

Clasificadora de Basura por Colores

Juan Pablo Higueta Echavarría, Melissa Galeano Ruiz, Luis Fernando Arango López, Juan David Roldán

Ríos, Julián Andrés Moreno Lotero

jpablo.higueta@udea.edu.co; melissa.galeanor@udea.edu.co; luis.arango6@udea.edu.co;
juan.roldan1@udea.edu.co; julian.moreno@udea.edu.co

Docentes:

Pedro León Simancas, Jaime Alonso Montoya López

Facultad de ingeniería, Universidad de Antioquia

Departamento de Ingeniería Mecánica

Resumen: Se ha desarrollado un sistema de clasificación de desechos basado en colores que emplea inteligencia artificial en combinación con una cámara convencional. Se fabrica un prototipo a menor escala que incorpora una estructura construida con paneles de fibra de densidad media (MDF), una cinta transportadora en látex, y diversos actuadores tales como pistones neumáticos y un motor eléctrico. El control y la gestión del sistema se llevan a cabo a través de un Controlador Lógico Programable (PLC) de la marca Siemens, utilizando el software TIA Portal. Este PLC recibe como entrada la información proporcionada por el software de reconocimiento de colores mediante inteligencia artificial, desarrollado en Python y Arduino. La clasificación se realiza principalmente en base a los colores de los residuos, identificando tonalidades de verde, azul y rojo. Sin embargo, el sistema tiene la capacidad de ser adaptado para clasificar una gama más amplia de colores, de acuerdo con los estándares establecidos en materia de gestión de residuos y saneamiento.

Palabras claves: Inteligencia artificial, visión por computadora, automatización, clasificación, medio ambiente.

I. INTRODUCCIÓN

En la era actual de la automatización y la tecnología avanzada, la eficiencia en la gestión de residuos se ha convertido en una prioridad crucial. En este contexto, el proyecto integrador de la clasificadora de basura por colores emerge como una solución innovadora para abordar los desafíos asociados con la gestión de residuos sólidos. Esta iniciativa propone diseñar, implementar y evaluar una clasificadora automatizada que utiliza tecnología de visión por computadora, inteligencia artificial y algoritmos de aprendizaje automático para clasificar los desechos en categorías específicas según su color. La clasificadora de basura por colores se diseñará para reconocer y separar eficientemente los distintos tipos de desechos, facilitando así la posterior gestión y reciclaje de manera más efectiva.

A lo largo de este informe, se abordarán los aspectos técnicos, conceptuales y prácticos relacionados con el desarrollo de la clasificadora, incluyendo la selección de sensores y cámaras, la implementación de algoritmos de visión por computadora, y la integración de un sistema de control.

Este proyecto busca no solo proporcionar una solución tecnológica innovadora, sino también fomentar la conciencia ambiental y promover prácticas más responsables en la gestión de residuos. La clasificadora de basura por colores representa un paso significativo hacia un

futuro más sostenible, donde la tecnología se utiliza para abordar los desafíos medioambientales de manera efectiva y eficiente.

II. OBJETIVOS

- El principal objetivo es diseñar y construir una clasificadora de basura por colores que utilice tecnología de visión por computadora y algoritmos de aprendizaje automático para una identificación precisa y eficiente de los desechos según su color.
- Se buscará implementar algoritmos avanzados de visión por computadora que puedan reconocer una amplia gama de colores teniendo presente el código de separación de residuos, permitiendo así una clasificación más precisa y versátil.
- Se pretende integrar un sistema de control eficaz que coordine los movimientos de los actuadores (como pistones neumáticos y motores eléctricos) para la separación adecuada de los desechos identificados por el sistema de visión por computadora.
- Este proyecto tiene como objetivo mostrar cómo la tecnología puede ser empleada como una herramienta eficaz para abordar los desafíos ambientales, promoviendo así un futuro más sostenible y alentando la adopción de soluciones similares en otros contextos.

