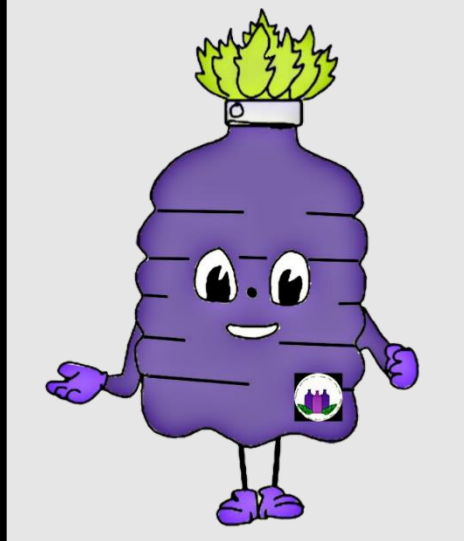
Auto- Plenus

BEBIDAS NATURALES

Manual de Operaciones

Docente: Ing. Imelda Puente Hernández

Alumnos:	Matrículas:
Melissa Díaz Villanueva	301910672
Kevin Alberto Domínguez Charles	301910094
José Gerardo Flores Góngora	301910650
Mario Kevin García Carrazco	301910446
Christian Yonathan Morales Loa	301910445



INDEX

1. BACKROUND

Bottling series machines are made to process the filling of bottles in beverage activities or industries. No matter the size of your company: the bottling process always requires some necessary steps:

1. A filling machine is obviously the core of bottling machines. Through a belt, the bottles move under some filling valves that inject liquid into the bottles.

2. The capping machine is the final device to integrate if you want to complete the bottling machines. After filling, the bottles proceed to pass to this last machine, where a cap is placed on each bottle, sealing the liquid and making the product ready for storage or transport to other places.

Examples of the previous research:

José Porras and Miguel De la Cruz. As shown in the image, the project dealt with an automated system controlled by artificial vision, using hardware (light sensor, stepper motor control) and software (MATLAB control). The basic principle of operation was to transport colored objects (red, green, and blue) along a conveyor belt, detect them with a sensor, capture the image with a common webcam and process it by MATLAB to classify them into three different containers. With this process they achieved an efficient classification of objects in both color and shape.

Figure 1. Project of José Porras and Miguel De la Cruz

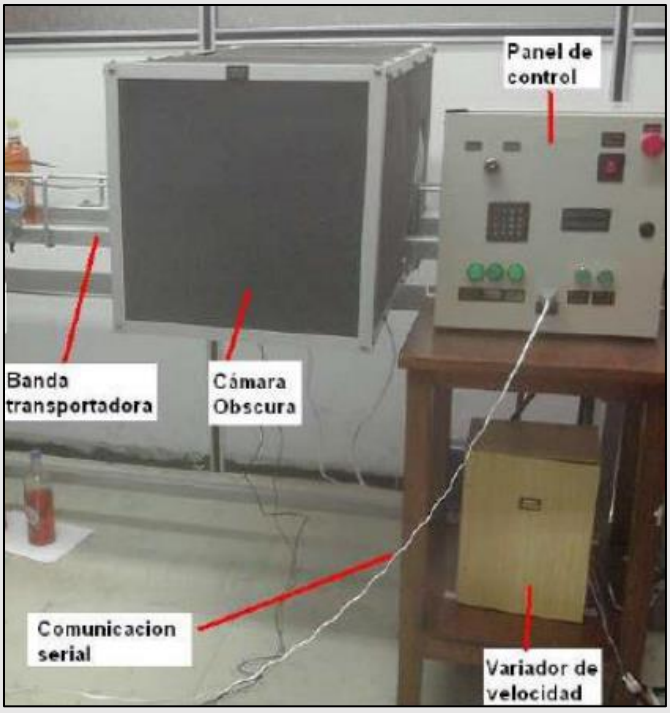


Figure 2. Gavilénez F, Héctor M. et al. The project deals with a quality control system for the filling of glass bottles using the LabView program and the artificial vision toolkit. The system consists of a conveyor belt that allows the movement of bottles to a camera, which obtains images of each bottle for subsequent analysis.

2. SCOPES & LIMITATIONS

First, we will start with the scopes that the mechanical belt can offer us with this project, using the resources of the mechatronic engineering, it is possible to make work a complete line of work in a fast, safe and effective way, it can have local distribution and it can produce around 30 bottles per minute.

On the other hand, we have the limitations of this project, some limitations of this project could be that the belt cannot move any type of object, but they must be bottles with specific measures. Another limitation is that we must put the sugar manually, shake the bottle to mix it well, correct amount of soda, the motors do not resist the mineral water, the sealing needs to be more accurate also to remove the bottle without the interference of a human.

3. OBJECTIVES

General objective

- Design an automatic bottle filling system with liquid level control.

Specific objectives

- Build a one-way conveyor belt for bottle movement. Implement a wheel to transport the bottles to the filling and sealing stations.

- Develop a mechanism for filling and level control of the bottles using three different liquids.

- Develop a sealing mechanism for the bottles.

-Implement a control system connected to the computer to manage the automatic bottling sequence.

4. DESIGN

While doing the scheme, it was made with creative ideas. Besides just drawing it, the team members imagined how could it be made and with which parts it could be constructed of [images will be attached at the end of the document too].

The design of Auto-Plenus (**which is pronounced as *oto-plenos***), is based on the machines that fill beverages at any place of fast food. Bur, albeit being a filler machine with the human hand interacting, it just performs the duty without lots of help of human hand.

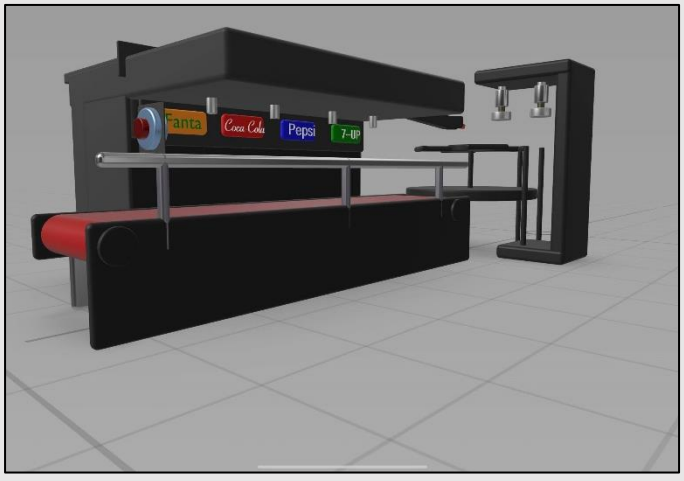
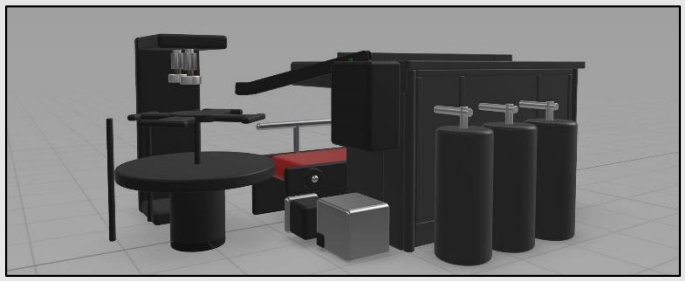
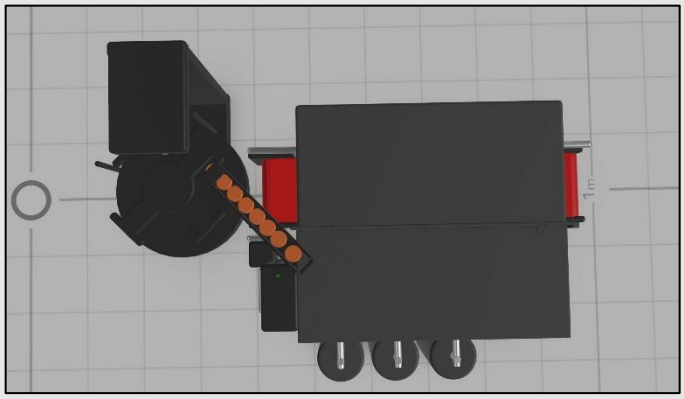


Figure 3. All 3D designs were made in Reality Composer App from Appstore.



From this visual plane we can regard crystal-clear the motor NEMA 34, the power source form a old computer, the sealing stage, and all containers of liquid.

This other picture shows the design from another angle, specifically from top view.



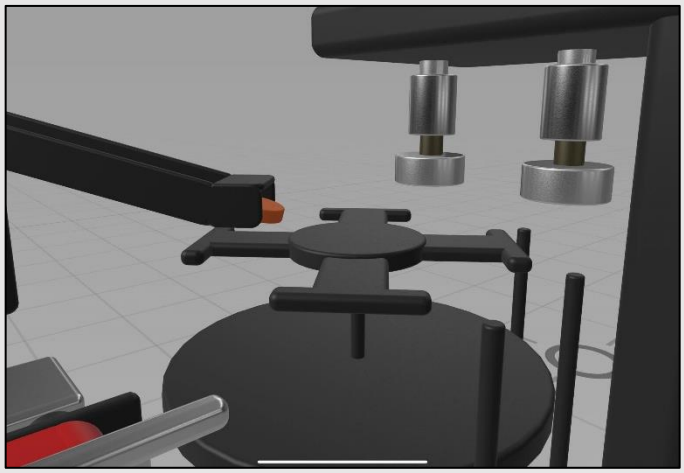


Figure 4. An approach to the sealing section.

5. PLANNING & COSTS

The planning of activities that we were carrying out was done by weeks, as can be seen in the following chart (the charts will also be attached at the end of the document for a better understanding:

ACTIVITY	May			June				July				August	
	1	2	3	4	5	6	7	8	9	10	11	12	13
1. Project selection													
2. Projects design													
3. Quotation of materials													
4. Purchase of materials													
5. Structure elaboration													
6. Testing													
7. Components instalation													
8. Testing													
9.Final programming													
10. Final corrections													

We also began registering all the activities with a simple chart as shown on the next entire page.

Planeación de actividades - Cronograma

Cronograma	Kevin Alberto Domínguez Charles	Jose Gerardo Flores Gongora	Mario Kevin García Carrazco	Melissa Diaz Villanueva	Christian Yonathan Morales Loa
10-Mayo a 16-Mayo	Búsqueda de bombas y válvulas. Reunión para determinar materiales de llenado de refresco.	Búsqueda de información sobre bandas transportadoras	Organización del proyecto, diseño físico del proyecto, y exposición del proyecto y válvulas. Reunión para determinar materiales de llenado de refresco.	Elaboración de la lista de componentes.	Elaboración de la lista de componentes.
17-Mayo a 23-Mayo	Compra de mini brushless dc pump para experimentación. Experimentación con el llenado, recolección de componentes.	Conseguir distribuidores de materiales.	Experimentación con el llenado. Recolección de componentes.	Logística del tapado y pedir materiales.	Logística del tapado y pedir materiales.
24-Mayo a 30-Mayo	Conseguir materiales restantes. Exposición de la presentación.	Elaboración de material de PPT. Exposición de la presentación. Armado de la banda.	Conseguir materiales restantes, prácticas con la fuente de poder de computadora, recolección de materiales extras y elaboración de material de ppt. Exposición de la presentación.	Elaboración de material de PPT. Recolección de materiales. Exposición de la presentación.	Elaboración de material de PPT. Recolección de materiales. Exposición de la presentación.
31-Mayo a 6-Junio	Avance de estructuración del circuito. Preparar cables para el circuito.	Avance a la banda de producción.	Avance de estructuración del circuito. Preparar cables para el circuito.	Avance de estructuración del circuito. Preparar cables para el circuito.	Avance de estructuración del circuito. Preparar cables para el circuito.
7-Junio a 13-Junio	Investigación de diagramas.	Avance a la banda de producción.	Investigación de diagramas.	Investigación de diagramas.	Descripción de lazo cerrado para sellado.
14-Junio a 20-Junio	Descripción de lazo cerrado de banda de refil	Descripción de lazo cerrado para banda.	Descripción de lazo cerrado de banda de refil	Descripción de lazo cerrado para sellado.	Descripción de lazo cerrado para sellado.
21-Junio a 27-Junio	Predeterminado de medidas de base de refil.	Pintar la banda.	Predeterminado de medidas de base de refil.		

Figure 5. First part of the simple chart of recording the activities that every member performed.

Cronograma	Kevin Alberto Domínguez Charles	Jose Gerardo Flores Gongora	Mario Kevin García Carrazco	Melissa Diaz Villanueva	Christian Yonathan Morales Loa
28-Junio a 4-Julio	Pedido de la base de refil. Pruebas con el ultrasonido.	Pruebas con el ultrasonido. Pintar la banda.	Pedido de la base de refil. Pruebas con el ultrasonido.	Pruebas con el ultrasonido.	Pruebas con el ultrasonido.
5-Julio a 11-Julio	Armado del circuito e investigación de ultrasonido ArduinoxLabVIEW. Documento de pines (avance).	Toma de medidas para acoplamiento de la banda al refil. Reunión para corrección de ideas.	Armado del circuito y principios de programación. Declaración de pines de refil. Reunión para corrección de ideas.	Reunión para corrección de ideas.	Reunión para corrección de ideas.
12-Julio a 18-Julio	Diseños	Programación	Desarrollo de VI de ultrasonido.	Diseños	Programación
19-Julio a 25-Julio	Cables	Programación	Programación	Cables	Programación
26-Julio a 1-Agosto	Cables	Programación	Programación	Cables	Programación
2-Agosto a 5-Agosto	Cables	Programación	Programación	Cables	Programación
6-Agosto	Presentación Final	Presentación Final	Presentación Final	Presentación Final	Presentación Final

Figure 6. Second part of the simple chart of recording the activities that every member performed.

