORS
- 4, 5 - WEIGHT SENSOR
- 6, 9, 10, 11 - servo motors
- SCL, SDA - LCD

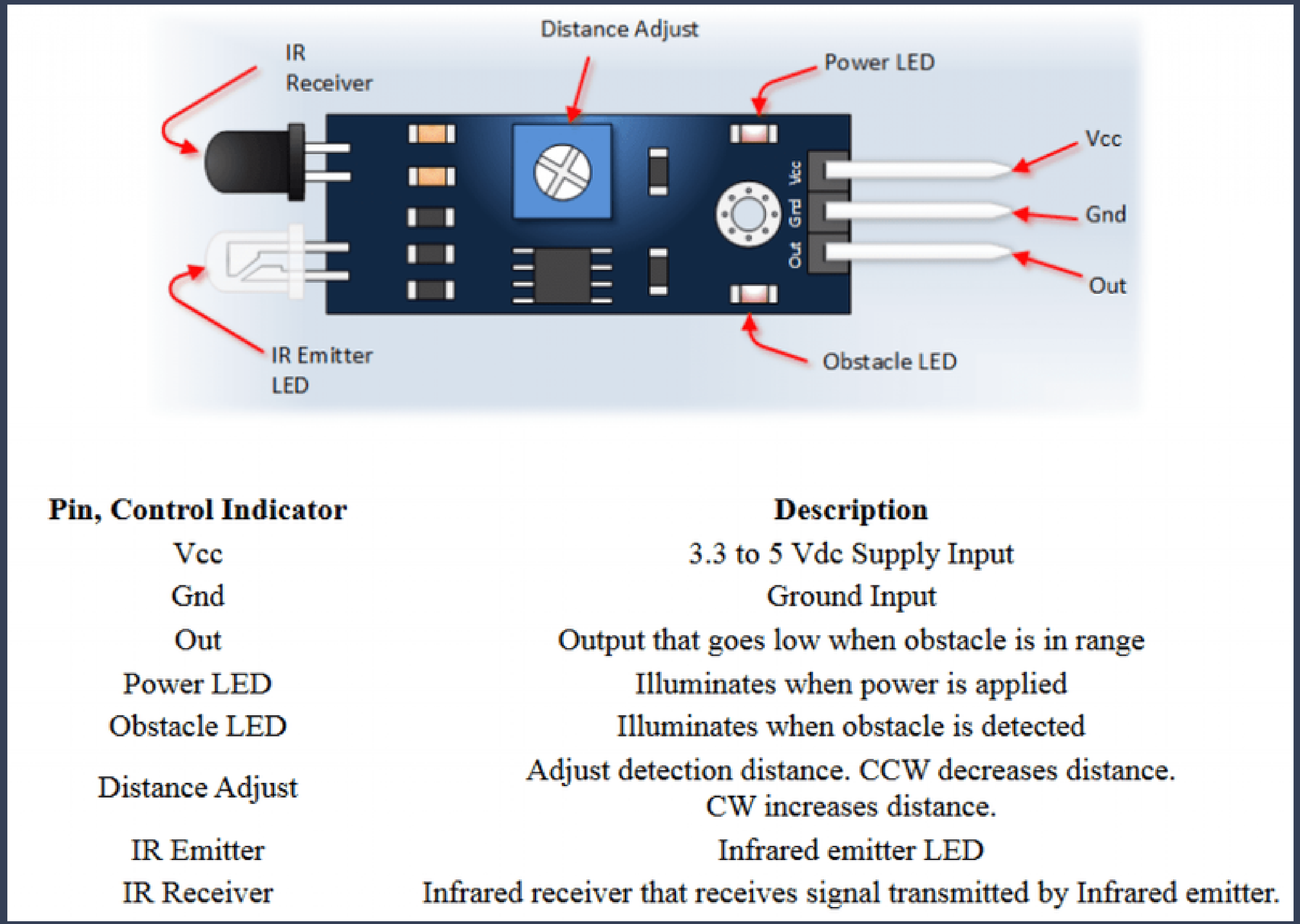


WEIGHT SENSOR

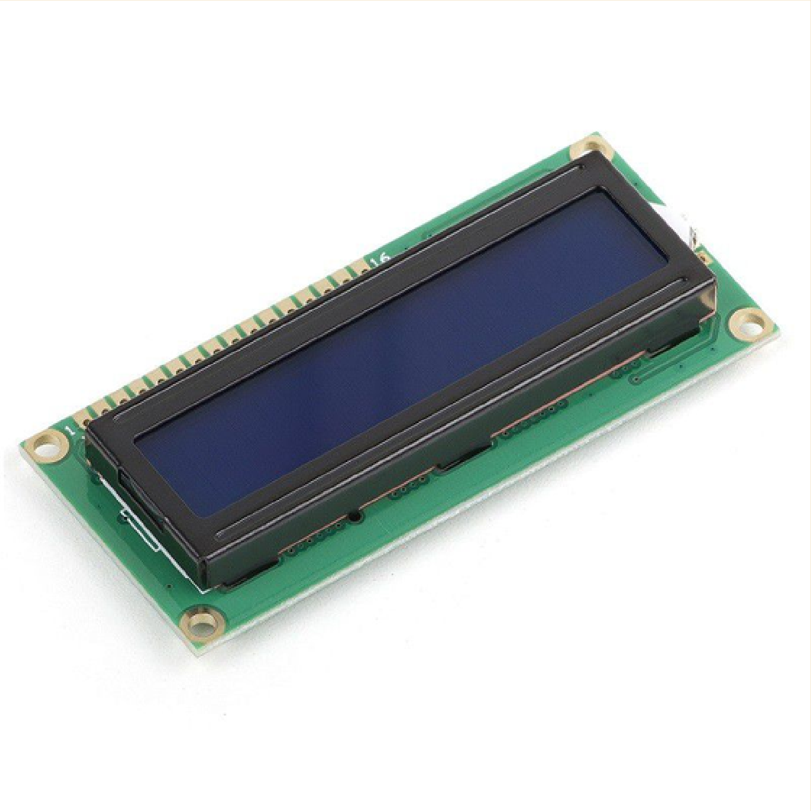
- Differentiate the coin types through their respective weight.
- Utilize High precision HX711 amplifier and 1kg Loadcell to detect small weight stuff like coins
- Easy to use with the Arduino Uno using its built-in analog-to-digital converter.
- Compact size for integration into the coin counting machine design compared to other weight measuring machine.

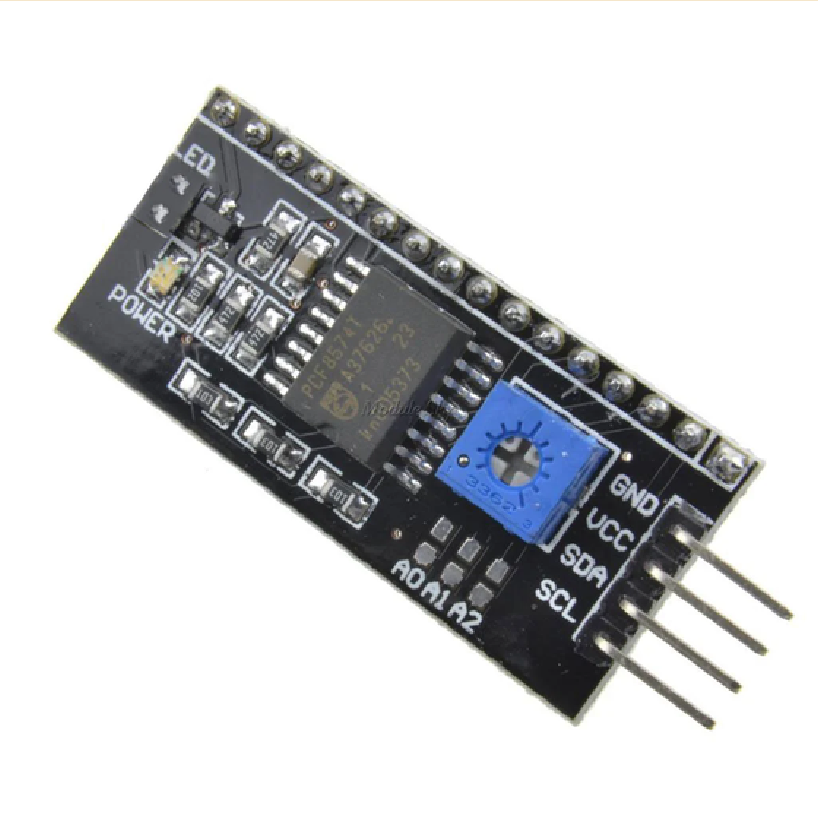
IR SENSOR

- Detect the coin passing and count the coins.
- Non-contact detection of coins to avoid wear and tear
- Adjustable range to adapt to the size of the coins being counted
- Relatively low cost for a cost-effective solution



Pin, Control Indicator	Description
Vcc	3.3 to 5 Vdc Supply Input
Gnd	Ground Input
Out	Output that goes low when obstacle is in range
Power LED	Illuminates when power is applied
Obstacle LED	Illuminates when obstacle is detected
Distance Adjust	Adjust detection distance. CCW decreases distance. CW increases distance.
IR Emitter	Infrared emitter LED
IR Receiver	Infrared receiver that receives signal transmitted by Infrared emitter.