III. DISEÑO CONCEPTUAL

El diseño conceptual de la clasificadora de basura por colores se fundamenta en la integración de tecnologías avanzadas para optimizar el proceso de clasificación de residuos sólidos. Este sistema innovador se compone de varios componentes interrelacionados que trabajan en conjunto para lograr una eficiente identificación y separación de los desechos, basándose principalmente en sus características cromáticas.

La clasificación de residuos sólidos es crucial para el reciclaje y el tratamiento adecuado de desechos por lo tanto es importante separar los desechos en categorías para ayudar a facilitar su recogida selectiva y su posterior tratamiento para el reciclaje o la disposición final adecuada, los residuos más comunes se clasifican en bolsas de color rojo, azul, verde y gris los cuales tienen la siguiente designación:

- **Bolsa verde:** Son para residuos orgánicos como restos de alimentos, residuos de jardinería, y materiales biodegradables.
- **Bolsa azul:** Son para residuos aprovechables, envases de cartón, periódicos, revistas, papel de oficina, cajas, en general productos reciclables.
- **Bolsa roja:** Son para residuos peligrosos, estas contienen desechos que podrían representar riesgos para la salud humana o el medio ambiente, como productos químicos tóxicos, materiales biológicos infecciosos, medicamentos vencidos, baterías, productos electrónicos viejos.
- **Bolsa gris:** Son para residuos no aprovechables, estas incluyen restos de comida no compostable, pañales desechables, artículos de higiene personal, ciertos tipos de plásticos no reciclables, materiales mixtos o contaminados que no pueden separarse para el reciclaje.



Figura 1. Código de Separación de Residuos por Colores.

Separar los desechos en estas categorías ayuda a facilitar su recogida selectiva y su posterior tratamiento para el reciclaje o la disposición final adecuada.

El diseño preliminar de la clasificadora se hizo con la separación de las bolsas descritas anteriormente y se presenta en la figura 2.

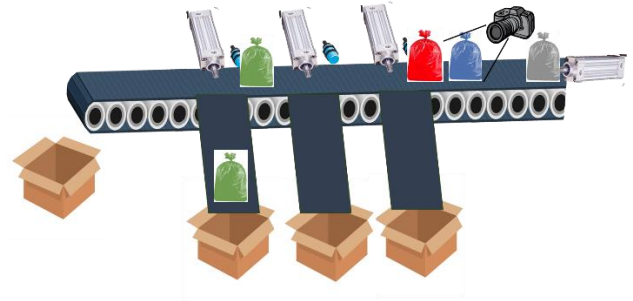


Figura 2. Diseño Preliminar del Montaje.

Este diseño se llevó a cabo utilizando las siguientes variables:

- *Sensores y Tecnología de Detección:*

La clasificadora estará equipada con 4 sensores capacitivos que son sensores que detectan la presencia o cambios en la proximidad de objetos basándose en cambios en la capacitancia, es decir, la capacidad de almacenar carga eléctrica entre dos conductores separados por un dieléctrico, y en el caso de estos sensores, los cambios en la capacitancia se utilizan para detectar la presencia de un objeto por lo que estos pueden detectar todo tipo de elementos. Además, se hace uso de una cámara de un dispositivo celular para la detección del color por su practicidad y porque podría ser una alternativa de bajo costo y que todas las personas pueden usar.

- *Sistema de Visión por Computadora:*

Se implementa un sistema de visión por computadora que procesará las imágenes capturadas por la cámara. El reconocimiento de los colores se hace por medio de inteligencia artificial usando OpenCV que es una librería de código abierto, diseñada para la visión por computadora y el procesamiento de imágenes y videos. Es una herramienta muy potente utilizada en una amplia gama de aplicaciones como el seguimiento de objetos, la detección de rostros, reconocimiento de patrones, segmentación de imágenes, entre otros. Esta programación se hizo en una interfaz de Python.