The costs are listed in the following table:

[All costs were rounded up]

Object	Quantity	Unitary Cost	Total Cost
Production Line			
Fomi	1	$20	$20
Motor Band	1	$170	$170
Can of painting	2	$40	$80
L298N	1	$60	$60
Motor Nema 34	1	$400	$400
Wood	$1m^2$ triplay 16mm	$70	$70
Wood anchors	1 Bag (50)	$30	$30
Rods	4	$120	$120
Bottle Sealing			
CNY70	2	$15	$30
Servomotor c120	3	$125	$375
Ultrasonic Sensor	1	$45	$45
Geared Motor	2	$45	$90
Wood	$1m^2$ triplay 16mm	$95	$95
H-Bridge	1	$60	$60

Bottle Filling			
Liquid Container	4	$29	$116
Concentra-ted liquid container	4	$19	$76
Pumps	9	$40	$360
Liquid tubes	6 m	$10 per meter	$60
Wooden machine case	1 (includes everything)	$750	$750
Plasti Loka	4	$22	$88
Computer Power Source	1	$120	$120
Cables	13 m	$7	$91
Insulating tape	2	$13	$26
Leds	1	$3	$3
Buzzer	5	$3	$15
Tcrt5000	2	$49	$88
Ultrasonic Sensor	3	$45	$135
Labels and printing material	1 (tabloid)	$35	$35
Acrilic Paint bottle	1	$40	$40
Kola loca	2	$ 24	$48
Two-sided adhesive tape	1	$62	$62
Total Cost			
Production Line			$950
Bottle Sealing			$695
Bottle Filling			$2113
TOTAL			$3758

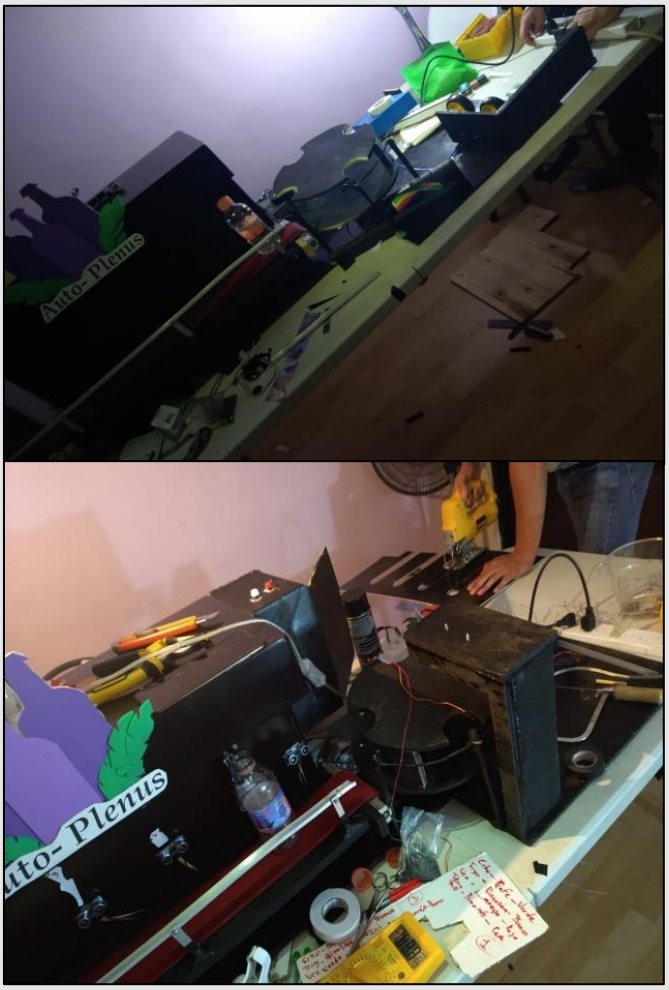
6. CONSTRUCTION

A) Cut

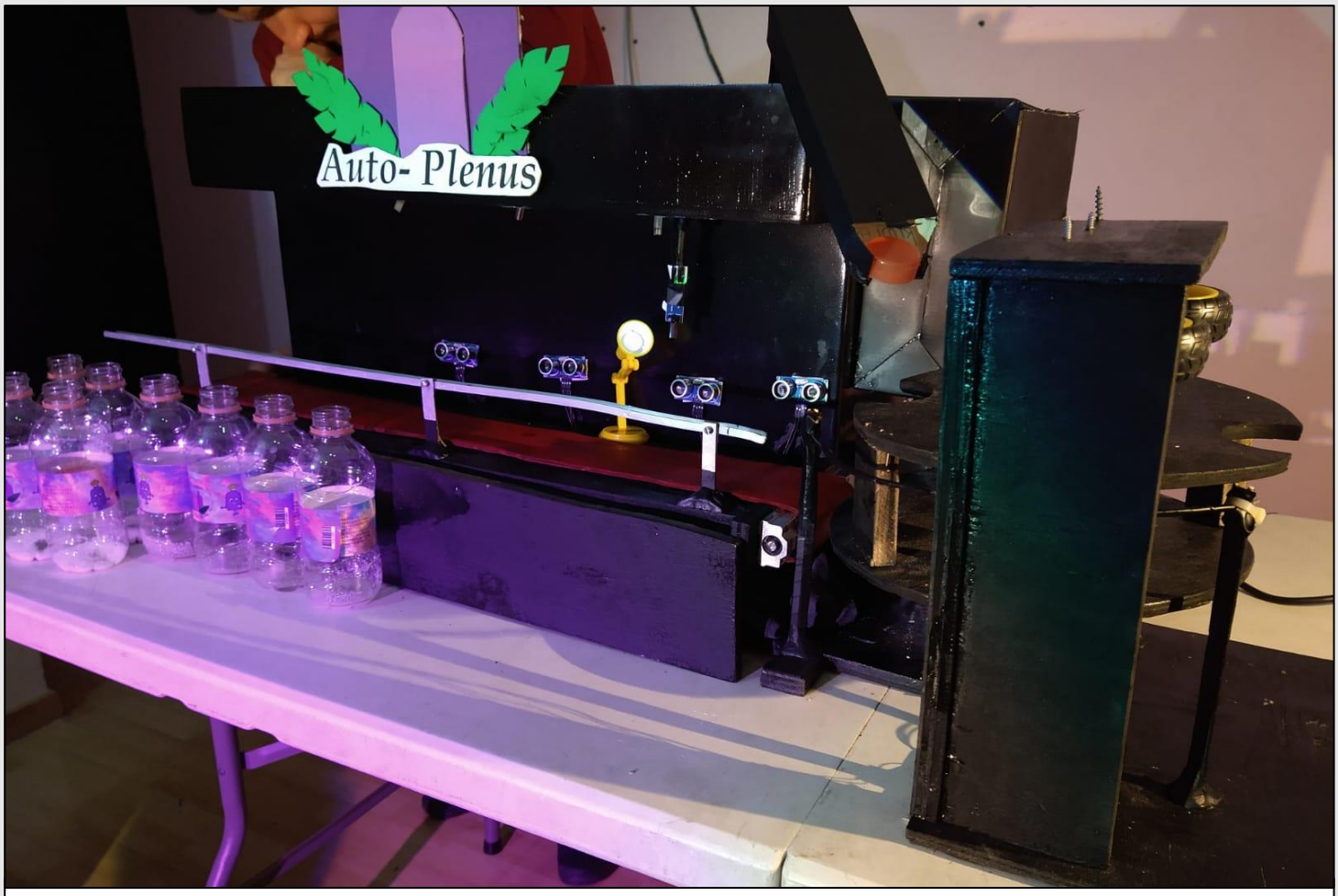
Figure 7. The previous images were from the sealing part.