LCD

- Display the amount of 1 baht, 5 baht and 10 baht coins in the containers.
- We used I2C communication with the LCD display to minimize wiring complexity to the Arduino.

SERVO MOTOR

- Used for pushing the coin off the weight sensor and opening and closing of container doors
- Precise Control and easy to implement
- 180 degree motors are enough to operate our desired operations like sweeping and door controlling

TWO STAGE APPROACHES

WEIGHT

The Weight sensor accurately measures the coin's weight and decides which hatch to open. If the coin is in neither of the weight ranges, the hatch will not open

SIZE

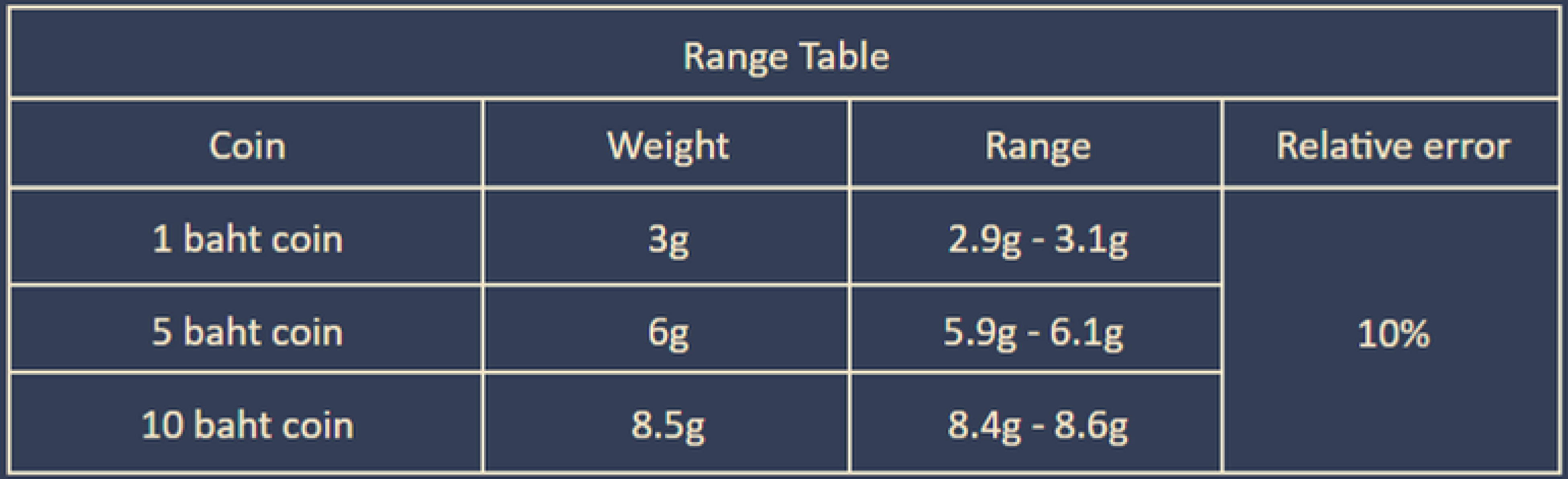
Even if the coin is in the same weight range, it will only go in if the coin has the same size as the designated size and vise versa.

GROUP 2 (COM)

METHODOLOGY

SORTING COINS BY WEIGHT

- When the coin fall on the weight sensor, the system will wait for a stable weight reading.
- If the weight is in range, the respective door will open
- If it is not in range, the coin will be discarded



Range Table

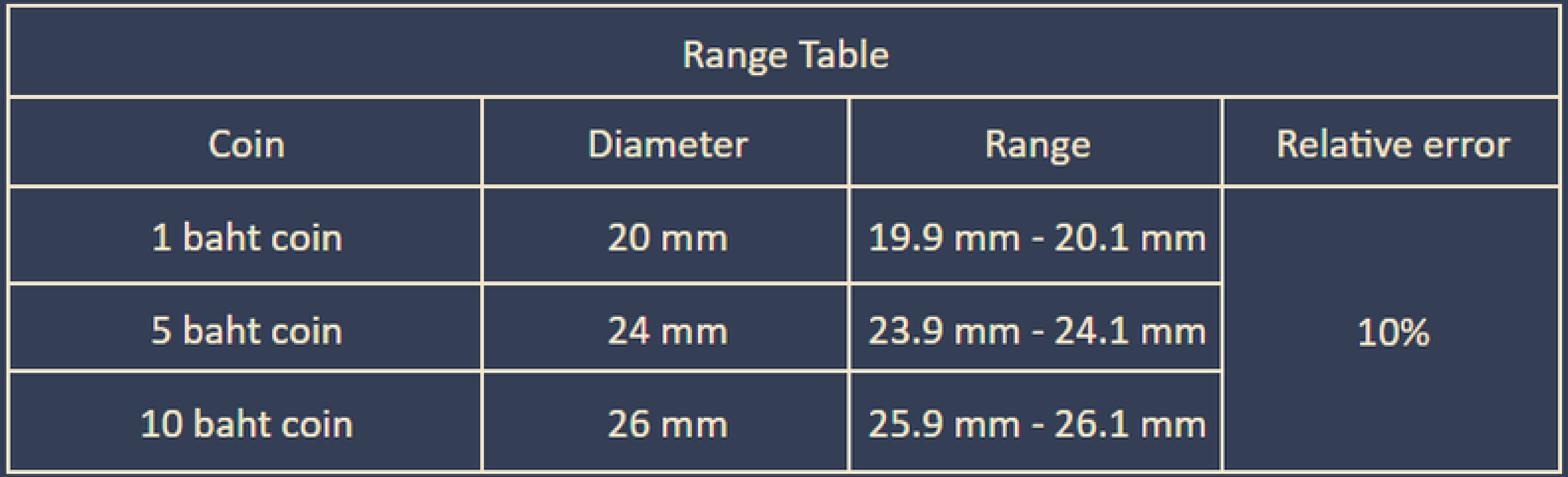
Coin	Weight	Range	Relative error
1 baht coin	3g	2.9g - 3.1g	10%
5 baht coin	6g	5.9g - 6.1g	
10 baht coin	8.5g	8.4g - 8.6g	

GROUP 2 (COM)