- *Plataforma de Control y Gestión:*

La clasificadora estará conectada a una plataforma central de control y gestión que supervisará y coordinará todas las operaciones. Este control se llevará a cabo por medio de un PLC o controladores lógicos programables los cuales están diseñados para supervisar y regular los procesos de

fabricación y las máquinas industriales mediante tecnología de automatización; además, pueden funcionar como sistemas autónomos para optimizar los procesos de manera inteligente e independiente.

Su principal objetivo consiste en la automatización de tareas a través de la interpretación de diversas entradas, que en este contexto se traducen en señales provenientes de sensores, interruptores y botones. Este sistema ejecuta acciones específicas en respuesta a estas entradas, con el fin de controlar dispositivos de salida, tales como motores y válvulas.

- *Actuadores y Sistema de Manipulación:*

Para garantizar una clasificación efectiva de las bolsas de basura, se integrarán cilindros doble efecto controlados por válvulas monoestables. Estos actuadores permitirán la selección y separación física de los desechos clasificados, direccionándolos hacia contenedores específicos para su posterior gestión o reciclaje.

- *Interfaz de Usuario Intuitiva:*

Se diseñará una interfaz de usuario intuitiva para facilitar la interacción con el sistema. Esta interfaz permitirá a los operadores controlar sin necesidad de acceder a los controladores físicos facilitando la gestión a distancia; y al mismo tiempo se busca llevar parámetros de control como lo es un contador de los elementos separados. La interfaz se hace por medio de HMI, el cual se refiere a la tecnología utilizada para la interacción entre humanos y máquinas; estos sistemas permiten a los operadores controlar, supervisar y gestionar equipos, maquinaria o procesos mediante interfaces gráficas intuitivas.

IV. DESARROLLO DEL PROYECTO

A continuación, se detallan minuciosamente los pasos considerados en el desarrollo del proyecto, abarcando desde la elección de los materiales hasta la elaboración de los códigos necesarios.

A. *Materiales Utilizados*

Para la ejecución de este proyecto, se han empleado diversos componentes, cada uno desempeñando un papel fundamental en su funcionamiento. A continuación, se detallan los elementos utilizados:

- ✓ Madera MDF para el diseño de la estructura.
- ✓ 4 cilindros de doble efecto.
- ✓ 4 electroválvulas monoestables.
- ✓ 4 sensores capacitivos para la detección de objetos.
- ✓ Botón start y stop.

- ✓ Motorreductor de 12 V.
- ✓ Banda transportadora.
- ✓ Sistema de transmisión por correa.
- ✓ Cámara de celular para la detección del color.
- ✓ Relé de 24 V para el motor.
- ✓ Placa de Arduino Uno.
- ✓ 3 relés del Arduino.
- ✓ PLC Siemens S7-1200.
- ✓ Cubos de colores.

B. *Estructura y Diseño*

La concepción del diseño estructural se centró en facilitar el ensamblaje, por lo que se optó por utilizar láminas de madera prensada tipo MDF con un espesor de 4,5 mm. Esto permitió implementar uniones de tipo solapado y entrelazado, asegurando un montaje más sencillo y una fijación precisa entre los diversos componentes, como se ilustra en la siguiente figura.

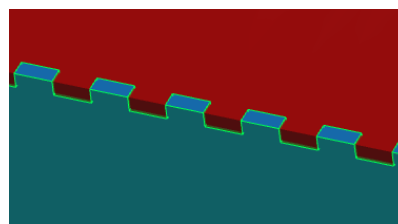


Figura 3. Unión estructural del montaje.

Para el corte del material se hizo uso de la cortadora laser que se encuentra en la universidad; para ello se hizo el despiece en un plano y luego se exportó en formato DXF.