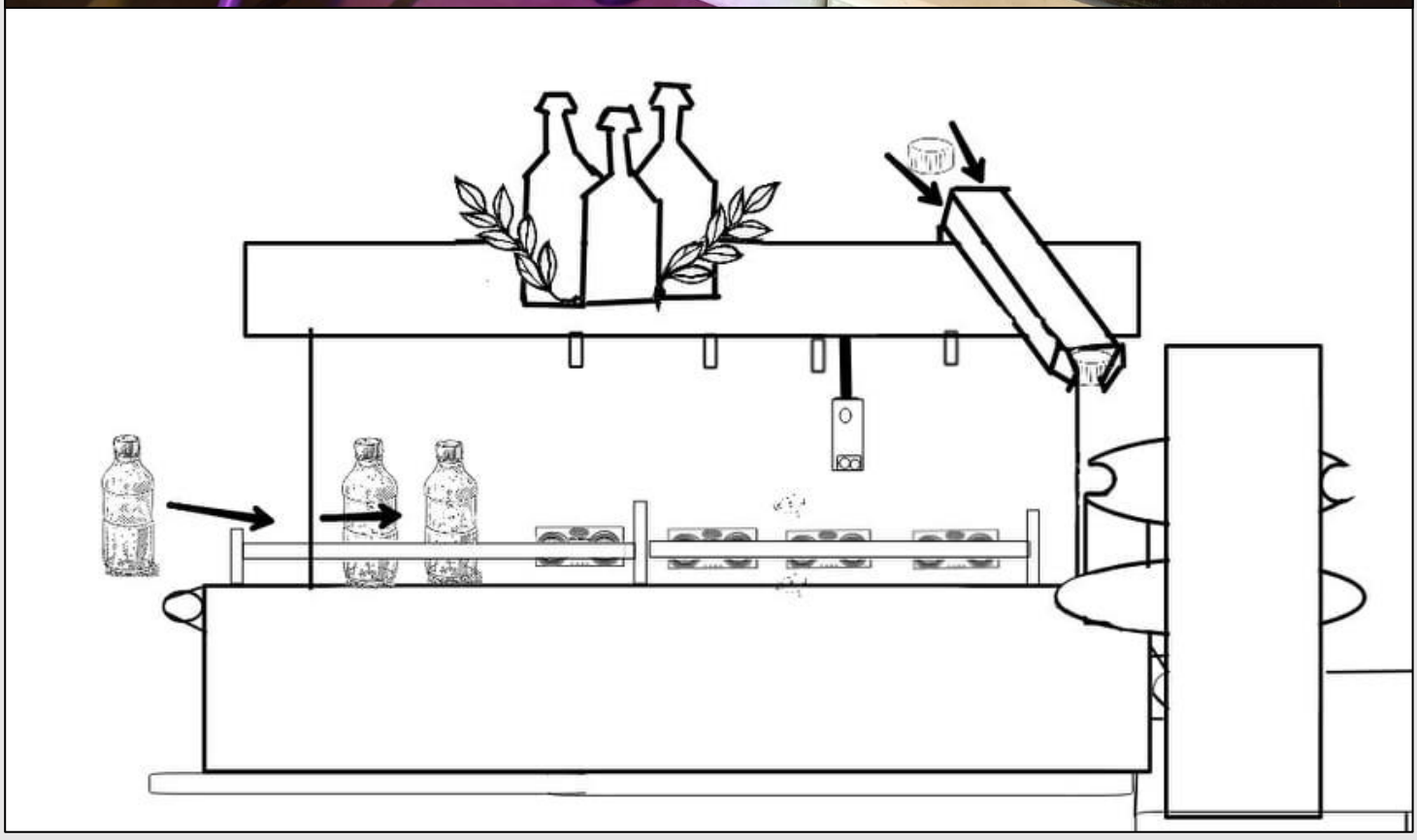
B) Painting

C) Assembling



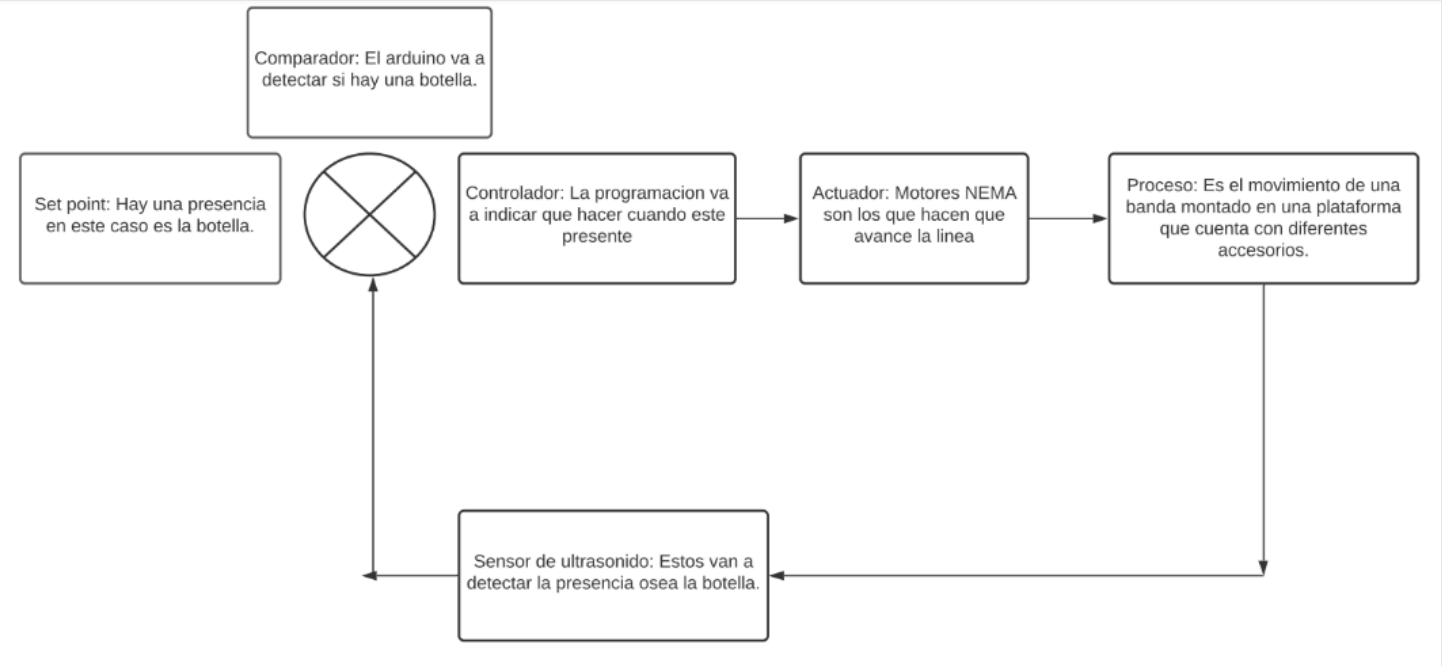
D) Final Product



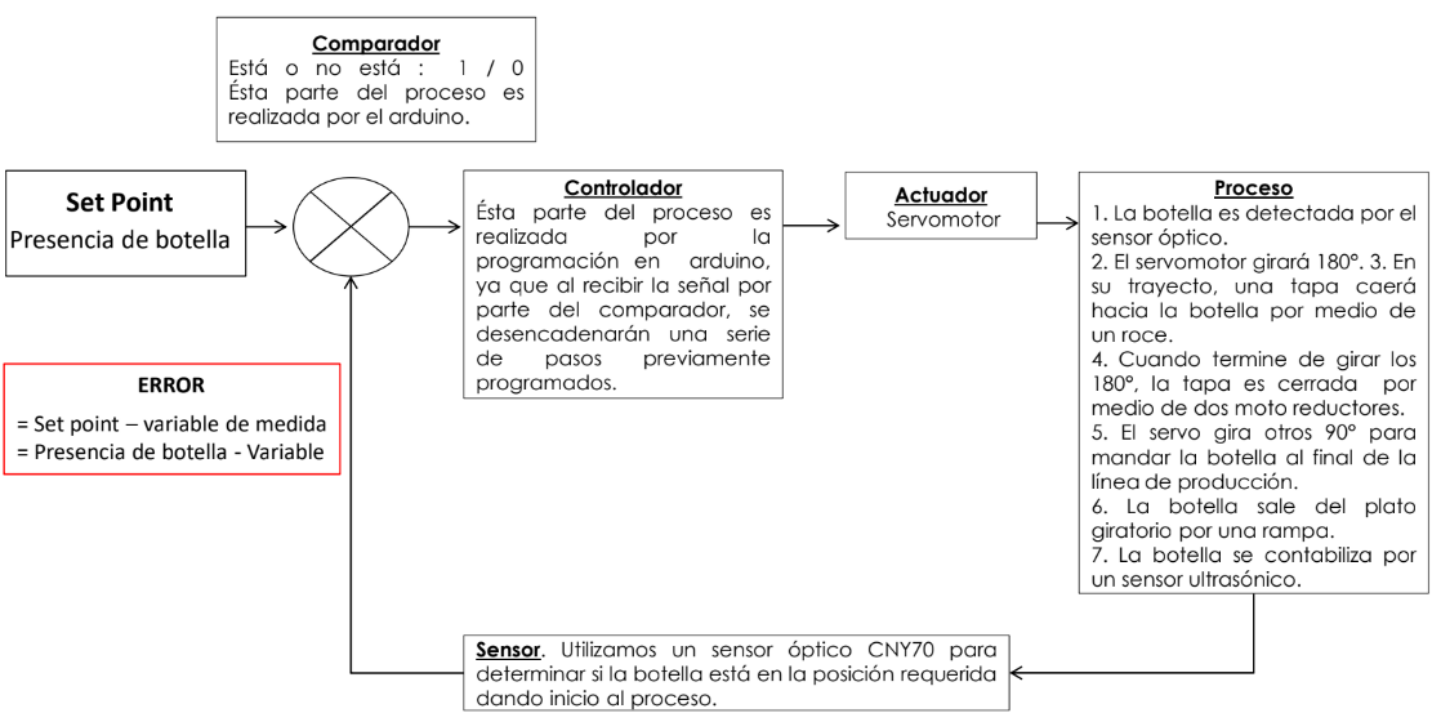


7. BLOCK & FRITZING DIAGRAMS

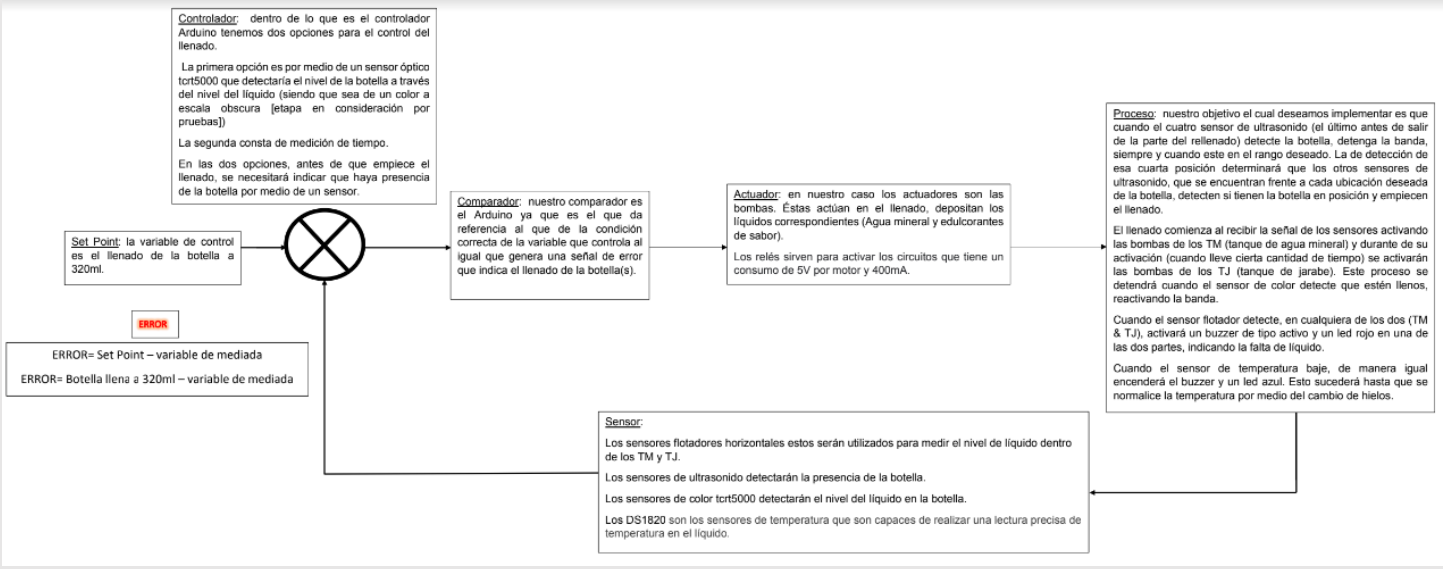
A) Production Line



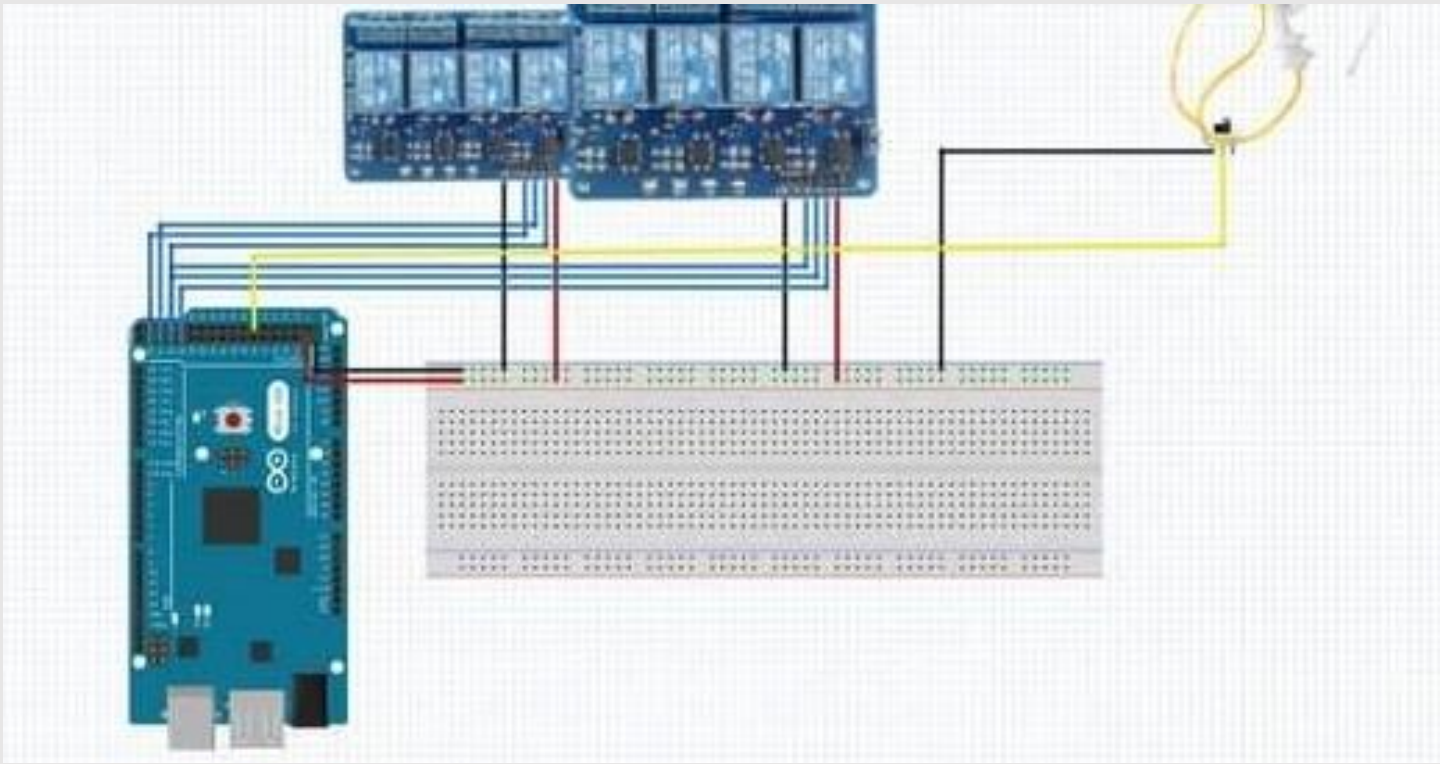
B) Bottle Sealing



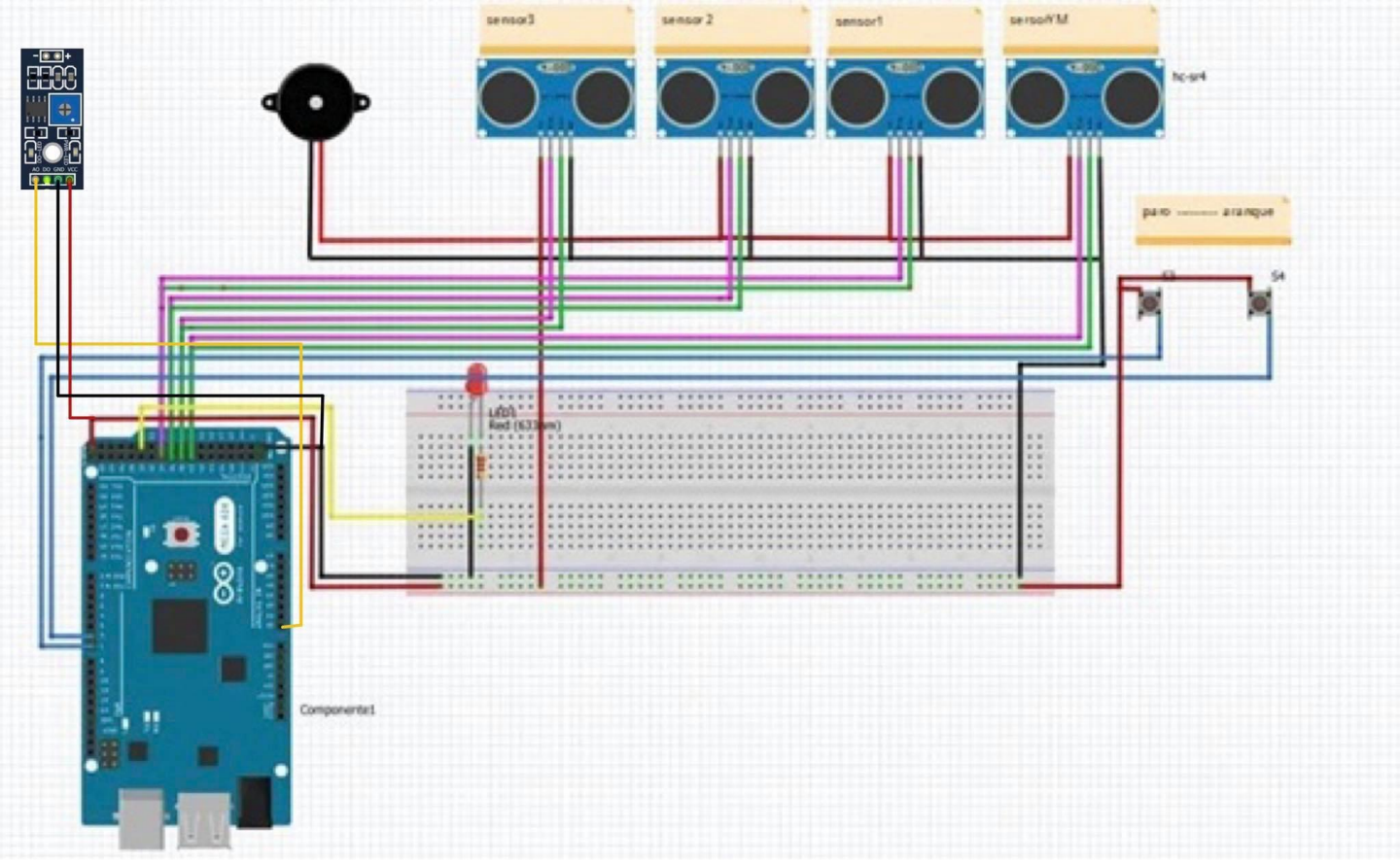
C) Bottle Filling



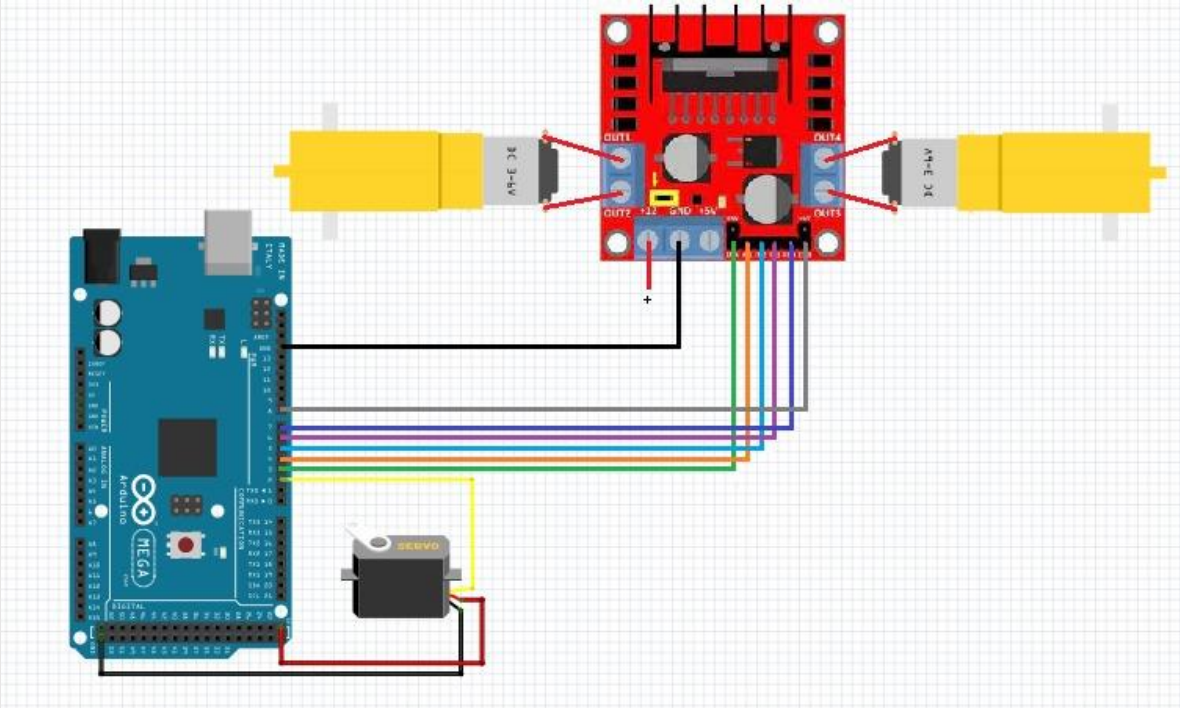
D) Pumps Connection