METHODOLOGY

SORTING COINS BY SIZE

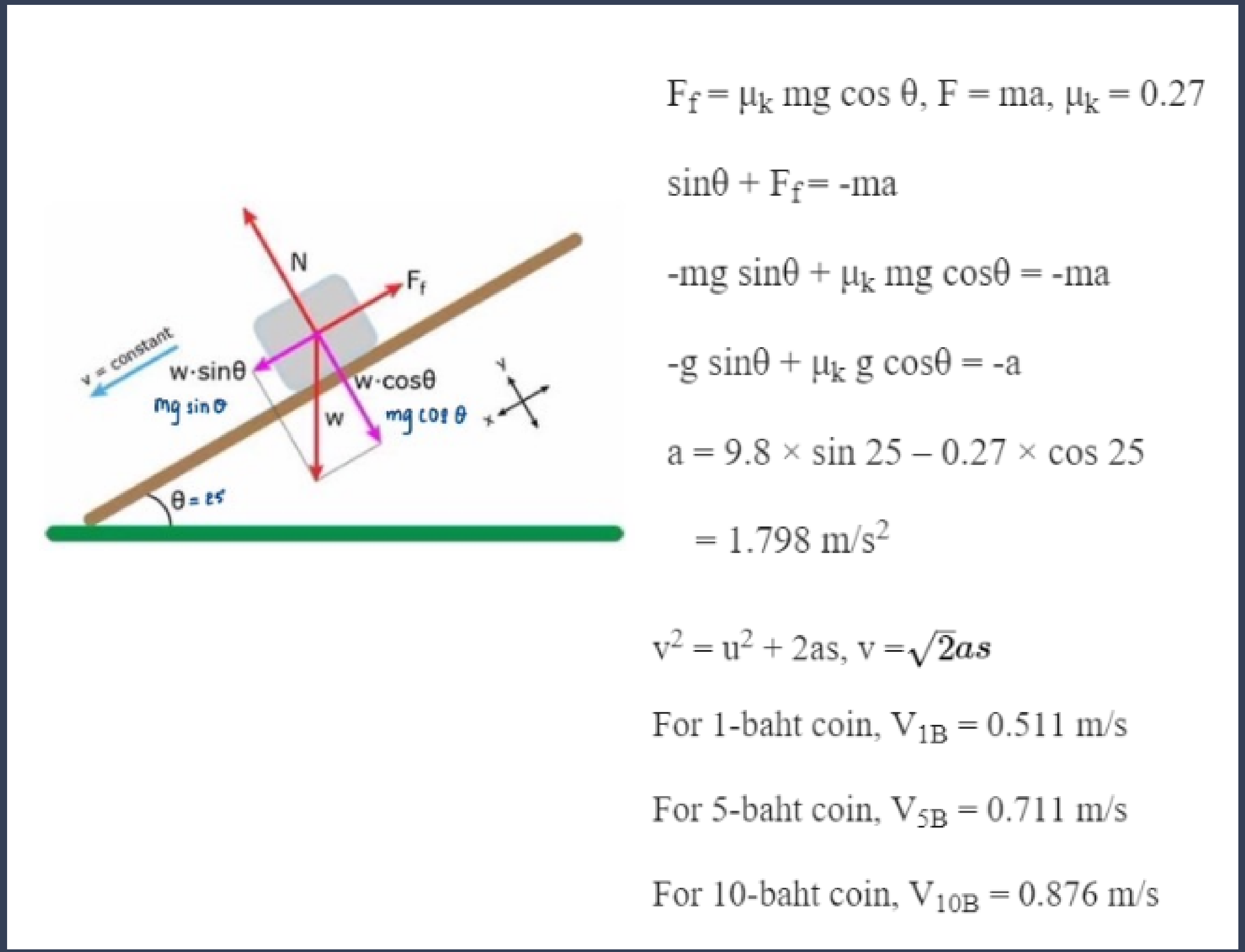
- If the coin is in range, the respective door will open and the coin will fall into the container if the size is in range.
- the IR sensor will detect the coin and the count will go up by one
- the door will close and the process will start again.



Range Table			
Coin	Diameter	Range	Relative error
1 baht coin	20 mm	19.9 mm - 20.1 mm	10%
5 baht coin	24 mm	23.9 mm - 24.1 mm	
10 baht coin	26 mm	25.9 mm - 26.1 mm	

GROUP 2 (COM)