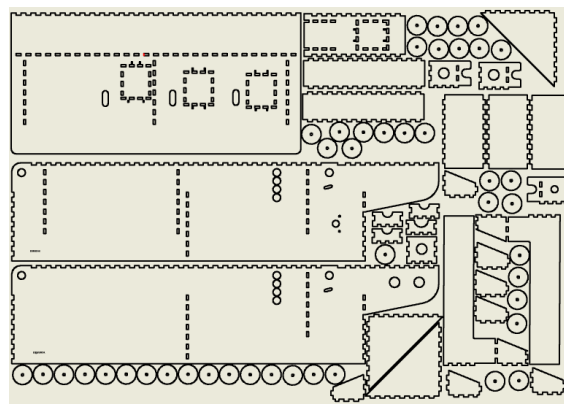


Figura 4. Plano de corte del montaje.

La disposición de la estructura está pensada para que la misma máquina esté soportada por un bloque o cajón central, el cual es la columna vertebral de toda la maquina y

contiene en su interior el motorreductor para impulsar la banda transportadora. El mecanismo de banda transportadora está fabricado en una lámina de látex natural. Asimismo, se incorporan botones físicos de inicio y paro, junto con un mecanismo de alimentación autónomo. Como resultado de esta configuración, se diseña la tolva de alimentación en la entrada para los elementos que se van a separar. En este caso específico, se utilizan cubos de madera de 40 mm x 40 mm x 40 mm, los cuales están pintados con diferentes colores para simular los distintos tipos de residuos que se deben separar.

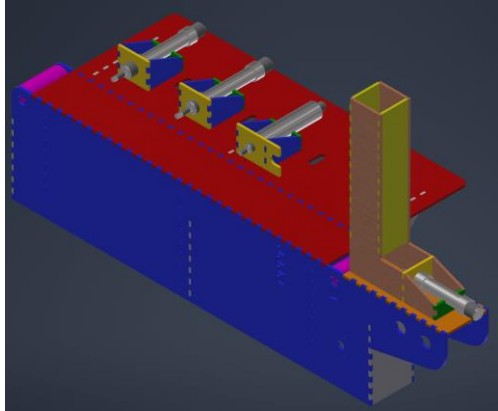


Figura 5. Diseño Preliminar del Montaje.

Se ha diseñado la superficie de trabajo de manera lateral, con soportes que albergan los actuadores responsables de llevar a cabo la separación por colores y la zona destinada a la inteligencia artificial.

Como se observa en la figura 5, en el extremo se la estructura se encuentra un dispensador, en el cual se ha posicionado que un actuador neumático que será el encargado de entregar los cubos a la banda transportadora de manera controlada, insertando un elemento a la vez. La cámara, mediante el uso de inteligencia artificial y código en PLC Siemens, determinará en cuál de los reservorios se ubicará cada cubo.

Finalmente, el sistema de banda transportadora se realizó por medio de anillos de madera cortados a laser unidos entre sí por medio de un eje de acero, el cual esta soportado por rodamientos confinados en la base de la máquina.

C. Circuito Neumático en Automation Studio

Para el circuito neumático se utilizan 4 electroválvulas monoestables, 4 cilindros doble efecto con sus reguladores de caudal y la alimentación de aire por medio de mangueras alimentadas por un compresor.

Lo anterior mencionado se presenta en la figura 6, donde se observan los cilindros designados para cada función.

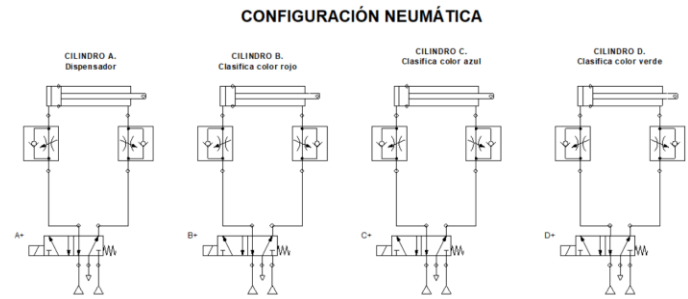


Figura 6. Circuito Neumático en Automation Studio.