E) Sensors Connection

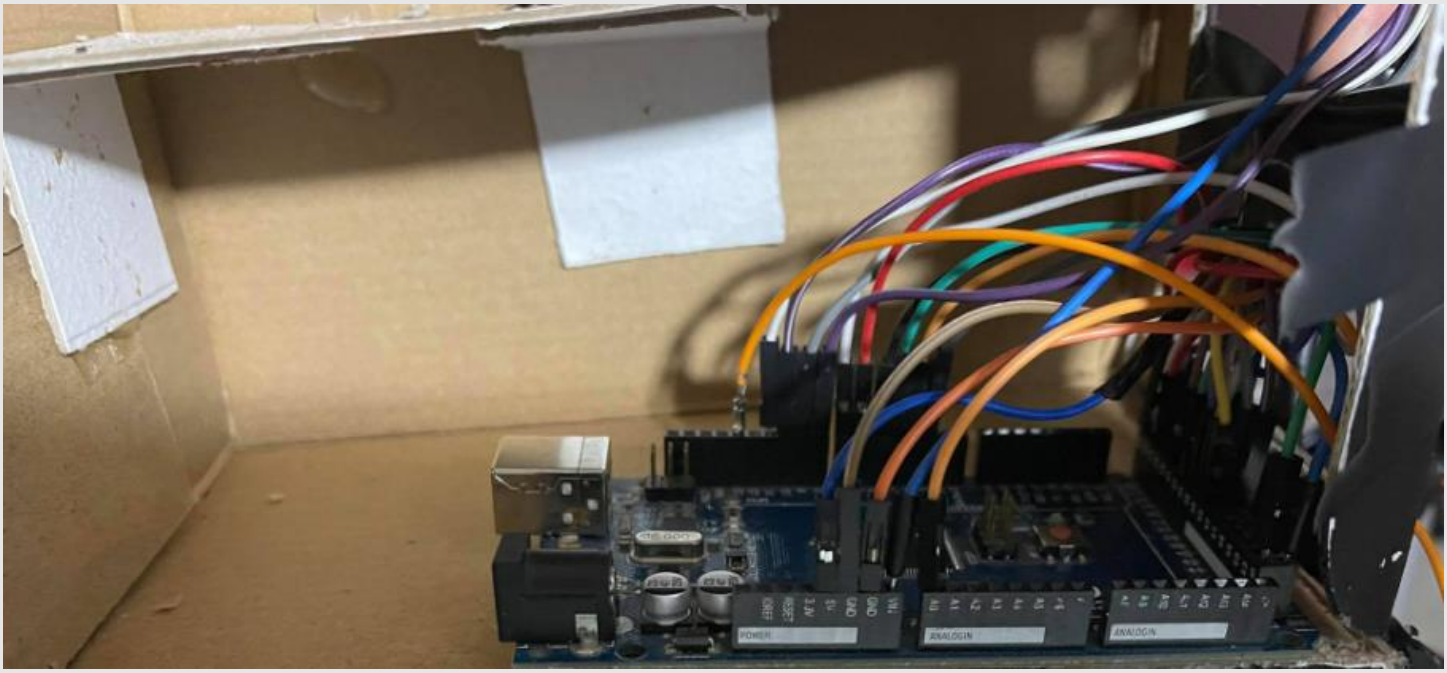


F) Sealing Connection

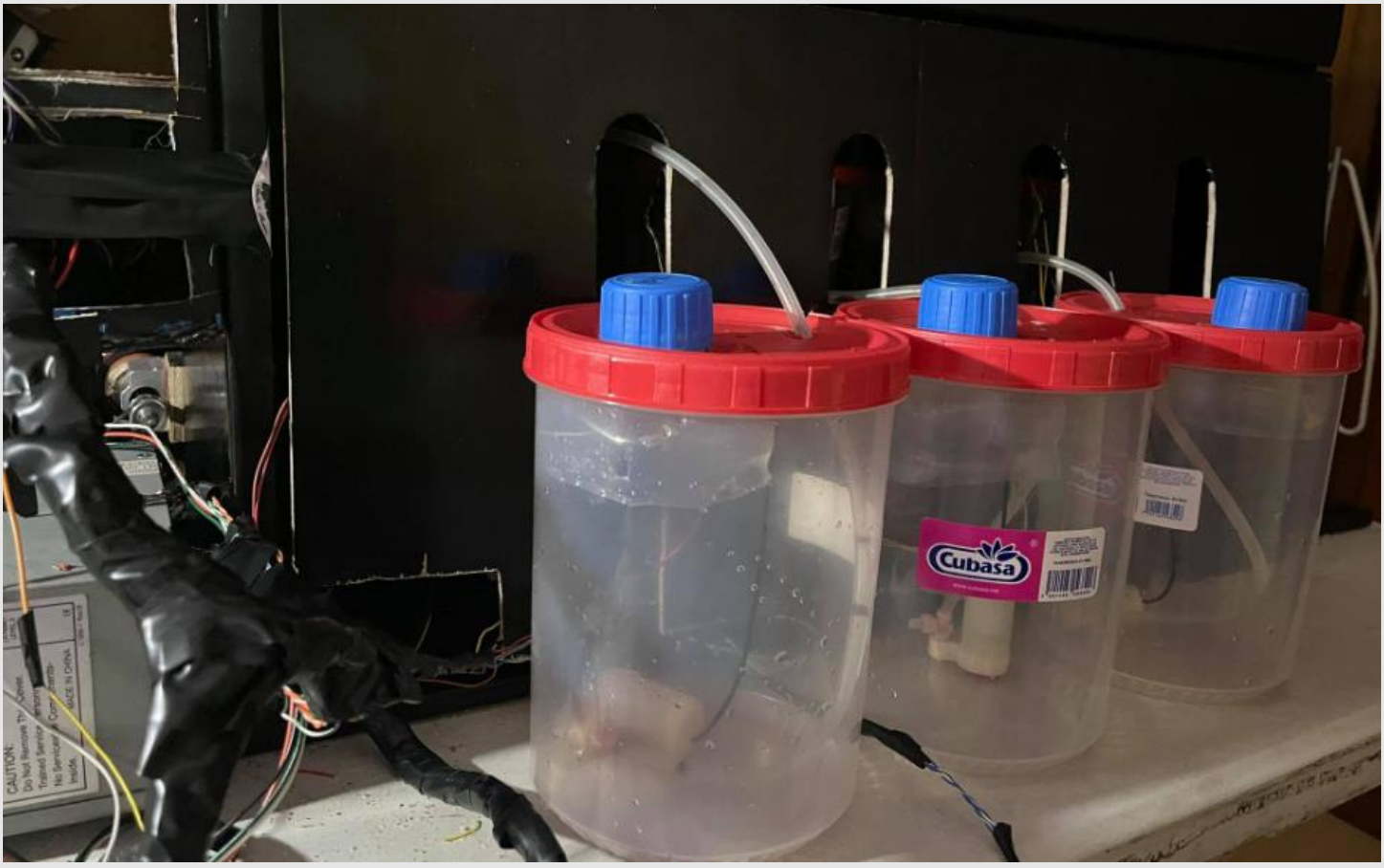


G) General Real Connections

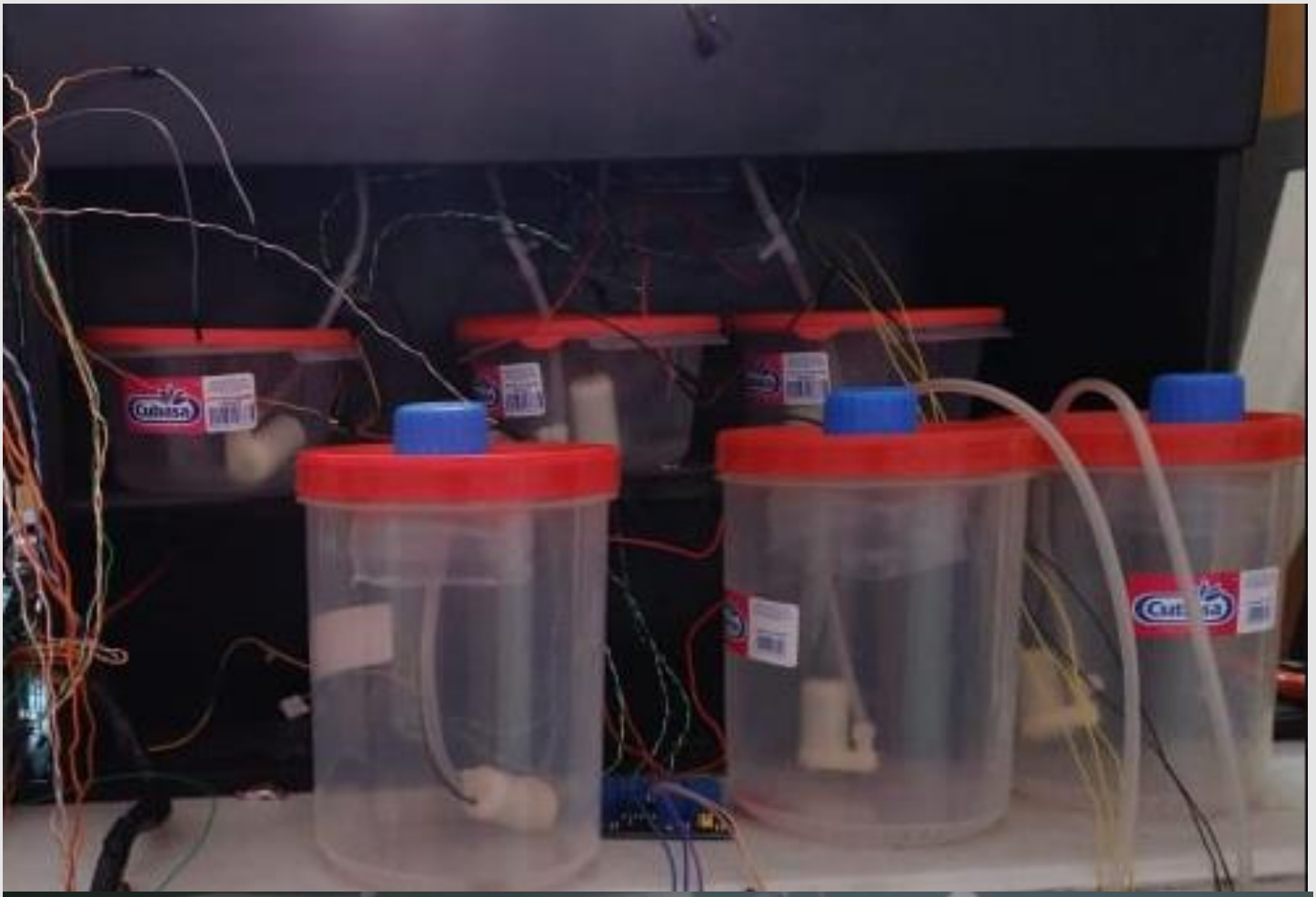
1. Arduino Mega’s

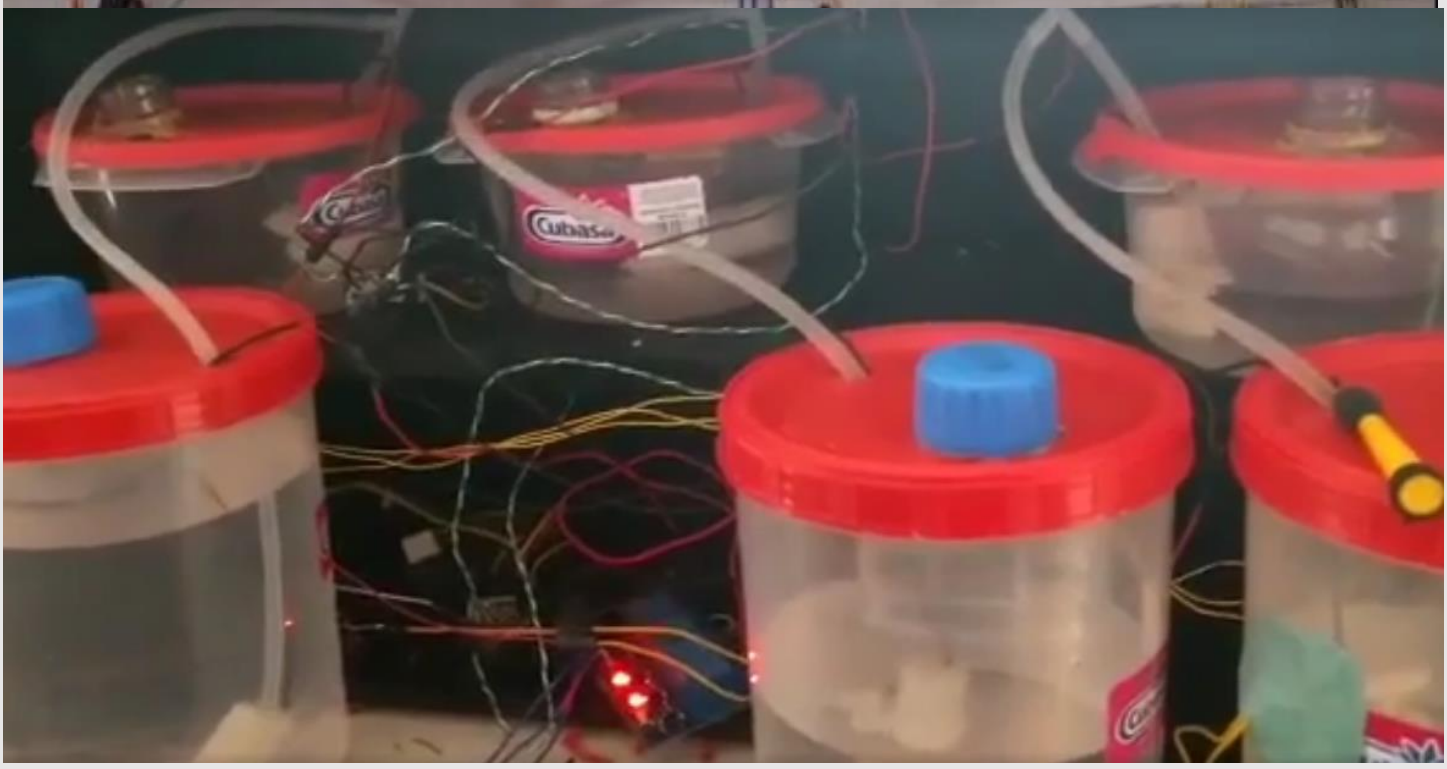


2. Tanks, sensor, power supply’s



3. Rele’s





[ALL THE DESORDED WAS REORDERED BEFORE EXPOSING THE PROJECT, SADLY DUE TO SOME PHOTOS WERE ERASED WE COULDN’T ATTACH MORE]