CALCULATIONS



$F_f = \mu_k$ mg cos θ, F = ma, $\mu_k = 0.27$

sinθ + F_f= -ma

-mg sinθ + μ_k mg cosθ = -ma

-g sinθ + μ_k g cosθ = -a

a = 9.8 × sin 25 – 0.27 × cos 25

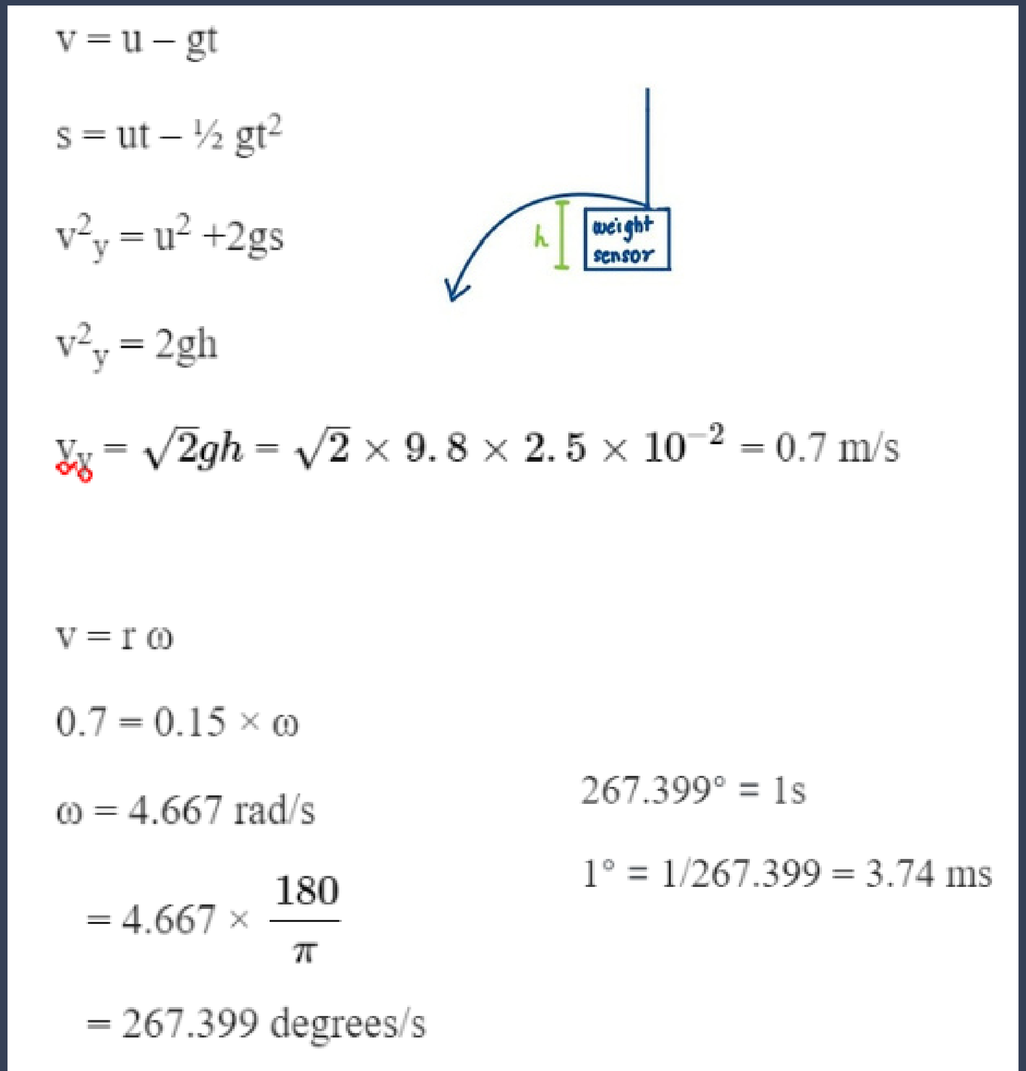
= 1.798 m/s^2

$v^2 = u^2 + 2as$, $v =\sqrt{2as}$

For 1-baht coin, V_{1B} = 0.511 m/s

For 5-baht coin, V_{5B} = 0.711 m/s

For 10-baht coin, V_{10B} = 0.876 m/s



v = u – gt

s = ut – ½ gt^2

$v^2{}_y = u^2 + 2gs$

$v^2{}_y = 2gh$

$v_y = \sqrt{2}gh = \sqrt{2} \times 9.8 \times 2.5 \times 10^{-2}$ = 0.7 m/s

v = r ω

0.7 = 0.15 × ω

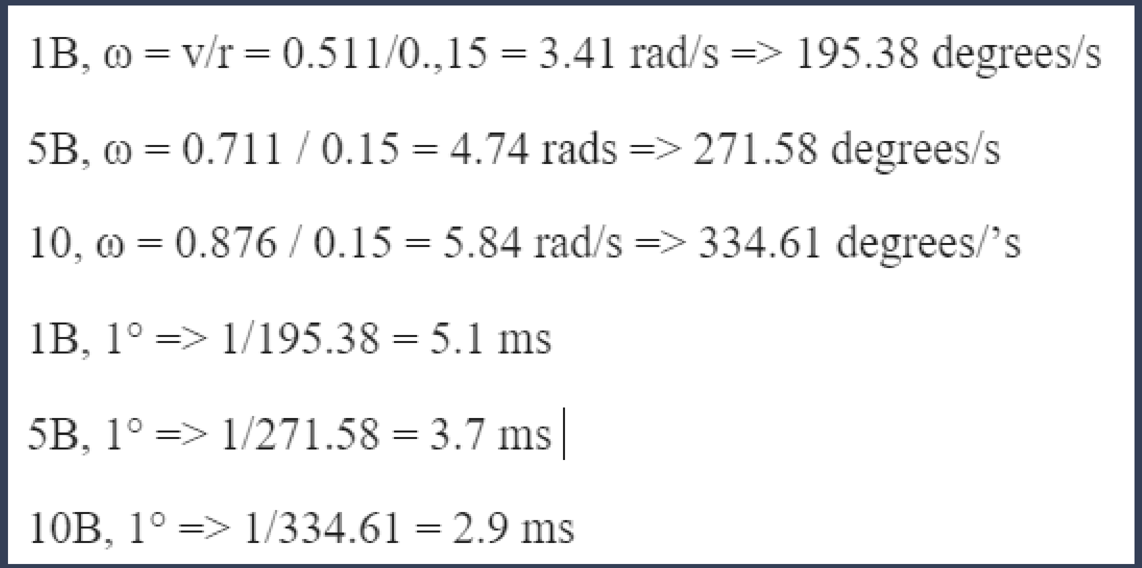
ω = 4.667 rad/s

$= 4.667 \times \frac{180}{\pi}$

= 267.399 degrees/s

267.399° = 1s

1° = 1/267.399 = 3.74 ms



1B, ω = v/r = 0.511/0.,15 = 3.41 rad/s => 195.38 degrees/s

5B, ω = 0.711 / 0.15 = 4.74 rads => 271.58 degrees/s

10, ω = 0.876 / 0.15 = 5.84 rad/s => 334.61 degrees/'s

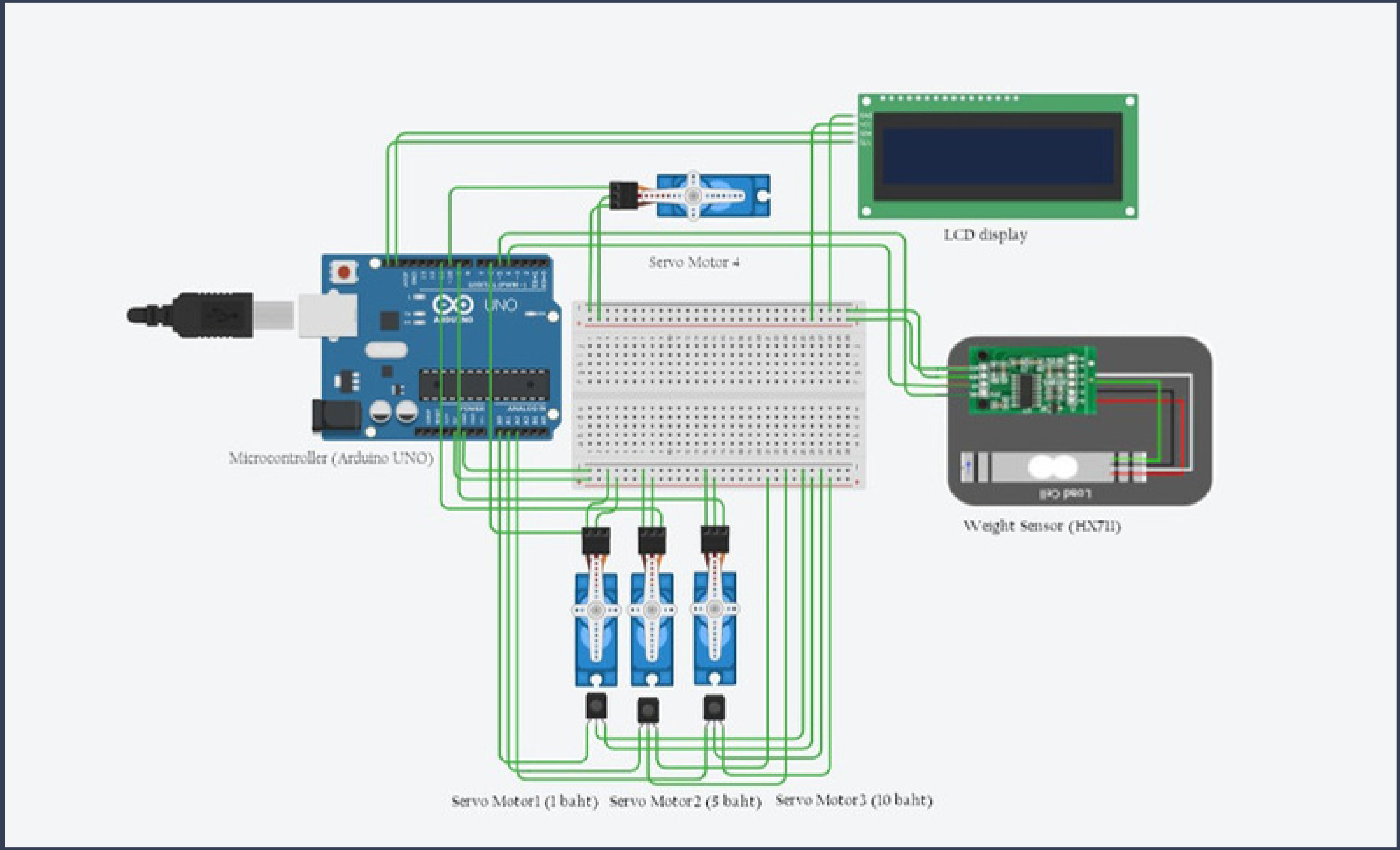
1B, 1° => 1/195.38 = 5.1 ms

5B, 1° => 1/271.58 = 3.7 ms

10B, 1° => 1/334.61 = 2.9 ms

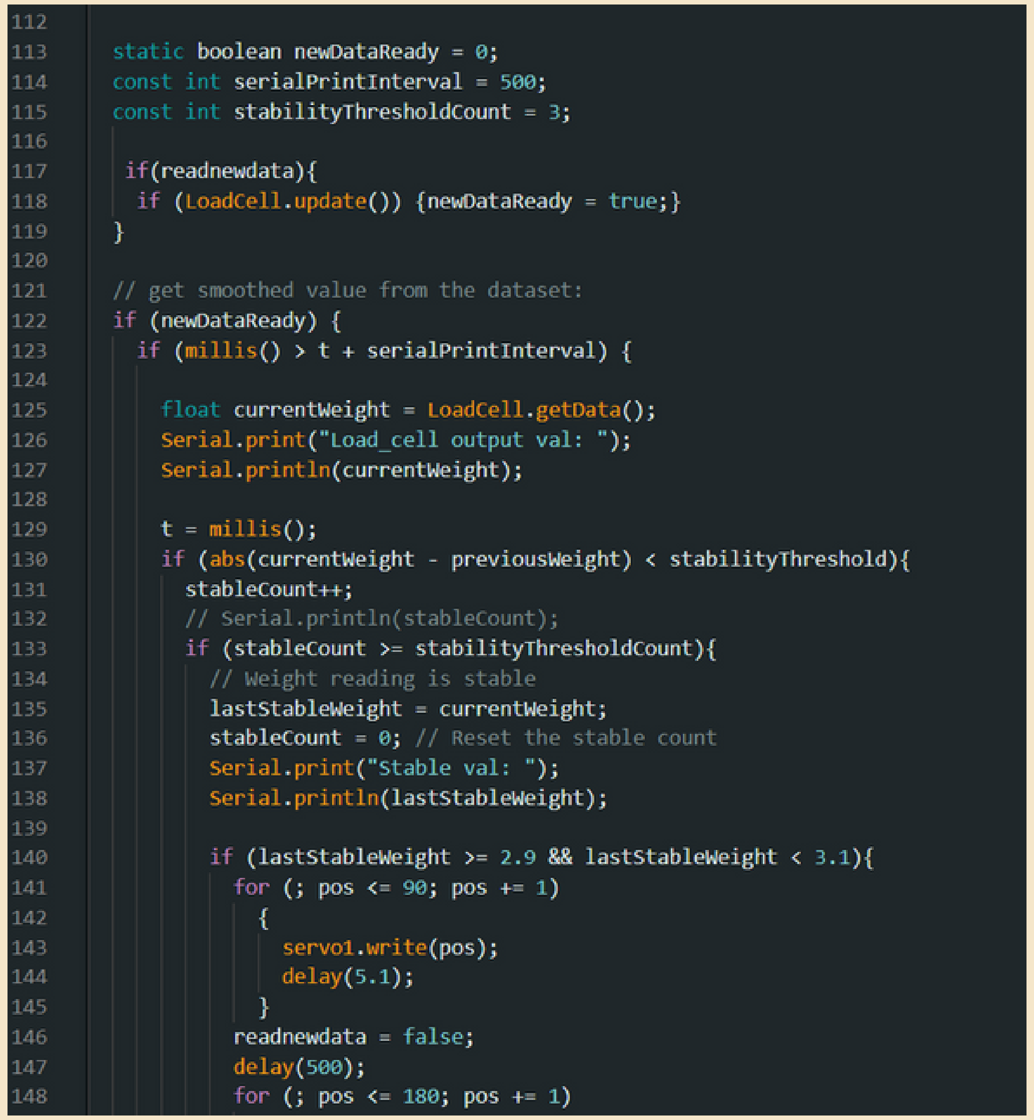
GROUP 2 (COM)