D. Unión de Python y Arduino para la detección de objetos mediante Inteligencia Artificial

La detección de colores se realiza utilizando el lenguaje de programación Python. En este contexto, se aprovecha la funcionalidad proporcionada por la librería OpenCV, la cual permite definir un rango específico para identificar colores.

El código asociado se conecta con la cámara del celular a través de una conexión Wi-Fi, estableciendo un punto o píxel de detección en la mitad de la imagen capturada para mayor precisión en la identificación. Además, para facilitar la identificación, se incluye un texto que indica el color detectado, como se ilustra en la figura 7.

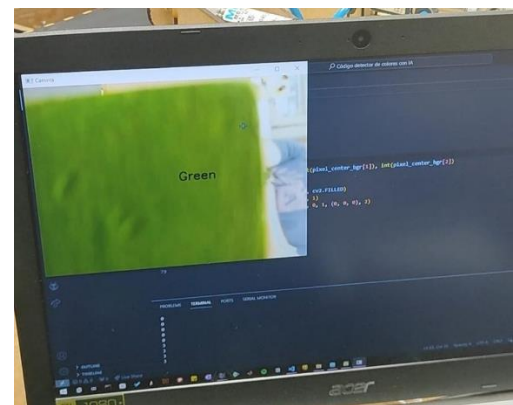


Figura 7. Detección de colores del código de Python.

Ante los desafíos experimentados en la conexión entre Python y el PLC, se opta por implementar una solución alternativa que implica el uso de una placa de Arduino. Esta placa actúa como intermediario, recibiendo la salida generada por el código en Python y transmitiéndola como entrada al PLC. Es relevante señalar que esta transición se

realiza mediante tres relés, asignando uno para cada salida o color. Esta elección se fundamenta en la diferencia de voltajes entre las entradas del PLC, que operan a 24 V, y la salida del Arduino, que trabaja a 5 V.

A continuación, se presentan el código de Python y Arduino utilizados, respectivamente.

```
import cv2
import numpy as np
import serial
import time

# Connection with Arduino
ser = serial.Serial('COM12', 9600, timeout=1)
time.sleep(2)

# To grab webcam feed:
webcam = cv2.VideoCapture(1)

# To read and display the camera/video:
while True:
    read_successful, img = webcam.read()
    hsv_img = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)

    # Let's take the width and the height of the camera displayed
    height, width, read_successful = img.shape

    # Let's define the point's coordinates [x, y]:
    cx = int(width / 2)
    cy = int(height / 2)

    # Pick pixel value
    pixel_center = hsv_img[cy, cx]

    # We only take the hue value because is the one that define the color:
    hue_value = pixel_center[0]

    #~ WE DEFINE THE COLOR:
    color = "Undefined"
    number = 0
    #^ RED:
    if hue_value < 5:
        color = 'Red'
        number = 2
        ser.write(b'R')
    elif hue_value < 21:
        color = "Orange"
        ser.write(b'D')
    elif hue_value < 33:
        color = "Yellow"
        ser.write(b'D')
    #^ GREEN:
    elif hue_value < 89:
        color = "Green"
        number = 3
        ser.write(b'G')
    #^ BLUE:
    elif hue_value < 131:
        color = "Blue"
        number = 4
        ser.write(b'B')
    elif hue_value < 172:
        color = "Violet"
        ser.write(b'D')
    else:
        color = 'Red'

    pixel_center_bgr = img[cy, cx]
    b, g, r = int(pixel_center_bgr[0]), int(pixel_center_bgr[1]), int(pixel_center_bgr[2])

    #print(pixel_center)
    cv2.circle(img, (cx, cy), 5, (b, g, r), cv2.FILLED)
    cv2.circle(img, (cx, cy), 5, (b, g, r), 1)
    cv2.putText(img, color, (cx+10, cy+10), 0, 1, (0, 0, 0), 2)
    print(number)

    cv2.imshow('Camera', img)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
    webcam.release()
    ser.close()
```