8. PROGRAMMING CODE

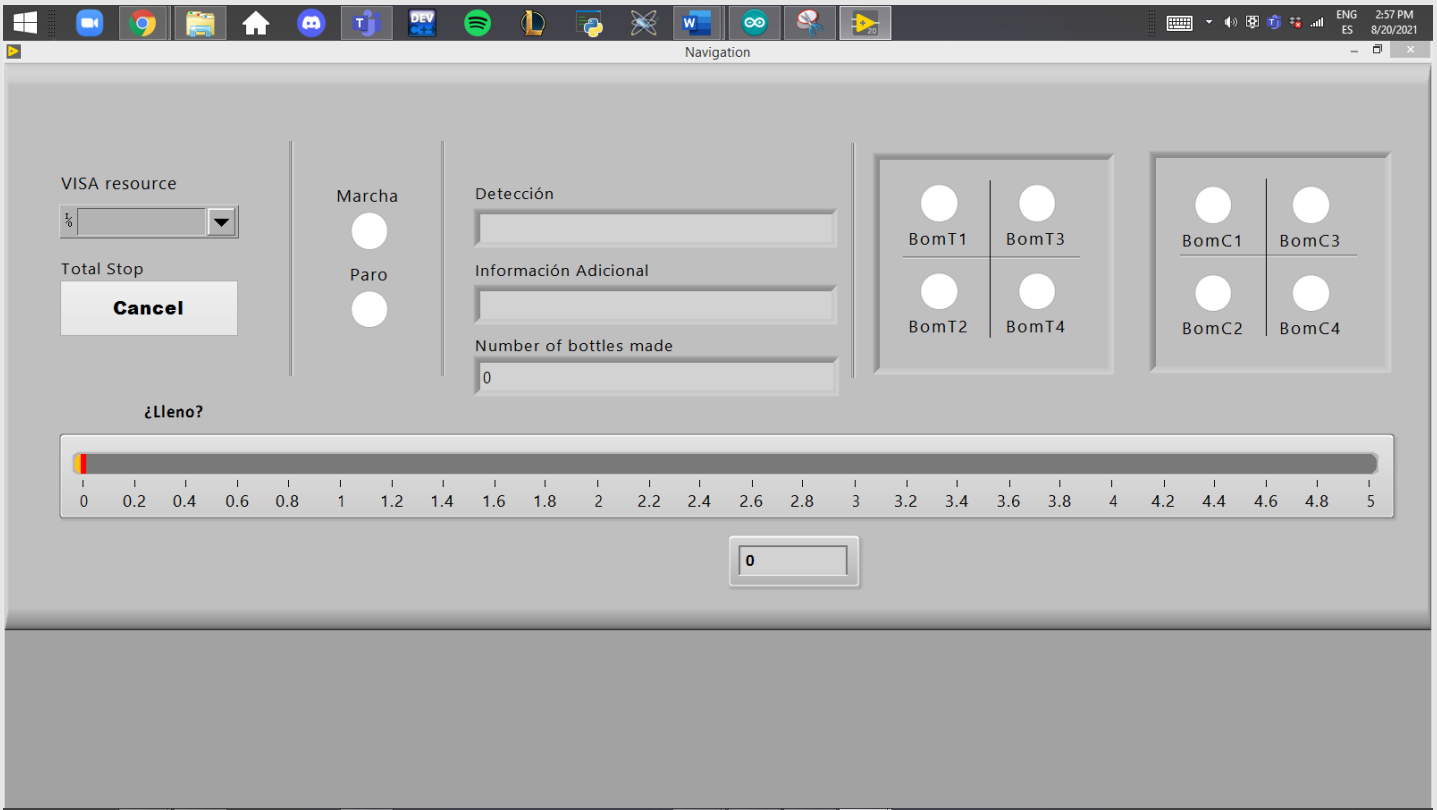
A) ARDUINO

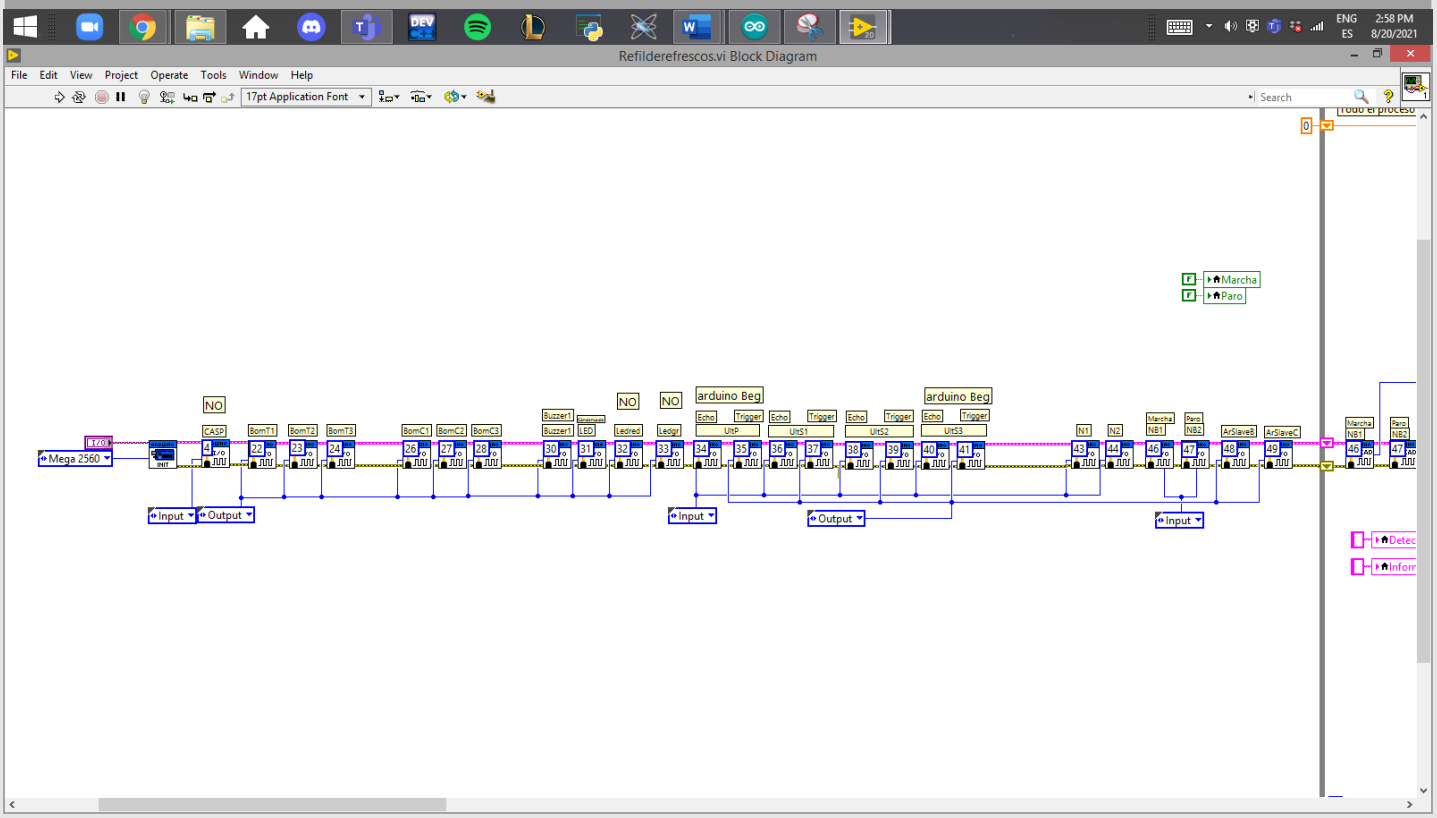
```
#include <Stepper.h>
#include <Servo.h>
Servo Base; //BASE CIRCULAR
const int stepsPerRevolution = 16;  // change this to fit
the number of steps per revolution
Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11);

// PINES DE LA PARTE DE SELLADO///////////////
// MOTOR A
int enA= 2; // PIN FOR ENA output in bridge H
int in1= 3; // PIN FOR in1 output in bridge H
int in2= 4; // PIN FOR in2 output in bridge H

// MOTOR B
int enB= 5; // PIN FOR ENB output in bridge H
int in3= 6; // PIN FOR in3 output in bridge H
int in4= 7;  // PIN FOR in4 output in bridge H
//////////////////////////////////////////////
//PINES
int NB1 = 6; //Boton de Inicio
int NB2 = 7;// Boton de Paro
int BomT1 = 22;
int BomT2 = 23;
int BomT3 = 24;
//NO HAY PIN 25
int BomC1 = 26;
int BomC2 = 27;
int BomC3 = 28;
//NO HAY PIN 29
int BuzzerNB = 30; //Buzzer Nivel Bajo
int LEDNB = 31; //LED Nivel Bajo
//NO HAY PIN 32
//NO HAY PIN 33
int UltPEcho = 34;
int UltPTrig = 35;
int UltS1Echo = 36;
int UltS1Trig = 37;
int UltS2Echo = 38;
int UltS2Trig = 39;
int UltS3EchoYN = 40;
int UltS3TrigYN = 41;
//NO HAY PIN 42
int N1 = 43; //Sensor de Nivel
//NO HAY PIN 44
//NO HAY PIN 45
//NO HAY PIN 46
//NO HAY PIN 47
int ArSlaveB = 48; //Banda
int ArSlaveC = 49; //Sellado
int Tcrt5000 = A0; // parte análoga



//VARIABLES Y CONSTANTES
int Time;
int Distance;
int salir;
int salir1;
int Time1;
int Distance1;
int Time2;
int Distance2;
int ok1;
int ok2;
int analogValue = 0;
int Time3;
int Distance3;
int contador = 0;
int q = 0;
int salirllenado;

void setup()
{
  // set the speed at 60 rpm:
myStepper.setSpeed(200);
 // Iniciamos el monitor serie para mostrar el resultado
  Serial.begin(9600);
  //Servo de parte de sellado
  Base.attach(13);
  Base.write(0);
  //L298N PARTE DE SELLADO
  pinMode(enA, OUTPUT);   // the ENA is declared as
output
  pinMode(enB, OUTPUT);  // the ENB is declared as
output
  pinMode(in1, OUTPUT);     // the in1 is declared as
output
  pinMode(in2, OUTPUT);   // the in2 is declared as
output0
  pinMode(in3, OUTPUT);     // the led 3 is declared as
output0
  pinMode(in4, OUTPUT);    // the led 4 is declared as
output
  ////////////////////////////////////
  pinMode(NB1, INPUT);//Boton de Inicio
  pinMode(NB2, INPUT);// Boton de Paro
  pinMode(UltPEcho, INPUT);
  pinMode(UltPTrig, OUTPUT);
  pinMode(UltS1Echo, INPUT);
  pinMode(UltS1Trig, OUTPUT);
  pinMode(UltS2Echo, INPUT);
  pinMode(UltS2Trig, OUTPUT);

  pinMode(UltS3EchoYN, INPUT);
  pinMode(UltS3TrigYN, OUTPUT);
  pinMode(BuzzerNB, OUTPUT);
  pinMode(LEDNB, OUTPUT);
  pinMode(BomT1, OUTPUT);
  pinMode(BomT2, OUTPUT);
  pinMode(BomT3, OUTPUT);
  pinMode(BomC1, OUTPUT);
  pinMode(BomC2, OUTPUT);
  pinMode(BomC3, OUTPUT);
  pinMode(N1, INPUT); //Sensor de Nivel
}
```