CIRCUIT DIAGRAM

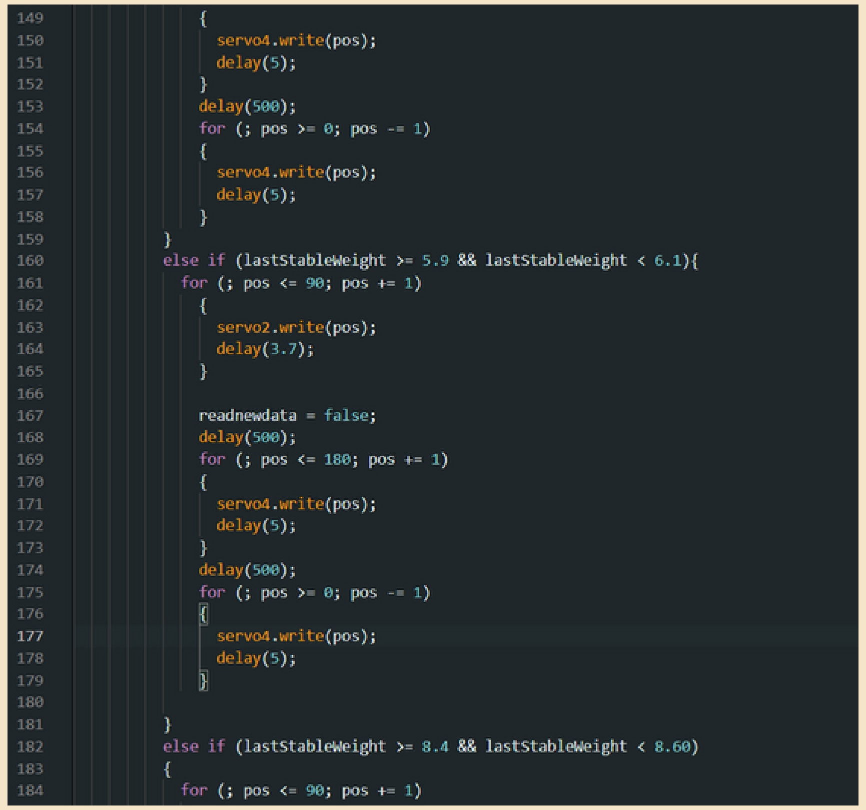


CODE SNIPPET

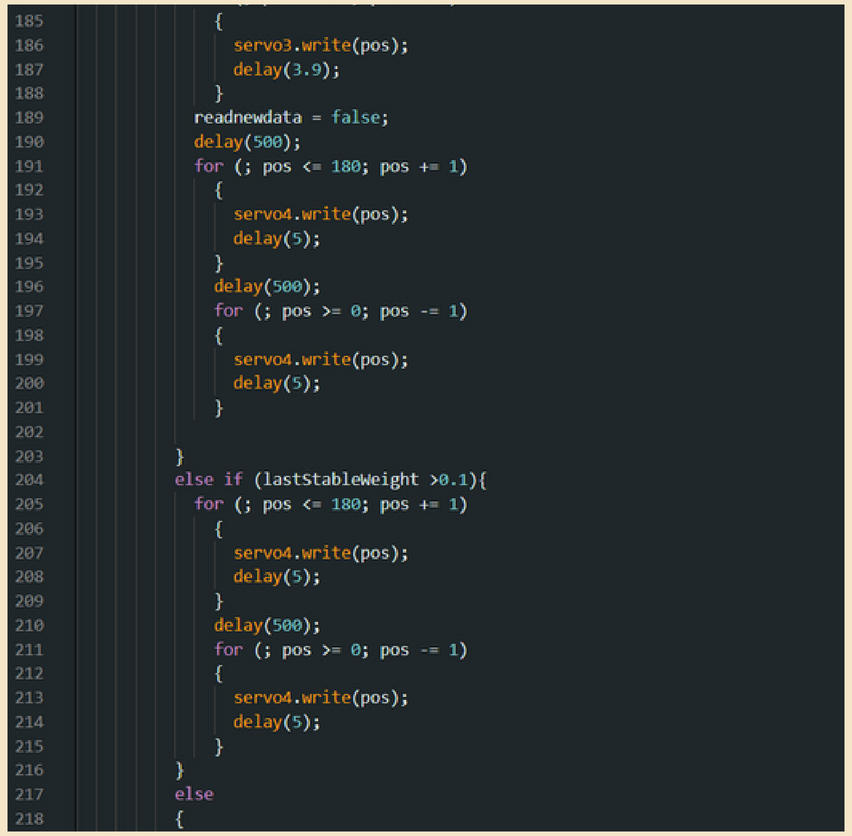
GROUP 2 (COM)



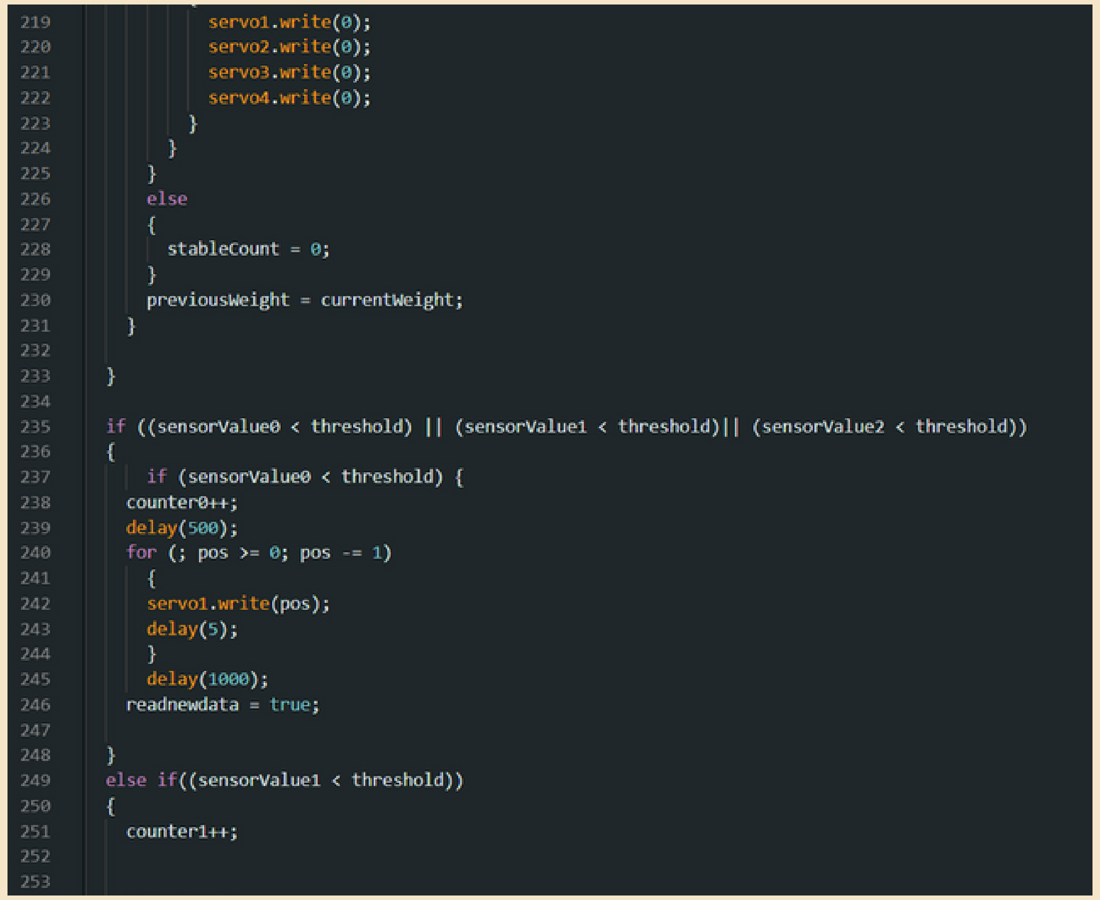
```
static boolean newDataReady = 0;
const int serialPrintInterval = 500;
const int stabilityThresholdCount = 3;

 if(readnewdata){
  if (LoadCell.update()) {newDataReady = true;}
}

// get smoothed value from the dataset:
if (newDataReady) {
  if (millis() > t + serialPrintInterval) {

    float currentWeight = LoadCell.getData();
    Serial.print("Load_cell output val: ");
    Serial.println(currentWeight);

    t = millis();
    if (abs(currentWeight - previousWeight) < stabilityThreshold){
      stableCount++;
      // Serial.println(stableCount);
      if (stableCount >= stabilityThresholdCount){
        // Weight reading is stable
        lastStableWeight = currentWeight;
        stableCount = 0; // Reset the stable count
        Serial.print("Stable val: ");
        Serial.println(lastStableWeight);

        if (lastStableWeight >= 2.9 && lastStableWeight < 3.1){
          for (; pos <= 90; pos += 1)
            {
              servo1.write(pos);
              delay(5.1);
            }
          readnewdata = false;
          delay(500);
          for (; pos <= 180; pos += 1)
```