Figura 8. Código de Python.

```
int RED = 2;
int GREEN = 3;
int BLUE = 4;
int option;

void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600);
    pinMode(RED, OUTPUT);
    pinMode(GREEN, OUTPUT);
    pinMode(BLUE, OUTPUT);
    option='D';
}

void loop() {
    // put your main code here, to run repeatedly:
    if (Serial.available() > 0){
        option = Serial.read();
        Serial.println(option);
        if (option == 'R'){
            digitalWrite(RED, HIGH);
            digitalWrite(GREEN, LOW);
            digitalWrite(BLUE, LOW);
        }

        if (option == 'G'){
            digitalWrite(RED, LOW);
            digitalWrite(GREEN, HIGH);
            digitalWrite(BLUE, LOW);
        }

        if (option == 'B'){
            digitalWrite(RED, LOW);
            digitalWrite(GREEN, LOW);
            digitalWrite(BLUE, HIGH);
        }

        if (option == 'D'){
            digitalWrite(RED, LOW);
            digitalWrite(GREEN, LOW);
            digitalWrite(BLUE, LOW);
        }
    }
}
```

Figura 9. Código Arduino.

E. Código y Desarrollo del PLC.

Para la programación del PLC, se emplea una metodología que simplifica el proceso, como es el caso de las redes de Petri. Estas redes implican la definición de los estados de la máquina, los cuales están interconectados a través de transiciones. En estas transiciones, se incluyen todos los accionamientos o entradas necesarios para llevar a cabo las operaciones de la máquina, ejemplificados por los actuadores.

En la figura 10 se observa la red de Petri que se planteó para el proyecto integrador. En ella, a cada estado de la máquina se le da una marca denotada con la letra M, cuyo direccionamiento se da en bíte de 0.0 a 0.7 para completar un byte. Cada transición se denota con la letra T; en estas, se ponen todas las entradas que luego van al PLC. En una

transición pueden ir más de una entrada, pero es importante aclarar que para pasar de un estado a otro siempre se debe tener una transición.

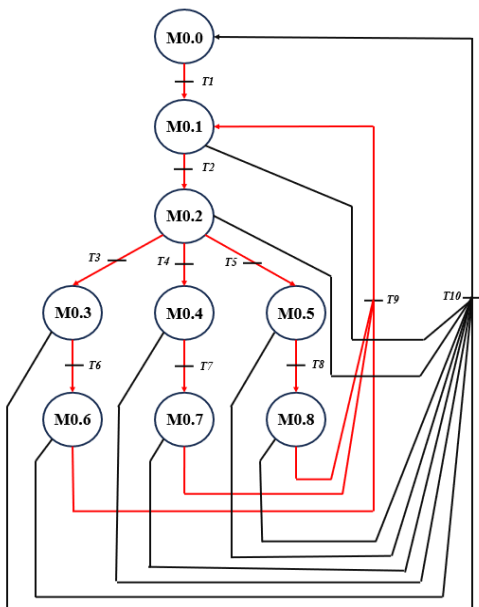


Figura 10. Red de Petri.

En la tabla 1 se presentan las entradas y direccionamientos que se utilizaron del PLC S7-1200. A cada color del código en Python se le dio una salida del Arduino que posteriormente se conectaba al PLC. Esas entradas están identificadas con los identificadores de la señal de color.

Transición	Entradas	Direccionamiento
T1	Start	I0.0
T2	Sensor de posición inicial	I0.1
T3	Identificador de la señal del color rojo	I0.2
T4	Identificador de la señal del color azul	I0.3
T5	Identificador de la señal del color verde	I0.4
T6	Sensor de posicion color rojo	I0.5
T7	Sensor de posicion color azul	I0.6
T8	Sensor de posicion color verde	I0.7
T9	Temporizador	I1.0
T10	Stop	I1.5

Tabla 1. Entradas PLC.

