

INSTITUTO TECNOLÓGICO SUPERIOR

VIDA NUEVA



CARRERA:

TECNOLOGÍA EN ELECTROMECAÁNICA

TEMA DEL PROYECTO DE TITULACIÓN:

CONSTRUCCIÓN DE UN SISTEMA DE SERIGRAFÍA PARA EL ESTAMPADO DE
SELLOS UTILIZANDO ELECTROVÁLVULAS

AUTOR:

CHICAIZA ORTEGA OSCAR GERARDO

TUTOR:

ING. RUIZ GUANGAJE CARLOS RODRIGO

FECHA:

ENERO 2019

QUITO – ECUADOR

CESIÓN DE DERECHOS DE AUTOR

Yo, **CHICAIZA ORTEGA OSCAR GERARDO** portador/a de la cedula de ciudadanía **180507068 - 5**, facultado/a de la carrera **TECNOLOGÍA EN ELECTROMECHANICA**, autor/a de esta obra certifico y proveo al Instituto Tecnológico Superior Vida Nueva, usar plenamente el contenido plasmado en este escrito con el tema **“CONSTRUCCIÓN DE UN SISTEMA DE SERIGRAFÍA PARA EL ESTAMPADO DE SELLOS UTILIZANDO ELECTROVÁLVULAS”**, con el objeto de aportar y promover la lectura e investigación, autorizando la publicación de mi trabajo de titulación en la colección digital del repositorio institucional bajo la licencia de Creative Commons: Atribución-NoComercial-SinDerivadas.

En la ciudad de Quito, al mes de Enero de 2019.

CERTIFICACIÓN

En mi calidad de Tutor del Proyecto: CONSTRUCCIÓN DE UN SISTEMA DE SERIGRAFÍA PARA EL ESTAMPADO DE SELLOS UTILIZANDO ELECTROVÁLVULAS, presentado por el estudiante CHICAIZA ORTEGA OSCAR GERARDO, para optar por el título de Tecnólogo en Electromecánica, certifico que dicho proyecto ha sido revisado en todas sus partes y considero que reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del tribunal que se designe.

Tutor: Ing. Ruiz Guangaje Carlos Rodrigo

APROBACIÓN DEL TRIBUNAL EXAMINADOR

Los miembros del tribunal examinador aprueban el informe del Proyecto de Aplicación Práctica, con el tema:

CONSTRUCCIÓN DE UN SISTEMA DE SERIGRAFÍA PARA EL ESTAMPADO DE SELLOS UTILIZANDO ELECTROVÁLVULAS

Del Sr. estudiante: **CHICAIZA ORTEGA OSCAR GERARDO**

De la Carrera, Tecnología en Electromecánica.

Para constancia firman:

.....

.....

.....

DECLARACIÓN DE AUTENTICIDAD

Yo, CHICAIZA ORTEGA OSCAR GERARDO, estudiante del Instituto Tecnológico Superior Vida Nueva, declaro que he realizado este trabajo de titulación tomando en consideración citas bibliográficas que se nombran en este texto.

El Instituto Tecnológico Superior Vida Nueva puede utilizar este trabajo de titulación como una ayuda bibliográfica.

Chicaiza Ortega Oscar Gerardo

C.I 180507068-5

AGRADECIMIENTO

El agradecimiento de mi trabajo práctico de titulación en primer lugar es a Dios quien me ha guiado y me ha dado fortaleza de seguir adelante y cumplir mi sueño tan anhelado.

Agradezco también al Instituto Tecnológico Superior Vida Nueva por abrirme sus puertas y ser un profesional, gracias a cada uno de los docentes con su profesionalismo y rectitud nos ha compartido sus conocimientos para poder desarrollar mi proyecto de titulación y culminar mis estudios con éxito.

Igualmente quiero agradecer a mi tutor el Ingeniero Carlos Ruiz por brindarme su valiosa colaboración y orientación para el desarrollo este trabajo.