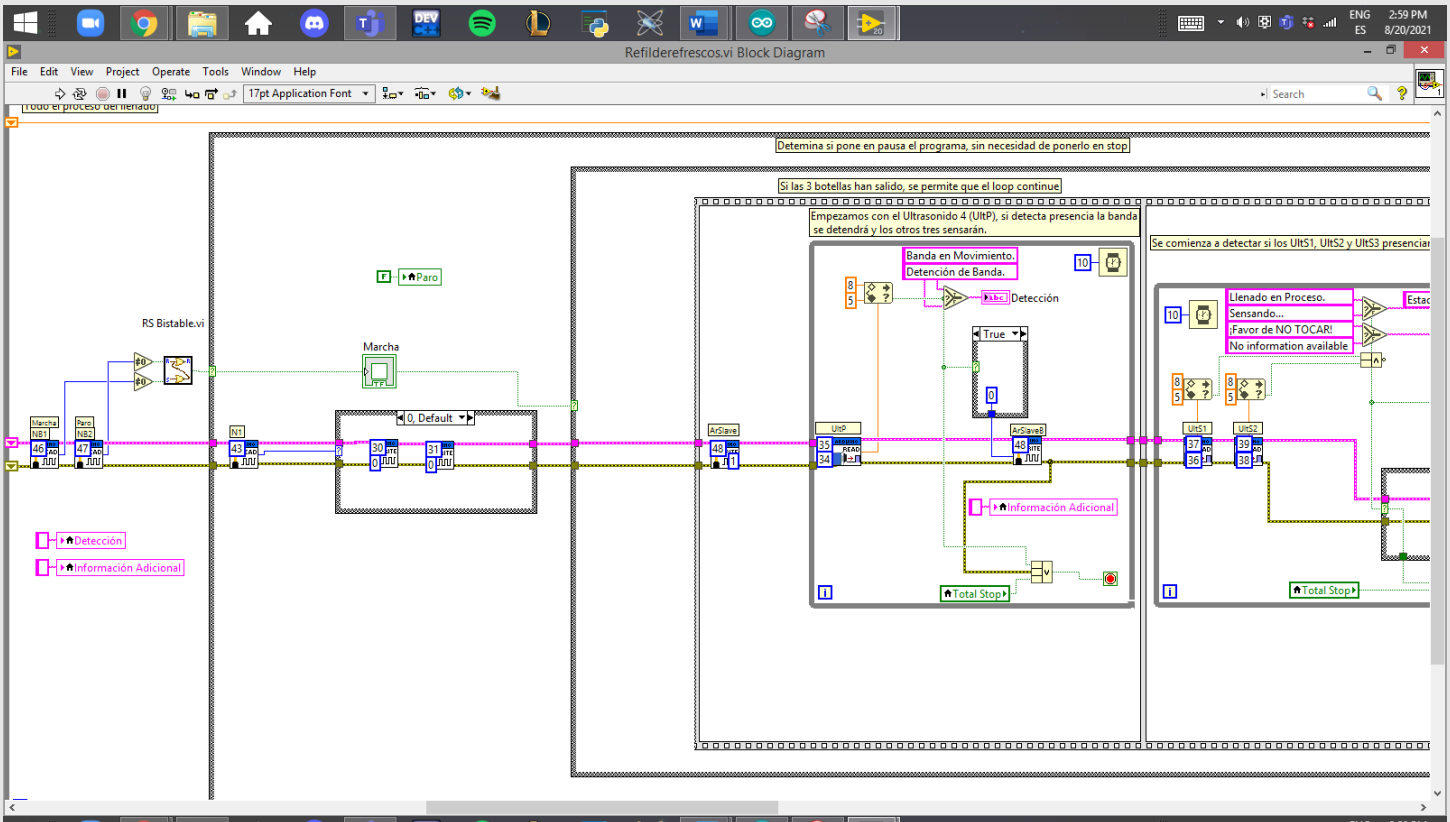
```
void loop()
{
 digitalWrite(BomT1, HIGH);
 digitalWrite(BomC1, HIGH);
 digitalWrite(BomT2, HIGH);
 digitalWrite(BomC2, HIGH);
 digitalWrite(BomT3, HIGH);
digitalWrite(BomC3, HIGH);
 while (Serial.available() == 0);
 char val = Serial.read();
  if (val=='1'){ //BOTON DE MARCHA
 Serial.println("INICIA PROCESO");
     do //EMPIEZA EL PROGRAMA IMPORTANT
   {
        do //sensor de nivel
   {
     if (digitalRead(N1) == HIGH) //considera si el tanque
está vacío
     {
      digitalWrite(LEDNB,HIGH); //ENCIENDE LED
VERDE
       digitalWrite(BuzzerNB,HIGH); //ENCIENDE
BUZZER
       delay (10);
     }
     } while (digitalRead(N1) == HIGH); //cuando detecte
que esté lleno se termina CONDICION FALSA
     digitalWrite(LEDNB,LOW); //APAGA LED VERDE
     digitalWrite(BuzzerNB,LOW); //APAGA LED
BUZZER
     delay (10);
//INICIA LA PRIMERA ETAPA**********************

     Serial.println ("Initializaing movement /n");
     Serial.println ("Put the bottles on the PL");
     Serial.println ("First stage: Sensing and Filling");
     delay (10);
     myStepper.step(1250);; //Linea de producción
empieza a moverse
     Serial.println("The PL has been activated");
     Serial.println ("The principal Ultrasonic sensor begins
working");
     do
     {
     salir = 0;
     digitalWrite (UltPTrig, HIGH);
     delay(1);
     digitalWrite (UltPTrig, LOW);
     Time = pulseIn (UltPEcho, HIGH);
     Distance = Time / 58.2;
     Serial.println("Production line is in movement...
Distance:");
     Serial.println(Distance);
     Serial.print(" cm");
     delay (350);
     if (Distance >= 5 && Distance <= 8)
     {
       myStepper.step(0); // se detiene la pl
       Serial.println ("PL has stopped");
       Serial.println ("A bottle is in front of the UltP");
       salir = 1;
     }
     } while (salir == 0); //Si está dentro del rango se
detiene
}while(val=='0');
}
 if (val=='0')
  {
      digitalWrite(BomT1, HIGH);
      digitalWrite(BomC1, HIGH);
      digitalWrite(BomT2, HIGH);
      digitalWrite(BomC2, HIGH);
      digitalWrite(BomT3, HIGH);
      digitalWrite(BomC3, HIGH);
      myStepper.step(0);
      Serial.println("The PL has been activated");
      Serial.println ("Auto-Plenus is in stop position");
      delay (1000);
  }
  // SEGUNDA ETAPA
       myStepper.step(0); //Linea de producción SE
DETIENE
     Serial.println ("The other two Ultrasonic sensors start
working");
     do
      {
      salir1 = 0;
      //ULTRASONIC2********
      //ULTRASONIC3********
     digitalWrite (UltS2Trig, HIGH);
      delay(1);
      digitalWrite (UltS2Trig, LOW);

      Time2 = pulseIn (UltS2Echo, HIGH);
      Distance2 = Time2 / 58.2;
      Serial.println("Distance:" );
      Serial.print(Distance2);
      Serial.print(" cm");
      delay (200);

      if (Distance2 >= 4 && Distance2 <= 8)
   {
        Serial.println ("Ultrasonic Sensor 3 has detected");
        salir1 = 1;
      }
      } while (salir1 == 0); //Si está dentro del rango se
detiene
      delay(3000);
      do
      {
      salirllenado = 0;
      int analogValue = analogRead(0);
      Serial.print("The ANALOG VALUE is: ");
      Serial.println(analogValue);
      digitalWrite(BomT1, LOW);
      digitalWrite(BomC1, LOW);
      digitalWrite(BomT2, LOW);
      digitalWrite(BomC2, LOW);
      digitalWrite(BomT3, LOW);
      digitalWrite(BomC3, LOW);
      delay(15000);
      digitalWrite(BomT1, HIGH);
      digitalWrite(BomC1, HIGH);
      digitalWrite(BomT2, HIGH);
      digitalWrite(BomC2, HIGH);
```

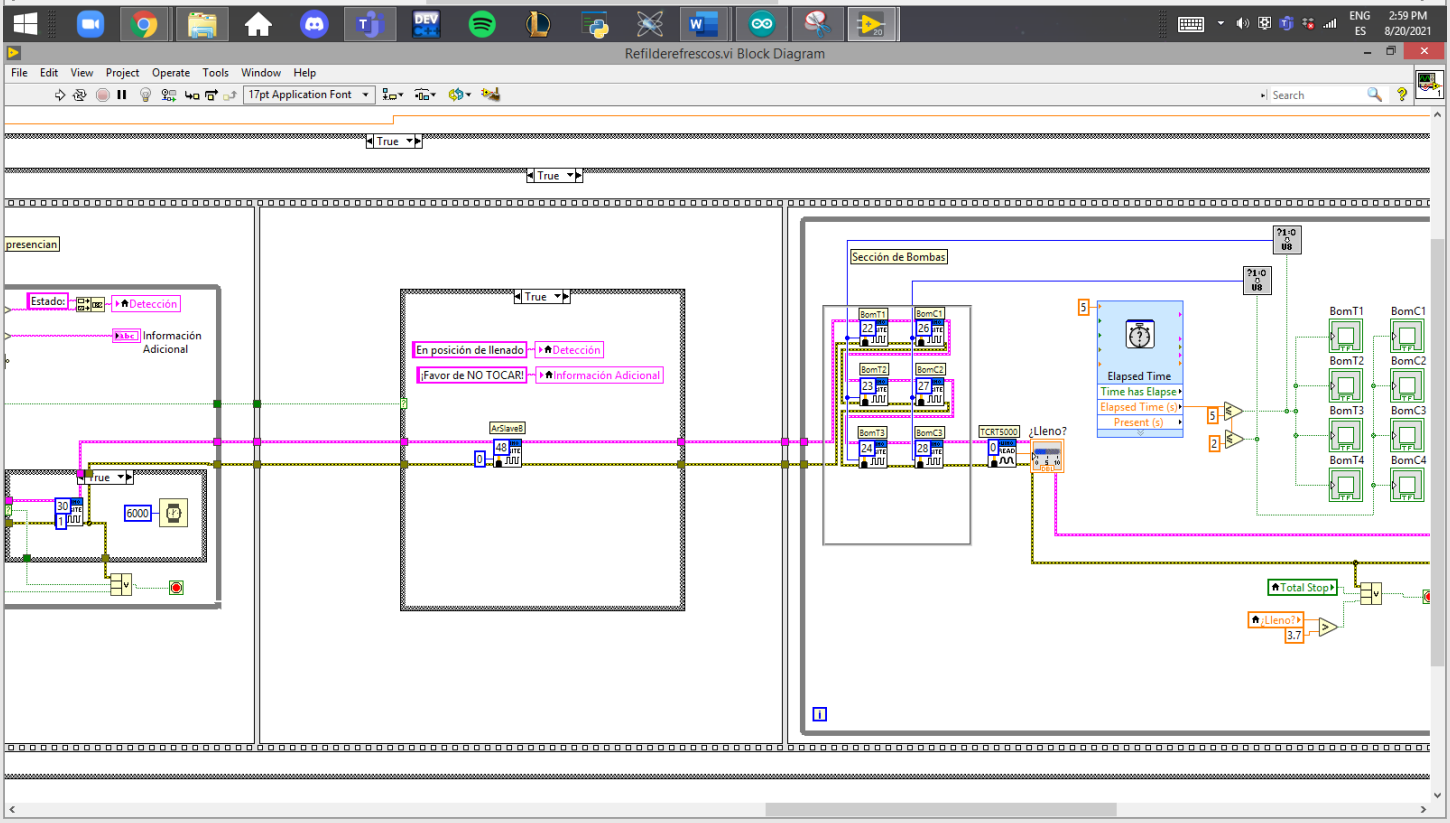
```
    digitalWrite(BomT3, HIGH);
    digitalWrite(BomC3, HIGH);
    delay(2000);
    if (analogValue >= 550)
    {
      Serial.println("The 3 bottles have been filled");
      salirllenado = 1;
    }
    } while(salirllenado = 0);

    // INICIAN LA TERCERA ETAPA**********

    Serial.println("Next stage: Sealing");
  q = 0;

    do    {
    digitalWrite (UltS3TrigYN, HIGH);

    delay(350);
    digitalWrite (UltS3TrigYN, LOW);
    Time3 = pulseIn (UltS3EchoYN, HIGH);
    Distance3 = Time3 / 58.2;
    Serial.println("Distance3333:");
    Serial.print(Distance3);
    Serial.print(" cm");
    myStepper.step(1250);
    delay (350);
    if (Distance3 >= 5 && Distance <= 8)
     {
    contador++;
    q++;
    myStepper.step(1250); //BANDA ACTIVADA
    Serial.println ("Bottle passing into the stage of
SEALING");
    delay(2000);
     }
    if(contador>=2)
    {
    contador=0;
    myStepper.step(0); //BANDA DESACTIVADA

    // INICIA ETAPA DE SELLADO

  Base.write(0);
  Serial.println("movimiento entrar al proceso");
  delay(500);
  Base.write(55);
  Serial.println("colocación de tapa");
  delay(5500);
  Base.write(136);
  delay(100);
  Base.write(136);
  Serial.println("Sellado de tapa");
 digitalWrite(enA, HIGH);  // habilita motor A (giro en un
sentido)
 digitalWrite(in1, LOW); // IN1 a 0
 digitalWrite(in2, HIGH);  // IN2 a 1
 digitalWrite(enB, HIGH);  // habilita motor B (giro en
sentido contrario)
 digitalWrite(in3, HIGH);  // IN3 a 1
 digitalWrite(in4, LOW); // IN4 a 0
 delay(5500);
 digitalWrite(enA, LOW); //apagado de motorreductores
 digitalWrite(in1, LOW);
 digitalWrite(in2, LOW);
 digitalWrite(enB, LOW);
 digitalWrite(in3, LOW);
 digitalWrite(in4, LOW);
 Serial.println("The bottle has being sealed");

 Base.write(180);
 delay(5500);
 Base.write(0);
 digitalWrite(enA, LOW); //apagado de motorreductores
 digitalWrite(in1, LOW);
 digitalWrite(in2, LOW);
 digitalWrite(enB, LOW);
 digitalWrite(in3, LOW);
 digitalWrite(in4, LOW);
 delay(5500);
 Serial.println("The process has ended");
 }
    // SELLADO WAIT
    else
    {
    Serial.println("ESPERANDO BOTELLA");
     // SELLADO DESACTIVADO
    }
    } while (q == 0 || q == 1 || q == 2);
      if (val=='0')
 {
  Serial.println ("Auto-Plenus is in stop position");
  delay (1000);
 }

}
```