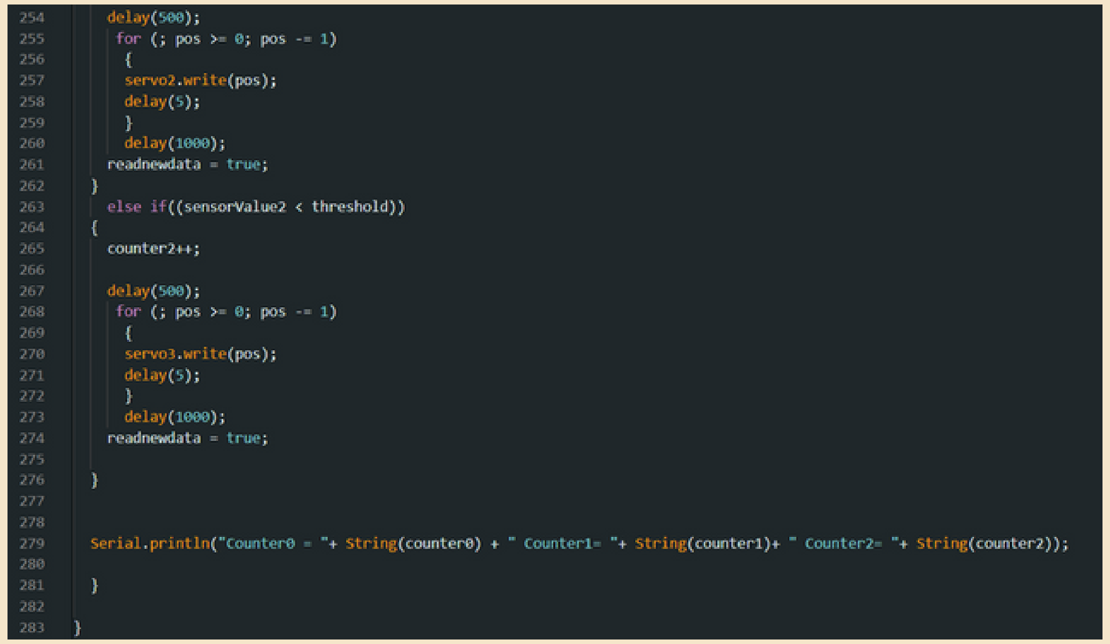
```
            {
              servo4.write(pos);
              delay(5);
            }
            delay(500);
            for (; pos >= 0; pos -= 1)
            {
              servo4.write(pos);
              delay(5);
            }
        }
        else if (lastStableWeight >= 5.9 && lastStableWeight < 6.1){
          for (; pos <= 90; pos += 1)
            {
              servo2.write(pos);
              delay(3.7);
            }

            readnewdata = false;
            delay(500);
            for (; pos <= 180; pos += 1)
            {
              servo4.write(pos);
              delay(5);
            }
            delay(500);
            for (; pos >= 0; pos -= 1)
            {
              servo4.write(pos);
              delay(5);
            }

        }
        else if (lastStableWeight >= 8.4 && lastStableWeight < 8.60)
        {
          for (; pos <= 90; pos += 1)
```



```
            {
              servo3.write(pos);
              delay(3.9);
            }
          readnewdata = false;
          delay(500);
          for (; pos <= 180; pos += 1)
            {
              servo4.write(pos);
              delay(5);
            }
            delay(500);
            for (; pos >= 0; pos -= 1)
            {
              servo4.write(pos);
              delay(5);
            }

        }
        else if (lastStableWeight >0.1){
          for (; pos <= 180; pos += 1)
            {
              servo4.write(pos);
              delay(5);
            }
            delay(500);
            for (; pos >= 0; pos -= 1)
            {
              servo4.write(pos);
              delay(5);
            }
        }
        else
        {
```



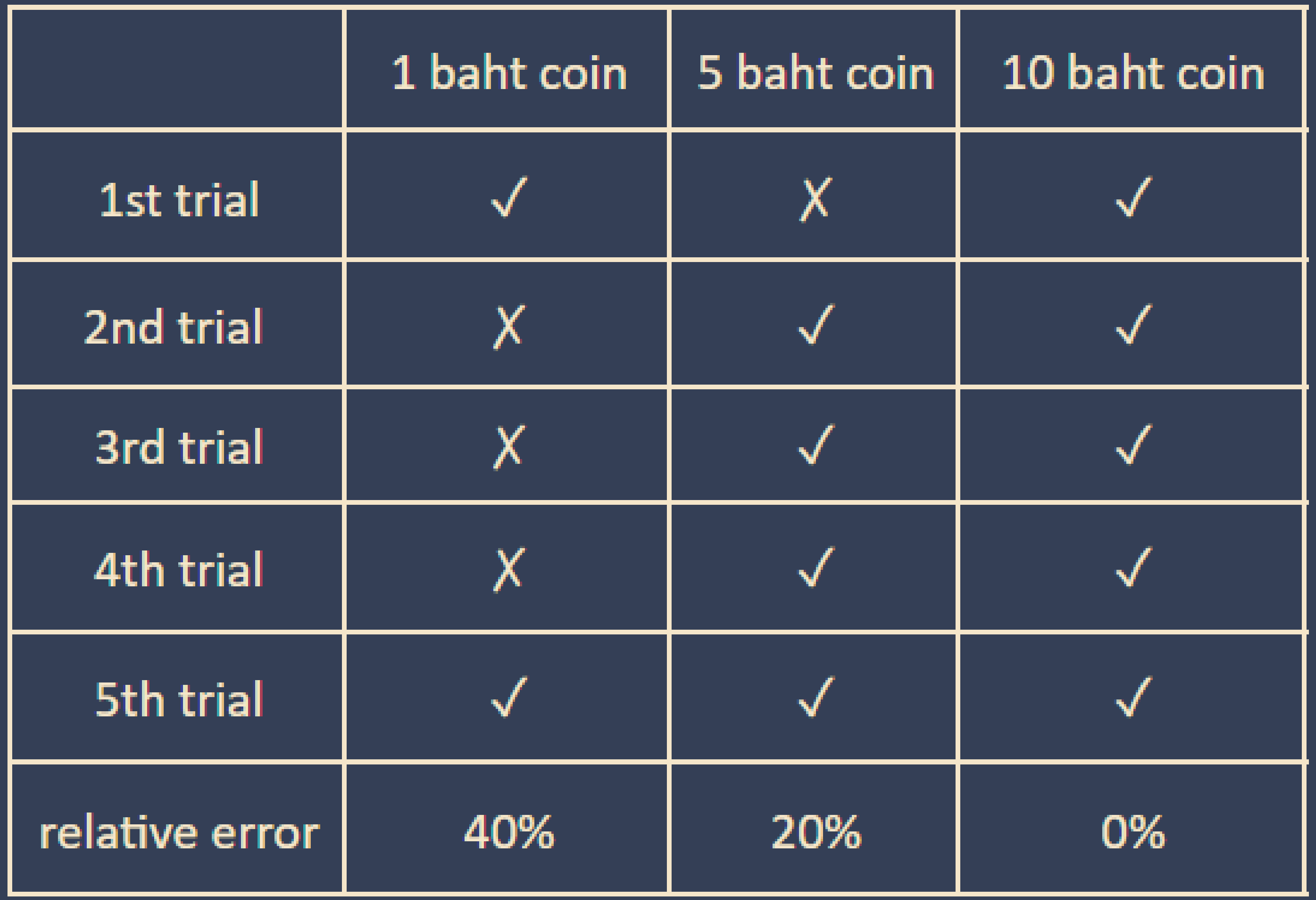
```
            servo1.write(0);
            servo2.write(0);
            servo3.write(0);
            servo4.write(0);
          }
        }
      }
      else
      {
        stableCount = 0;
      }
      previousWeight = currentWeight;
    }

  }

  if ((sensorValue0 < threshold) || (sensorValue1 < threshold)|| (sensorValue2 < threshold))
  {
      if (sensorValue0 < threshold) {
    counter0++;
    delay(500);
    for (; pos >= 0; pos -= 1)
      {
      servo1.write(pos);
      delay(5);
      }
      delay(1000);
    readnewdata = true;

  }
  else if((sensorValue1 < threshold))
  {
    counter1++;