Chicaiza Ortega Oscar Gerardo

DEDICATORIA

Este trabajo práctico esta dedicado a Dios y a mis padres Carlos y Yolanda, por haberme dado la vida y permitirme llegar hasta este momento importante de mi carrera profesional. A mi madre por su esfuerzo, apoyo, dedicación y confianza en mis momentos difíciles. A mi padre por sus consejos ha sabido guiarme para llegar a ser un profesional, todo lo que hoy soy es gracias a ellos, a cada uno de los docentes por su apoyo y compartirme sus conocimientos para culminar mi carrera profesional.

A mi esposa Jessica por su amor, compañía, paciencia y apoyo incondicional a pesar de los tropiezos ha estado conmigo siempre y a mi mayor tesoro mi hijo Dilan por ser fuente de motivación e inspiración para poderme superar cada día más y ser un gran ejemplo para él.

A mis hermanas y toda mi familia que me apoyaron para cumplir mi meta siempre dándome aliento para nunca rendirme y llegar a ser una persona perseverante y cumpla mis ideales.

Oscar Gerardo Chicaiza Ortega

ÍNDICE GENERAL

3.OBJETIVOS	5
3.1. Objetivo General.....	5
3.2. Objetivos Específicos.....	5
4.DESARROLLO O CUERPO PRINCIPAL	5
4.1. Marco Teórico – Conceptual.....	5
4.1.1.Serigrafía.....	6
4.1.2. Sistema Neumático	6
4.1.3. Cilindros Neumáticos	6
<i>4.1.3.1. Tipos de cilindros.....</i>	<i>7</i>
4.1.4 Electroválvulas	8
<i>4.1.4.1. Tipos de electrovalvulas</i>	<i>9</i>
4.1.5. Zelio PLC.....	10
<i>4.1.5.1. Beneficios del ZELIO PLC.....</i>	<i>10</i>
<i>4.1.5.2. Programación.....</i>	<i>11</i>
<i>4.1.5.3. Partes Principales del ZELIO PLC.....</i>	<i>11</i>
4.1.6. Arduino Megas.....	12
<i>4.1.6.1. Características del Arduino Mega.....</i>	<i>13</i>
4.1.7. Sensor GY-31.....	14
4.1.8. Sensor de Obstáculos TCRT5000.....	16
<i>4.1.8.1. Especificaciones.....</i>	<i>17</i>
<i>4.1.8.2. Esquema Eléctrico.....</i>	<i>17</i>
4.1.9. Motor DC 12V.....	18
<i>4.1.9.1. Especificaciones.....</i>	<i>19</i>
4.1.10. Módulo de relés de 8 canales.....	20
<i>4.1.10.1. Especificaciones.....</i>	<i>21</i>

4.1.10.2. Características.....	21
4.1.11. Fuente de alimentación.....	22
4.1.11.1. Etapas de las fuentes de alimentación.....	22
4.2. Procedimiento – Metodología.....	23
4.2.1. Diseño	23
4.2.1.1. Sistema Eléctrico.....	23
4.2.1.1.1 Panel de Control y Energización	23
4.2.1.1.2 Control Sensores y Motores	24
4.2.1.1.3. Parte interna de la caja de control	24
4.2.1.2. Sistema Neumático.....	25
4.2.1.2.1. Electroválvulas.....	25
4.2.1.2.2. Cilindros doble efecto.....	26
4.2.1.2.3. Unión Neumática tipo Y.....	26
4.2.1.2.4. Compresor.....	27
4.2.1.2.5. Manguera Neumática	27
4.2.1.3. Sistema Mecánico	27
4.2.1.3.1 Estructura	28
4.2.1.3.2 Ejes Cilíndricos	28
4.2.1.3.3 Chumaceras	28
4.2.1.3.4 Banda transportadora	29
4.2.1.3.5 Acople para motor	30
4.2.1.3.6 Pernos Prisioneros	30
4.2.2. Solución del problema.....	30
4.2.3. Proceso de control.....	31
4.3. Construcción.....	32
4.3.1. Estructura.....	32