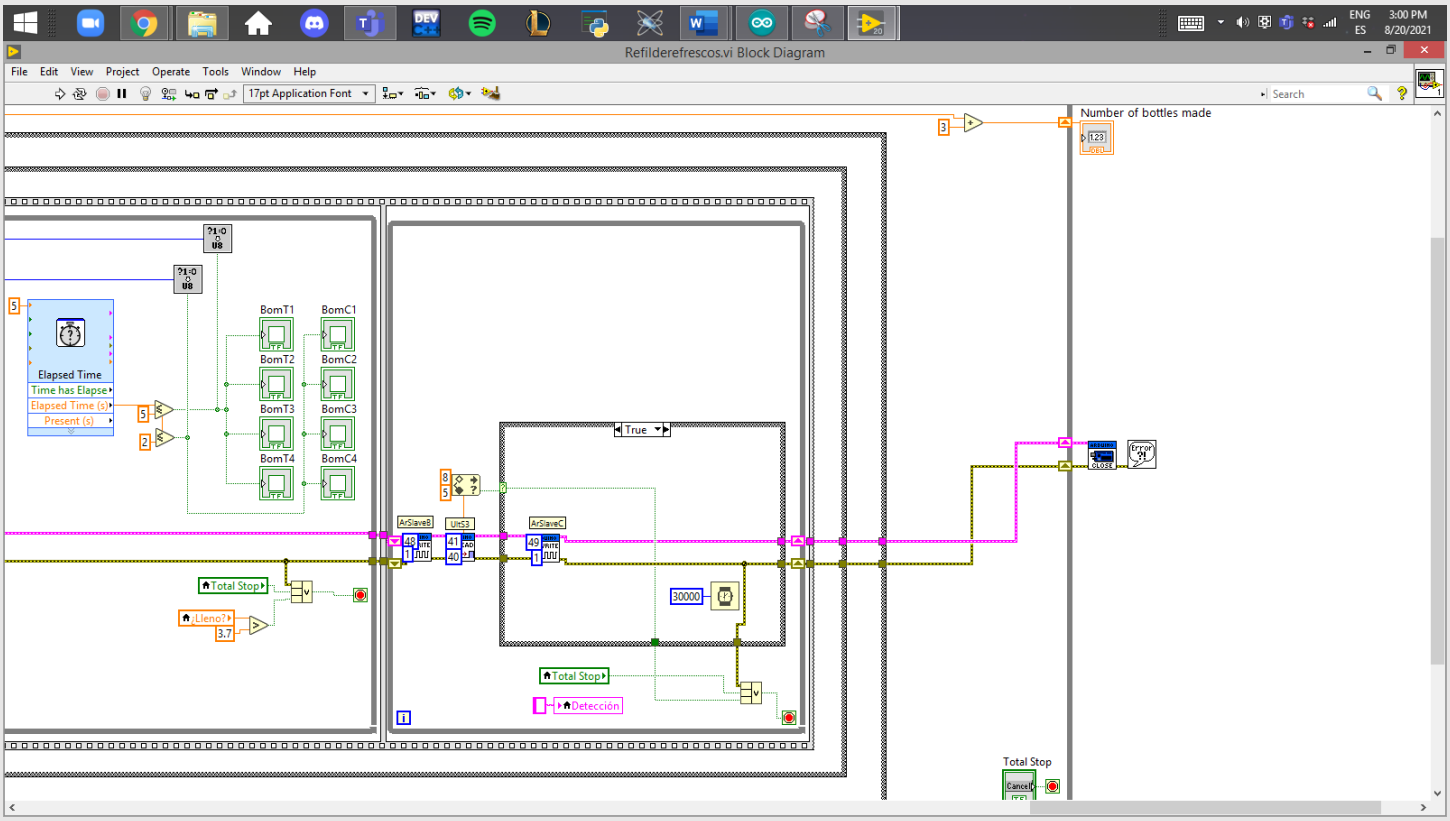
B) LABVIEW


Navigation
VISA resource
Total Stop
Cancel
Marcha
Paro
Detección
Información Adicional
Number of bottles made
0
BomT1
BomT3
BomT2
BomT4
BomC1
BomC3
BomC2
BomC4
¿Lleno?
0 0.2 0.4 0.6 0.8 1 1.2 1.4 1.6 1.8 2 2.2 2.4 2.6 2.8 3 3.2 3.4 3.6 3.8 4 4.2 4.4 4.6 4.8 5
0


Refilderefrescos.vi Block Diagram
File Edit View Project Operate Tools Window Help
17pt Application Font
Search
Marcha
Paro
NO
NO
NO
arduino Beg
arduino Beg
Mega 2560
Input
Output
Input
Output
Input


Refilderefrescos.vi Block Diagram
File Edit View Project Operate Tools Window Help
17pt Application Font
Search
Todo el proceso del llenado
Detemina si pone en pausa el programa, sin necesidad de ponerlo en stop
Si las 3 botellas han salido, se permite que el loop continue
Empezamos con el Ultrasonido 4 (UltP), si detecta presencia la banda se detendrá y los otros tres sensarán.
Se comienza a detectar si los UltS1, UltS2 y UltS3 presencian
RS Bistable.vi
Paro
Marcha
Banda en Movimiento.
Detención de Banda.
Detección
Llenado en Proceso.
Sensando...
¡Favor de NO TOCAR!
No information available
Información Adicional
Total Stop
0, Default


Refilderefrescos.vi Block Diagram
File Edit View Project Operate Tools Window Help
17pt Application Font
Search
presencian
Estado
Detección
Información Adicional
En posición de llenado
¡Favor de NO TOCAR!
Sección de Bombas
Elapsed Time
Time has Elapse
Elapsed Time (s)
Present (s)
¿Lleno?
Total Stop


Refilderefrescos.vi Block Diagram
File Edit View Project Operate Tools Window Help
17pt Application Font
Search
Number of bottles made
Elapsed Time
Time has Elapse
Elapsed Time (s)
Present (s)
BomT1
BomT2
BomT3
BomT4
BomC1
BomC2
BomC3
BomC4
True
ArSlaveB
ArSlaveC
Total Stop
¿Lleno?
3.7
30000
Detección
Total Stop

9. TESTS AND FAILURES

During the developing of the Auto-Plenus machine some problems were confronted with patience.

Some of the problems and adequations were:

- The power wasn't enough for 8 bombs and the motor NEMA 34, all the amperage wasn't enough. This was solved with withdrawing 2 bombs that belonged to the filling part (one from the containers and another one of the syrup containers.

- The room's light wasn't powerful enough to help the TCRT5000 IR sensor to sense the reflecting light towards itself. It was solved by placing a little Pixar lamp, right above the sensor.

- Due to the wrongdoing distribution of the wooden cover (the one that covers the syrup containers), it resulted difficult to access to that part to fill them, but by placing a tube and on the entry of the liquid, the problem was solved.

- There was a problem by trying to use some Arduinos as slaves and Masters, we investigated and we had to use a I^2C communication way which the team members didn't know how to use it, as a result of the short amount of time that was left, the team gather to create the Arduino's entirely programming code.

- The code of LabView wasn't functional due to the previous reason and also due to the flip flop function, and a overload of the PC memory… The PC used was blowing up.

- A pump was fried due to the mineral water. This pump was replaced immediately. A photo is place right here:

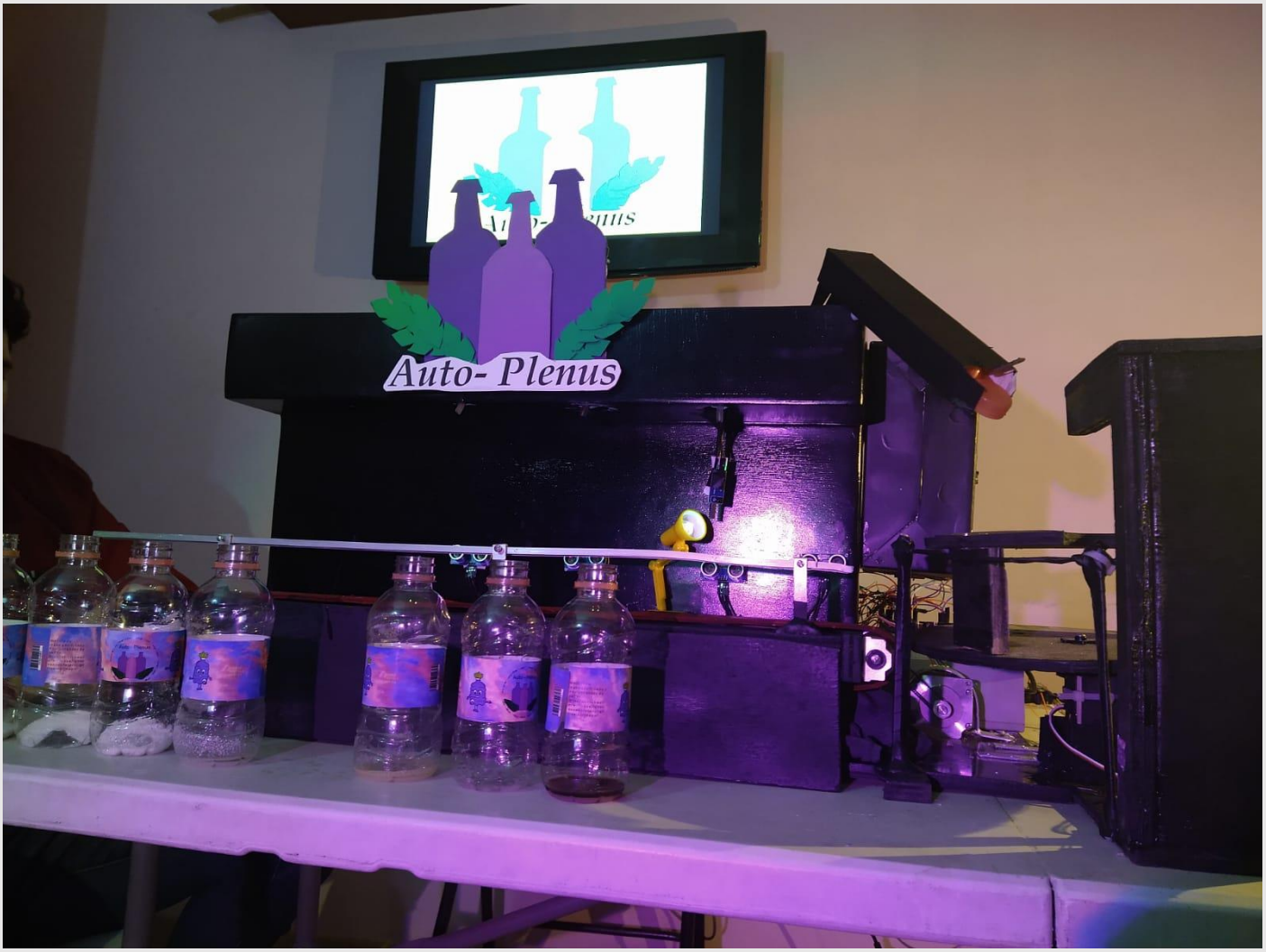
Figure 8. If you watch carefuly, you will appreciate the orange burned part.

- The real-life buttons didn't work, so… besides keeping them we placed some stop and running numbers in the Arduino's code, pretending to be buttons.

```
void loop()
{
 digitalWrite(BomT1, HIGH);
 digitalWrite(BomC1, HIGH);
 digitalWrite(BomT2, HIGH);
 digitalWrite(BomC2, HIGH);
 digitalWrite(BomT3, HIGH);
 digitalWrite(BomC3, HIGH);
 while (Serial.available() == 0);
 char val = Serial.read();
  if (val=='1'){ //BOTON DE MARCHA
  Serial.println("INICIA PROCESO");
```

```
 if (val=='0')
{
  Serial.println ("Auto-Plenus is in stop position");
  delay (1000);
}
```

10. PROTOTYPE



LINK OF THE VIDEO: https://youtu.be/2sa_Mo6SrPI

11. BIBLIOGRPHIES

NA [Nevon Projects]. (2018, September 5). *Automatic Bottle Filling & Capping Machine Using PLC* [Video]. YouTube. https://www.youtube.com/watch?v=JdXzMI1PXcs&list=PLHy0pyPHjYahF4hPqvSab6AllA-yPd4s3&index=2

NA [DIMEI Facultad Ingeniería UNAM]. (2017, December 12). *Sistema de envasado de botellas* [Video]. YouTube. https://www.youtube.com/watch?v=W3Y9RtnjLCM&list=PLHy0pyPHjYahF4hPqvSab6AllA-yPd4s3&index=3

NA [Muy Fácil De Hacer]. (2016, March 12). *Proyectos | Cinta Transportadora Casera (muy fácil de hacer)* [Video]. YouTube. https://www.youtube.com/watch?v=7UsmJgHU6wk&feature=youtu.be

NA [Robojax]. (2019, July 27). *Control Position and Speed of Stepper motor with L298N module using Arduino* [Video]. YouTube. https://www.youtube.com/watch?v=cYTICj4DWYc&feature=youtu.be

NA [Bitwise Ar]. (2017, March 8). *Arduino desde cero en Español - Capítulo 5 - HC-SR04 Sensor Ultrasónico (y uso del Monitor Serial)* [Video]. YouTube. https://www.youtube.com/watch?v=mlw3APOUt8U&list=PLBzzT977UjU-gNi4K2ugXoyJ0OA_b2ESb&index=5

NA [Bitwise Ar]. (2017, June 18). *Arduino desde cero en Español - Capítulo 19 - L298N Controlador de Motores DC (y velocidad PWM)* [Video]. YouTube. https://www.youtube.com/watch?v=63aitq3KTaI&list=PLBzzT977UjU-gNi4K2ugXoyJ0OA_b2ESb&index=8

NA [Bitwise Ar]. (2017, March 15). *Arduino desde cero en Español - Capítulo 6 - Servomotor (conexión, modelos, ajustes para uso óptimo)* [Video]. YouTube. https://www.youtube.com/watch?v=6bPVZg17vKc&list=PLBzzT977UjU-gNi4K2ugXoyJ0OA_b2ESb&index=7

Brian Valdivieso. (2016, August 17). *Curso Arduino - Capítulo 5: Estructuras de control* [Video]. YouTube. https://www.youtube.com/watch?v=Abp4VrjDycc&list=PLBzzT977UjU-gNi4K2ugXoyJ0OA_b2ESb&index=11

NA [vt en linea]. (2017, June 19). *Como usar el driver controlador de motores L298N* [Video]. YouTube. https://www.youtube.com/watch?v=c0L4gNKwjRw

NA [The Action Lab]. (2020, December 10). *This Tesla Valve Straw Only Works In One Direction* [Video]. YouTube. https://www.youtube.com/watch?v=sxvIQMnKhIg&t=212s

NA [ICV Creative]. (2020, April 18). *How to Make Automatic Hand Washing Machine || Without Arduino* [Video]. YouTube. https://www.youtube.com/watch?v=a4QHJmfp6q0&t=119s

NA [Integza]. (2019, May 26). *Tesla Valve 3D Printed* [Video]. YouTube. https://www.youtube.com/watch?v=ct0tljBtjG0

NA [Electrotec]. (2015, December 18). *Uso del sensor de NIVEL DE LIQUIDO y una MINIBOMBA* [Video]. YouTube. https://www.youtube.com/watch?v=bM7fvJvThXA

NA, N. A. (2020, March 21). *¿Qué tipo de sensor de proximidad utilizar y cuáles son sus aplicaciones?* Sepia Automatización Industrial. https://www.sepia.mx/que-tipo-de-sensor-de-proximidad-utilizar-y-cuales-son-sus-aplicaciones/

12. GUAARANTEE

AUTO PLENUS S.A. DE C.V., guarantees this product for a period of **one year** for all its parts and labor, against any manufacturing and operation defect from the date of delivery to the final consumer.

To make this warranty effective, the only requirement is the presentation of the product and the corresponding policy (or proof of purchase), duly stamped by the establishment that sold it. The warranty may be validated at the place where the product was purchased or through the authorized service centers attached to this document.