Las salidas del PLC están asociadas a los actuadores, siendo en este caso electroválvulas monoestables encargadas de controlar los cilindros de doble efecto y un motorreductor de 12 VAC como se presenta en la tabla 2.

Estados	Salidas	Direccionamiento
M0.0	Ninguna salida	-
M0.1	Piston dispensador, Motor	Q0.0, Q0.4
M0.2	Ninguna salida	-
M0.3	Motor	Q0.4
M0.4	Motor	Q0.4
M0.5	Motor	Q0.4
M0.6	Piston color rojo	Q0.1
M0.7	Piston color azul	Q0.2
M0.8	Piston color verde	Q0.3

Tabla 2. Salidas PLC.

Finalmente, la configuración de todas las entradas y salidas se realiza a través del software TIA Portal, tal como se ilustra en las figuras 10, 11 y 12.

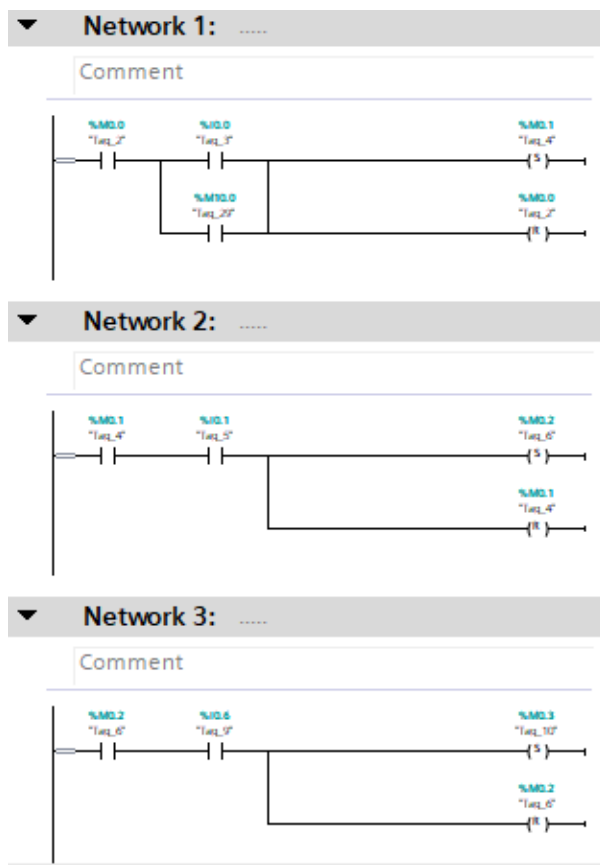


Figura 10. Ladder de los sensores

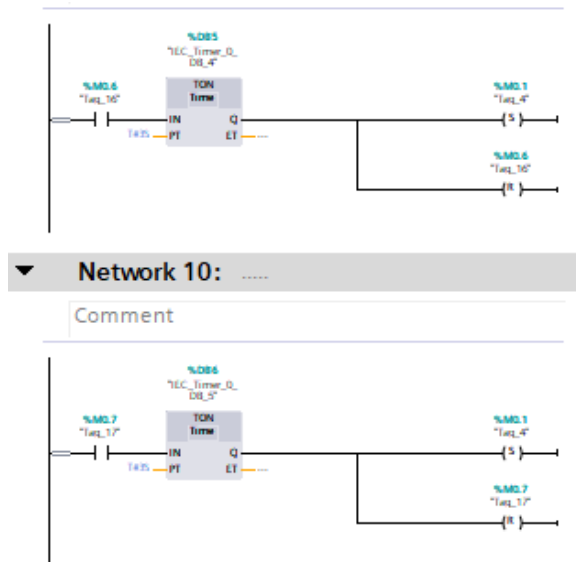


Figura 11. Ladder del control de las válvulas.