4.3.2. Sistema de control.....	32
4.3.3. Programación en el software de Arduino.....	33
4.3.3.1. Programación del sensor de color GY-31.....	33
4.3.3.2. Programación del sensor de obstáculos TCRT5000.....	35
4.3.4. Programación en el Software Zelio Soft.....	36
4.3.4.1. Programación del Zelio PLC.....	36
4.4. Implementación.....	37
5. CONCLUSIONES.....	39
6. RECOMENDACIONES.....	40
7. FUENTES.....	41
7.1 Bibliografía.....	41

ÍNDICE DE FIGURAS

FIGURA N° 1 Cilindro Neumático	7
FIGURA N° 2 Electroválvula Monoestable	8
FIGURA N° 3 Zelio PLC.....	10
FIGURA N° 4 Partes de Zelio PLC.....	12
FIGURA N° 5 Tarjeta Arduino Mega	13
FIGURA N° 6 Sensor GY-31 RGB	14
FIGURA N° 7 Declaración e inicialización de variables y constantes.	15
FIGURA N° 8 Declaración de los pines de salida	15
FIGURA N° 9 Sensor de obstáculos.....	17
FIGURA N° 10 Esquema eléctrico del sensor de obstáculos	19
FIGURA N° 11 Conexión de Arduino con el sensor	19
FIGURA N° 12 Motor DC.....	20
FIGURA N° 13 Sentido de Giro.....	22
FIGURA N° 14 Módulo de reles 8 canales.....	23
FIGURA N° 15 Fuente de alimentación.....	24
FIGURA N° 16 Sistema eléctrico	26
FIGURA N° 17 Panel de control	26
FIGURA N° 18 Panel de control interna.....	27
FIGURA N° 19 Electroválvulas monoestables.....	28
FIGURA N° 20 Cilindro Doble Efecto.....	29
FIGURA N° 21 Unión Neumática	29
FIGURA N° 22 Compresor.....	30
FIGURA N° 23 Manguera Neumática.....	30
FIGURA N° 24 Tubo cuadrado de acero.....	31
FIGURA N° 25 Ejes cilíndricos.....	31
FIGURA N° 26 Chumaceras.....	32
FIGURA N° 27 Banda transportadora.....	32
FIGURA N° 28 Acople de duralon	33
FIGURA N° 29 Prisioneros	33
FIGURA N° 30 Estructura terminada.....	35
FIGURA N° 31 Sistema de control	36
FIGURA N° 32 Protecciones del panel de control	36

FIGURA N° 33 Programación sensor GY-31	37
FIGURA N° 34 Void setup y loop de sensor GY-31.....	38
FIGURA N° 35 Programación sensor de obstaculos.....	39
FIGURA N° 36 Programación del Zelio PLC.....	39
FIGURA N° 37 Lineas de programación PLC.....	40

ÍNDICE DE ANEXOS

8.1. Pintada de la estructura	42
8.2. Implementación de los elementos de control	42
8.3. Sensor de obstáculos	42
8.4. Implementación de los elementos en la caja de control	43
8.5. Cableado de la caja de control	43
8.6. Implementación del motor DC con la banda	43
8.7. Unión de cables del sensor con el cable UTP	44
8.8. Tarjeta para controlar los PWM del motor DC	44
8.9. Prueba de programación del sensor de obstáculos.....	44
8.10. Prueba de programación de unión de sensores	45

INTRODUCCIÓN

Un sistema automatizado de control es un conjunto de componentes conectados o que se relacionan entre sí, los cuales regulan o actúan por sí mismos, es decir, sin la intervención humana, corrigiendo los posibles errores que se presenten desarrollando un sistema continuo.

En el transcurso del tiempo las grandes empresas han tenido un relevante incremento de producción, lo cual ha llevado a adquirir nuevas tecnologías permitiendo la automatización de ciertos procesos en sus plantas, tanto físicos como manuales, mejorando las actividades o tareas que cumpla cada uno de los procesos.

Las empresas actualmente que cuentan con un sistema de sellado tiene el problema de estampar sus sellos manualmente, lo que no es favorable para las empresas por lo que existen pérdidas de tiempo como de material al momento de sellar a la velocidad en la que se desplaza la banda, para lo cual se debía contar un grupo excesivo de operarios para estar pendientes de los objetos que no hayan sido sellados; todo esto conlleva a un incremento de costo de producción.