PRODUCT DATA

Description:_____

_____

Model: _____ No. Series: _____

Purchase date: _____

Date: _____

To make the warranty valid, it is necessary to present the receipt or purchase note, take the product intact to the establishment where it was purchased or directly to the factory offices, (without trying to repair it yourself and without opening the warranty seals), so that your product can be replaced or if you prefer, the amount paid for the product can be refunded.

This warranty is valid only during the first six months after the purchase of the product..

Void Warranty: The warranty will remain valid within the first twelve months of the purchase of the product, provided that the product has not been modified, opened, disassembled or repaired in any of its components.

- It has not been improperly used.
- It has not been tampered with improperly.
- It has not suffered deterioration or damage caused by mishandling.
- Has not been damaged by external agents for which the user is responsible (water, acids and other substances).

13. MAINTENANCE

a) Sensors

For the sensors, the most advisable thing is to give them a good cleaning every certain time or to give them preventive maintenance by cleaning them after a certain time of use of the system.

It is also recommended to calibrate them at least once a year, this to avoid possible failures in the device and in the ideal operation oif the machine. When cleaning, make sure not to damage the surface. Water droplets or dirt may adversely affect the proper functioning.

Note: Light accumulations of dust are not serious.

b) Motors

Early warning signs of potential problems in motors:

- Overcurrent faults: this condition occurs when the servomotors exceed the current maximum limit.

- Feedback errors: these problems can be caused by loss of feedback signal from your tachometer, encoder or resolver.

- Position errors: these are caused by mechanical problems or feedback signals.

- Overheating / odor: this is a common condition in servo motors that are exposed to excessive amounts of oil or coolant spray, which can contaminate the servo stator, armature, or reaction unit.

Preventive maintenance consists of checking, cleaning and readjusting the various elements of a servo motor. The maintenance of servomotors is advisable to avoid breakdowns, since we extend the useful life of the motor and protect it against future breakdowns.

c) Pumps

Daily maintenance

- Check that the speed of the water pump corresponds to the output.

- Examine the flanges for leaks, cracks, wear or rusted parts that could jeopardize the operation.

Monthly maintenance

- Inspect and repair seals.

- Check the elevation of the water pump in relation to its base.

- Check couplings.

- Clean filters.

- Remove accumulation of dust and dirt.

14. PROJECT EXPLANATIONS FOR THE PROFESSOR

1**. Stop button:** The stop button is implemented as the start button, through the serial monitor, the example of the start button is shown in minute 1:14 of the project video, the start button was not shown, so we will specify its operation by quoting the programming code as shown in the image, when it receives a "0" it will turn off the operation of the components giving the STOP instruction.

```
void loop()
{
 digitalWrite(BomT1, HIGH);
 digitalWrite(BomC1, HIGH);
 digitalWrite(BomT2, HIGH);
 digitalWrite(BomC2, HIGH);
 digitalWrite(BomT3, HIGH);
 digitalWrite(BomC3, HIGH);
 while (Serial.available() == 0);
 char val = Serial.read();
  if (val=='1'){ //BOTON DE MARCHA
  Serial.println("INICIA PROCESO");
```

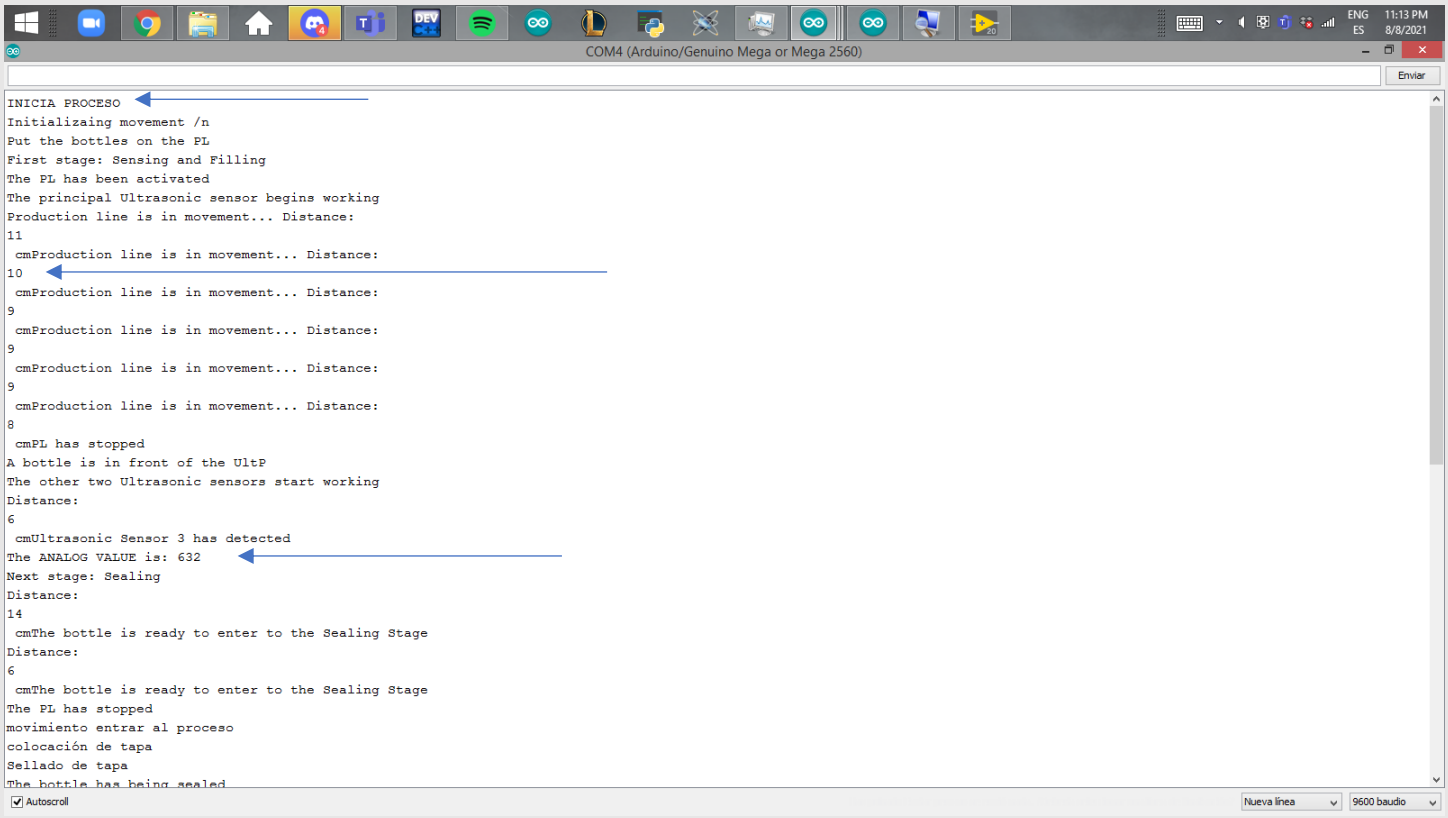
```
 if (val=='0')
{
  Serial.println ("Auto-Plenus is in stop position");
  delay (1000);
}
```

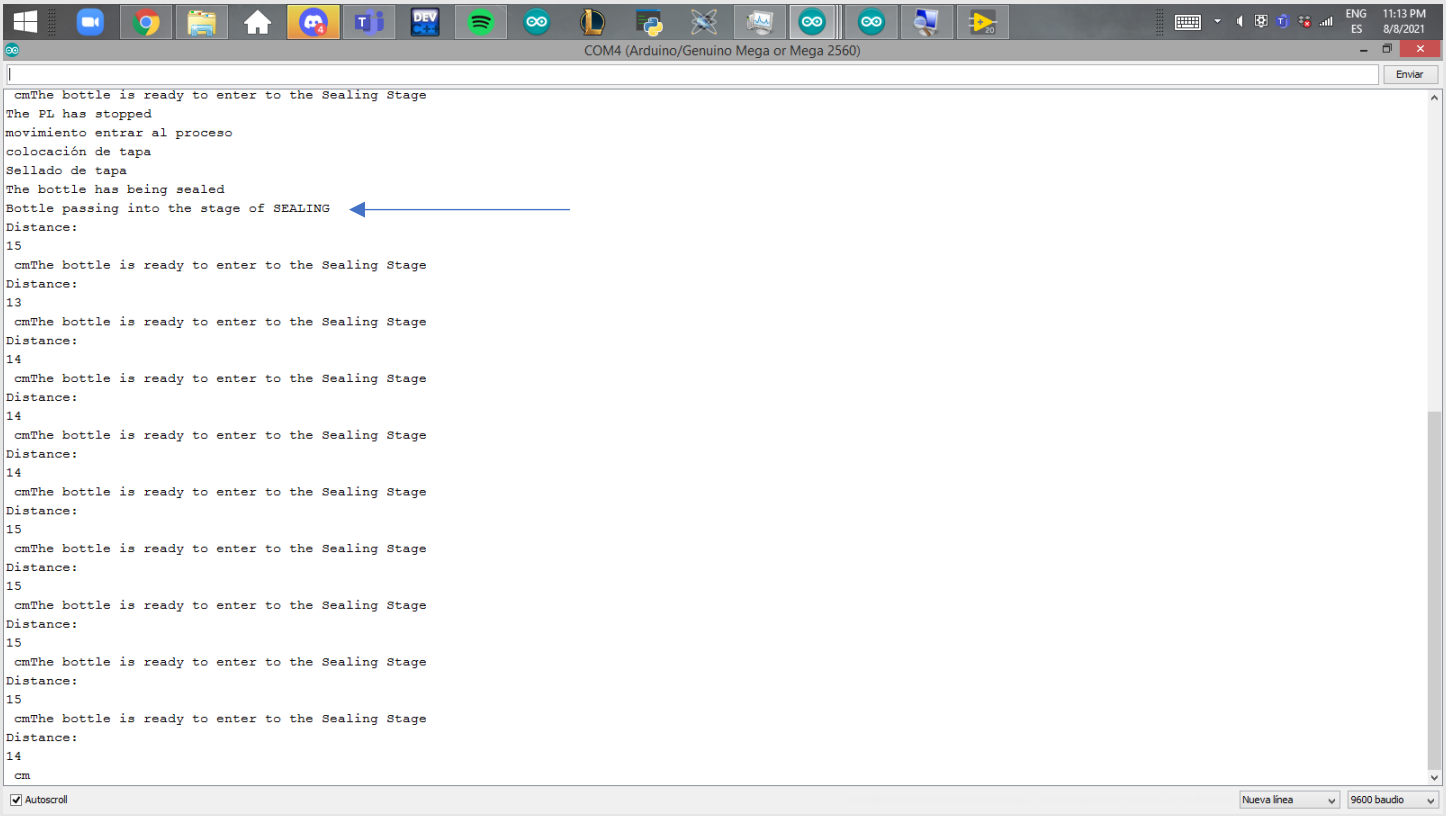
2. Data acquisition: For data acquisition we use again the serial monitor, in the video you can see the messages from the mount of pressing the "1" key which is the startup starts to work and in turn sending messages of what is happening for example here some images of the code and the minutes in the video where it is shown are 1:14, 1:45, 2:08, what was missing to show already by the end of its operation we spent flipping the camera but it will be shown in the photographs.

```
 char val = Serial.read();
 if (val=='5'){ //BOTON DE MARCHA
 Serial.println("The PL has stopped");


 Base.write(0);
 Serial.println("movimiento entrar al proceso");
 delay(500);

 Serial.println("colocación de tapa");
 delay(5500);
```

```
 Serial.println("Sellado de tapa");
digitalWrite(enA, HIGH);  // habilita m
digitalWrite(in1, LOW); // IN1 a 0

 digitalWrite(in4, LOW);
 Serial.println("The bottle has being sealed");
 Base.write(180);

delay(5500);
    Serial.println ("Bottle passing into the stage of SEALING");
    delay(2000);
      }
```

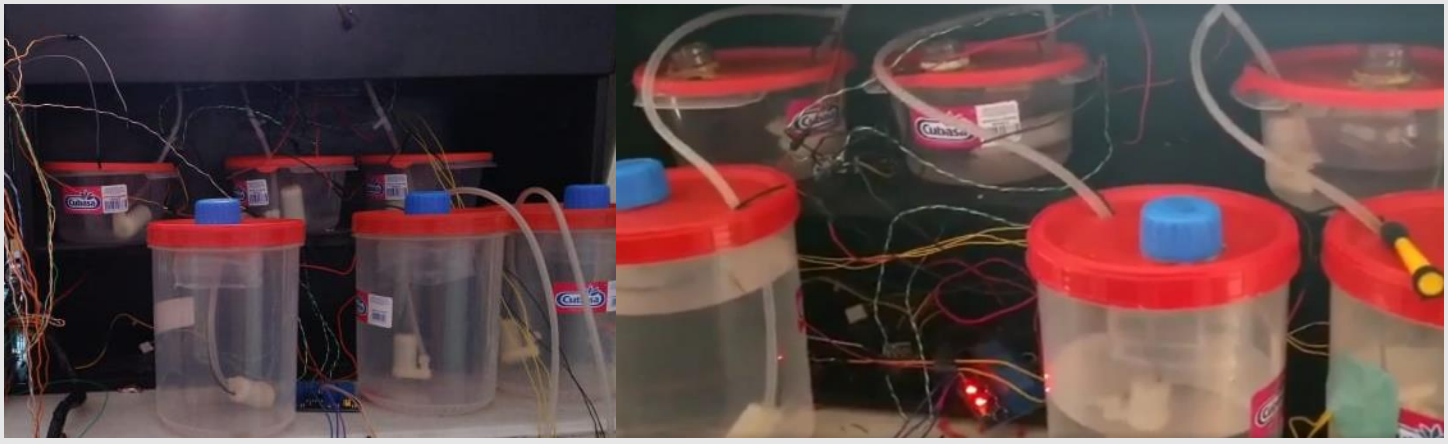
[The instructions were cut form the main program]





[You can compare these images with the ones of the video].

3. Relays and pumps: In this section which is the back part of the project shown in the video, the connections are hidden since we wanted to maintain a certain visual cleanliness and connections, this will be shown with the following images where we see the way in which the containers, pumps and relays are arranged along with the connections of each one. We used 6 containers because the big ones we could call them as the water collection containers and the other 3 smaller ones as the direct re-fill containers.



CONTACT US AT:

Evergreen Ave. No. 742, Springfield

(+52) 81-24-32-65-36

product@autoplenus.mx

Auto- Plenus