```



```
    delay(500);
     for (; pos >= 0; pos -= 1)
      {
      servo2.write(pos);
      delay(5);
      }
      delay(1000);
    readnewdata = true;
  }
    else if((sensorValue2 < threshold))
  {
    counter2++;

    delay(500);
     for (; pos >= 0; pos -= 1)
      {
      servo3.write(pos);
      delay(5);
      }
      delay(1000);
    readnewdata = true;

  }


  Serial.println("Counter0 = "+ String(counter0) + " Counter1= "+ String(counter1)+ " Counter2= "+ String(counter2));

  }

}
```

GROUP 2 (COM)

TESTING RESULTS

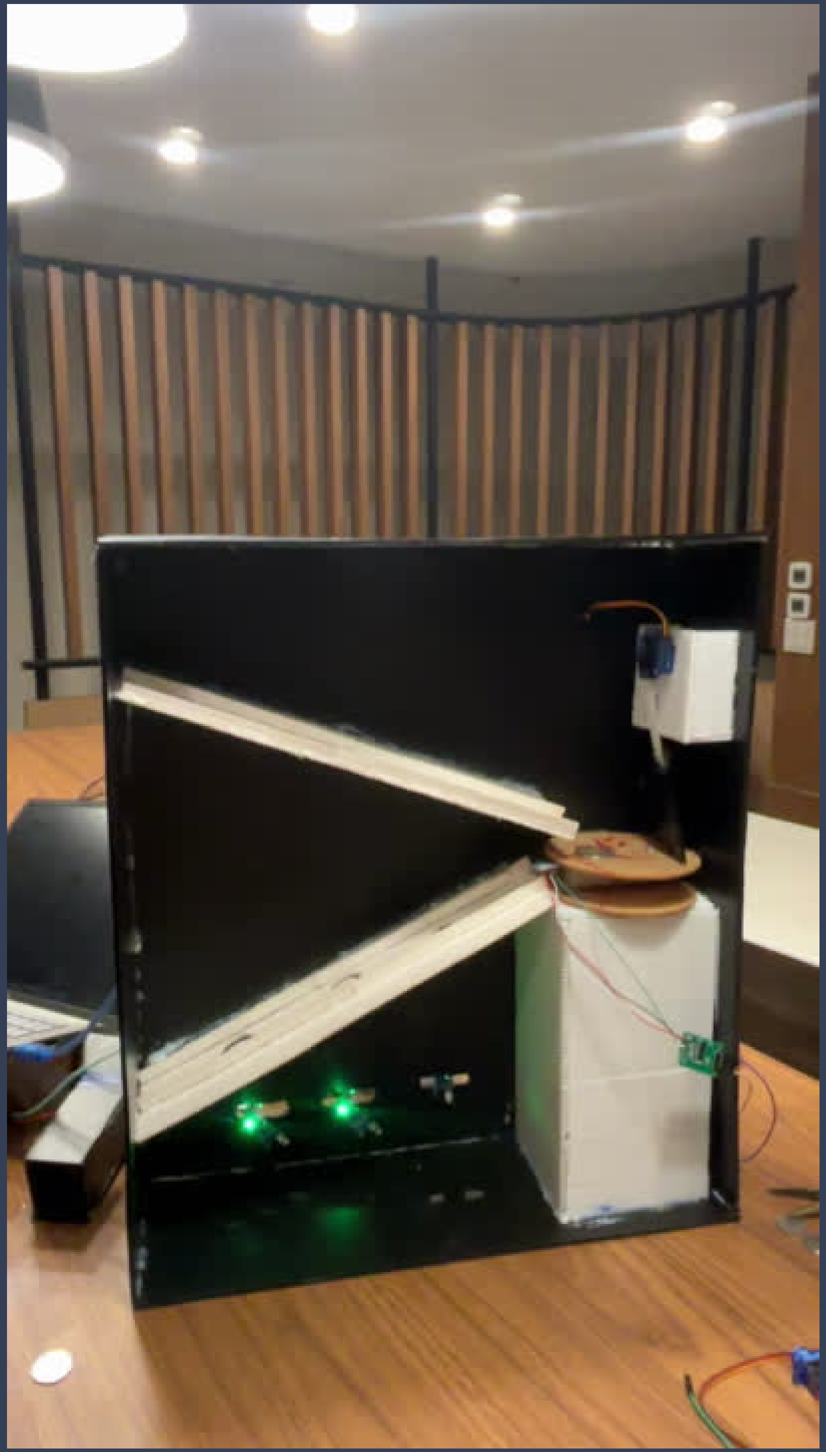


	1 baht coin	5 baht coin	10 baht coin
1st trial	✓	X	✓
2nd trial	X	✓	✓
3rd trial	X	✓	✓
4th trial	X	✓	✓
5th trial	✓	✓	✓
relative error	40%	20%	0%

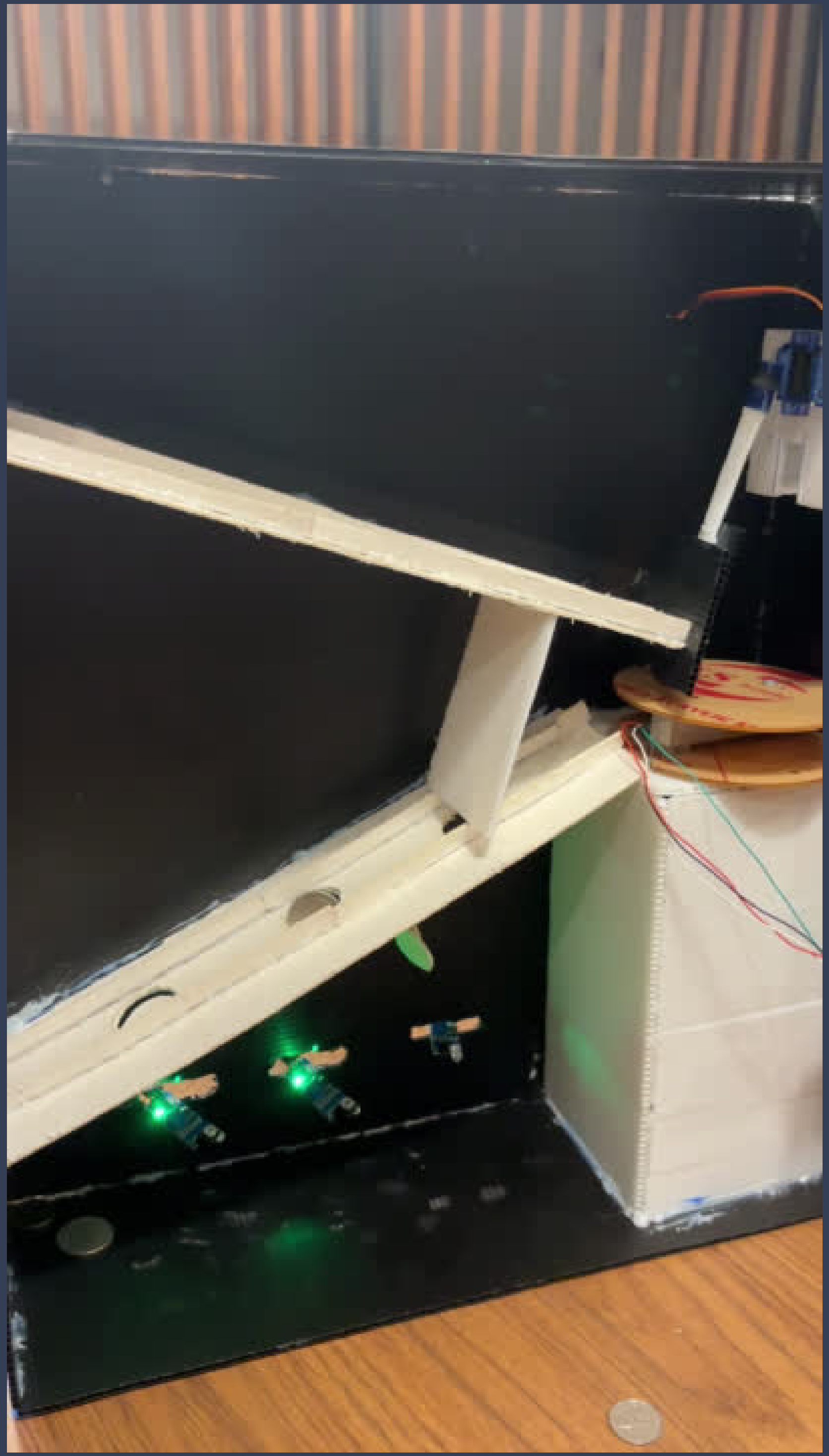
GROUP 2 (COM)