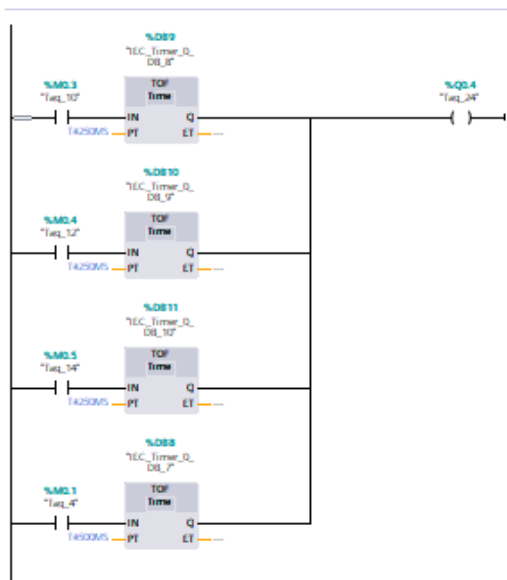


Figura 12. Ladder del control del motor.

F. Montaje y Prototipo.

El ensamblaje final se visualiza en la figura 13, la cual presenta la estructura completa con sus respectivos actuadores, la dispensadora ubicada a la derecha, la banda transportadora de látex, los botones de entrada "start" y "stop", el PLC con sus conexiones y HDMI, así como la cámara del celular encargada de captar los colores de los bloques.

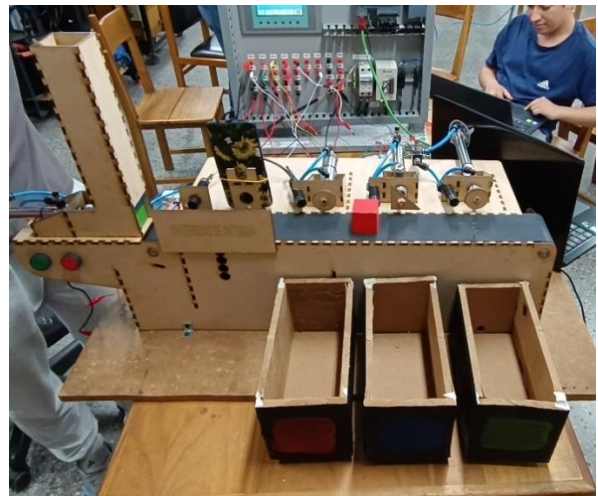


Figura 13. Montaje Final del Prototipo.

Se realizan las pruebas de funcionamiento como se observa en el vídeo del Proyecto integrador.

V. CONCLUSIONES

- Comunicar Python con el Software de Siemens TIA Portal conlleva el uso de lenguajes y códigos más robustos, siendo el software de Arduino una alternativa eficiente para comunicar Python con las entradas del PLC por medio de relés y una tarjeta Arduino.
- El uso de una cámara convencional acompañada de un código de Python resulta una herramienta versátil para diferentes usos en la automatización de procesos, tales como reconocimiento de imagen, objetos, colores y detección de objetos.
- La integración exitosa del Arduino con el PLC resultó ser una excelente opción para implementar inteligencia artificial a través de Python.

VI. REFERENCIAS

- [1] *Simple Color recognition with Opencv and Python* - https://www.youtube.com/watch?v=t71sQ6WY7L4&ab_channel=Pysource
- [2] *Comunicación PYTHON - ARDUINO con Pyserial / Mediapipe - OpenCV - YouTube.* (n.d.). Retrieved https://www.youtube.com/watch?v=qWESWDKYWFU&ab_channel=OMES

VII. ANEXOS

Video el Proyecto Integrador: [Proyecto Integrador Potencia Fluida: Clasificadora de Basura por Colores](#)