EARLY STAGE

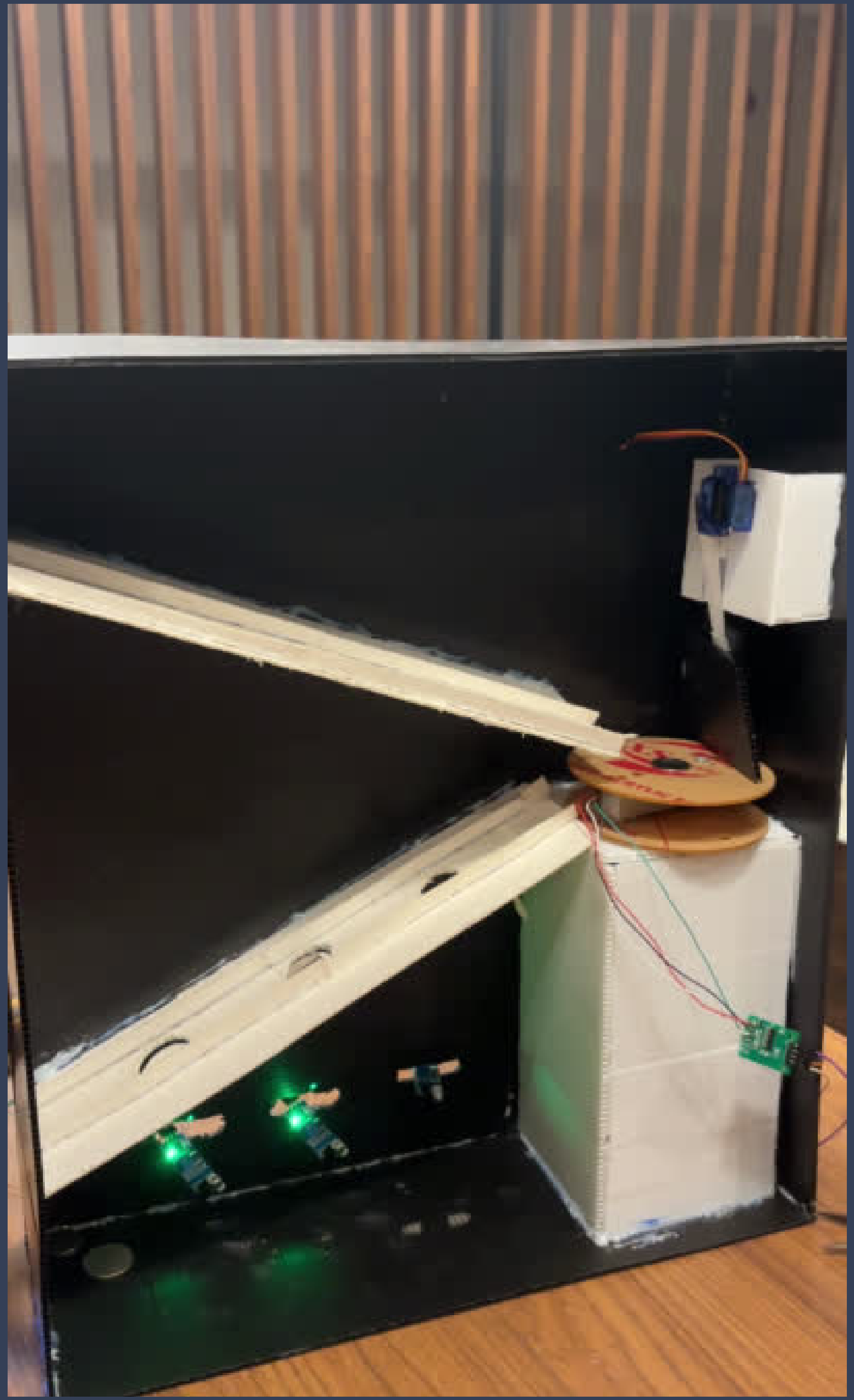
10 baht coin



5 baht coin



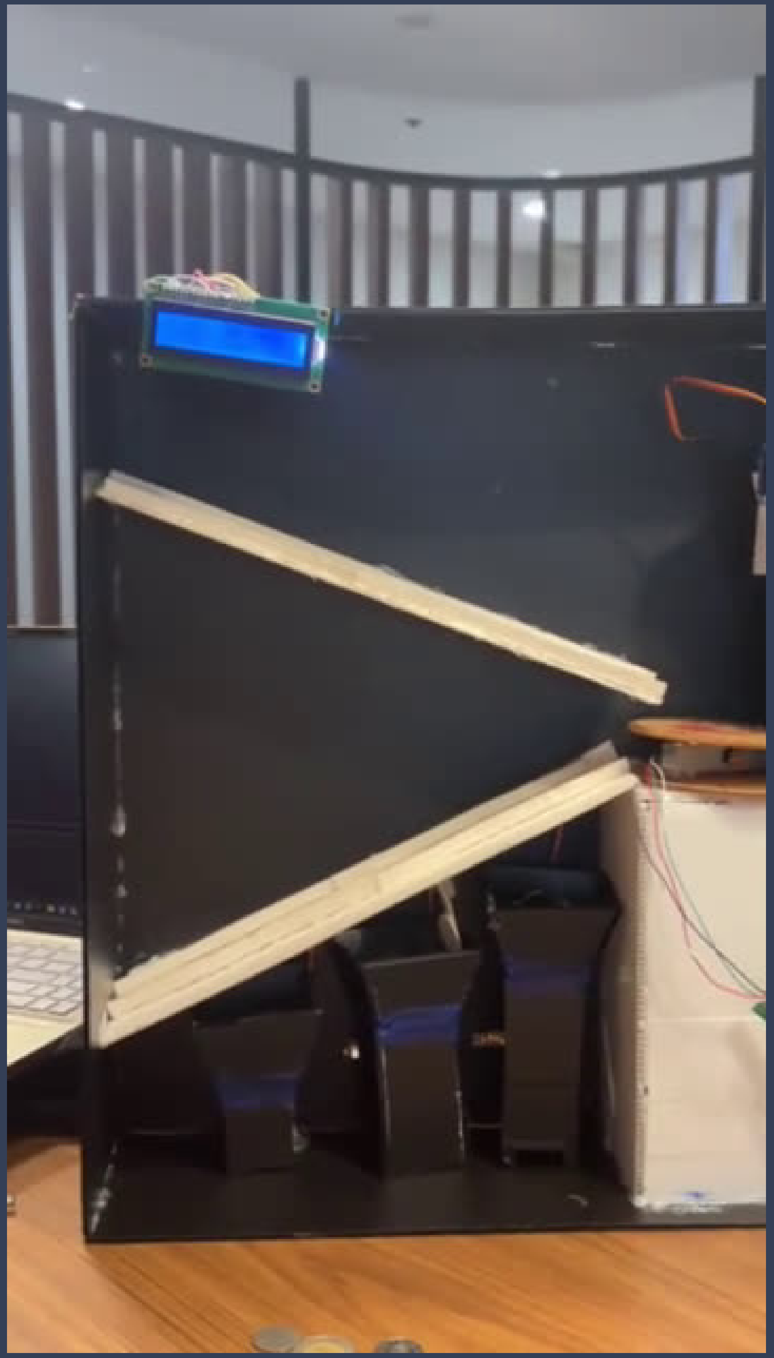
1 baht coin



GROUP 2 (COM)

FINAL RESULT

10 baht coin



5 baht coin

1 baht coin

DISCUSSION

Although the whole circuit works perfectly, we faced some issues with the design and chosen materials when operating the whole system.

CHALLENGING PART

- The coin sliding path
- The weight sensor wiper servo motor
- IR Sensor Counting

POTENTIAL CAUSE

- Since we build the whole model manually, there're some human errors in our design.
- Due to many trials, the material of the path got some additional friction and it's slowing down the coin speed missing our calculations.
- The wiper upon the weight sensor was adding additional weights making weight sensor fail to detect weight correctly or slowing down the wiper.
- Due to high coin's speed, IR sensor rarely detect the coin passing.

OUR APPROACH

- We tried to change the position of the sliding pane and angle, but sometimes the small coins like 1 baht coin may not go through our lane.
- We tried to tilt the wiper, setting that in some degree, not to add additional weight on the weight sensor. Now, it's successfully reading the accurate weight, but sometimes it may miss to wipe out the small and thin coin like 1 baht coin.
- For IR sensors, we tried to build some containers and guided lane to put the coin closer to the IR sensor. But, most of the time, it's working for 10 Baht coin only.

GROUP 2 (COM)

FUTURE PLANS

DESIGN

- Modify the design so that it can handle bulk coin inputs.
- Build a more compact machine design (For better visual, transportation etc)

CIRCUIT

- Modify the design and circuit so that it can handle bulk coin inputs.
- Implement image processing to get a better sorting system.
- Adjust the delays for minimum processing time.
- Test coin conductivity to get rid of fake coins

THANK YOU SO MUCH SIR!