El proceso que es manejado por los operarios es ineficiente ya que están sometidos a trabajar a velocidades excesivas por la que deben acoplarse a la rapidez con la que trabaje la banda transportadora al momento de sellar lo cual puede llegar a dañar la calidad del producto y aumentar la cantidad de pérdidas.

El presente proyecto se enfoca principalmente en resolver el problema de sellado de los objetos según sus colores, el cual facilita el trabajo para los operadores mediante sistemas inteligentes, de esta manera se evitará que el operador o los operadores estén verificando

constantemente cada uno de los productos, por lo tanto, solo bastará que el operador presione el botón de inicio para que el sistema comience a operar automáticamente.

El sistema propuesto será aplicable en diversas industrias, ya que se mejorará la productividad de toda la empresa al lograr producir más objetos en menos tiempo, destacando la calidad de la misma y así se reducirá los costos de producción.

Este sistema de sellado también se podrá utilizar para el aprendizaje de programación e implementación del PLC, ya que la utilización del mismo es fundamental en aquellas aplicaciones en donde es necesario un proceso de control, el sistema de sellado automatizado tendrá una infinidad de beneficios industriales los cuales abarcan a una variedad de empresas que contiene este tipo de proceso de sellado de objetos, reduciendo tiempo, costos y mejorando la calidad del producto.

ANTECEDENTES

Según, Moreta, G. (2012), la automatización se viene desarrollando a partir del siglo XX, en donde se utilizaban mecanismos simples para automatizar tareas sencillas de manufactura en el área de la mecánica e ingeniería. El concepto llegó a ser realmente práctico con la adición y evolución de las computadoras digitales, cuya flexibilidad permitió manejar cualquier clase de tarea. Las computadoras digitales con la combinación requerida de velocidad y el desarrollo de la programación han permitido ser aplicadas en las industrias textiles como estampado de objetos que aparecieron en los años 1960.

Así también Laguna, L. (2009), expresa que antiguamente el proceso de serigrafía se realizaba en las hojas de plátano agujereadas para la respectiva distribución de la tinta, las primeras impresiones se hacen en el año 3000 a.C., desde ahí se origina el proceso de serigrafía, por lo que se dice que este sistema de estampado es milenario.

En el Ecuador la serigrafía se ha venido actualizando a partir del año 2010, lo que representa que aún no se cuenta con la tecnología suficiente como en otros países, a nivel mundial.

El PLC (Controladores Lógicos Programables) se remonta a finales de la década de 1960, cuando la industrias buscaban nuevas tecnologías electrónicas para la automatización de máquinas que permitan facilitar el trabajo, se ha desarrollado en varios campos, sin tomar en cuenta las industrias que se encargan del estampado de objetos, para lo cual se sugiere del software Zelio Soft que facilitará realizar la programación de manera más rápida y sencilla.

Actualmente la automatización del estampado en industrias grandes ha permitido un trabajo eficiente y un ahorro considerable para la empresa, ya que las máquinas son controladas por un ordenador y los controladores lógicos programables. Gracias a los PLC's es posible ahorrar tiempo en la elaboración de proyectos, por lo que se puede realizar cualquier tipo de

modificación en lo que se refiere a la programación sin tener ningún tipo de costo. Estos son de un tamaño pequeño lo cual permite un ahorro económico a lo que se refiere a mano de obra y de tiempo también la posibilidad de controlar más de una máquina con el mismo equipo.

RESUMEN

Los sistemas automatizados están conformados por un conjunto de componentes entre sensores, sistema de control y actuadores, cuya finalidad es relacionarse entre sí para ejecutar una función específica, todo esto sin la intervención humana lo cual permite que en las empresas sus procesos de producción sean más económicos, el producto final sea de mejor calidad e incluso mejorar el tiempo estimado para cierta cantidad de producción, en la actualidad gran cantidad de las empresas que se encargan de realizar el estampado y sellado de materiales o productos desarrollan esta actividad de forma manual lo que se traduce directamente en baja productividad o producción de mala calidad, es por esto que el presente proyecto tiene como finalidad construir un sistema de estampado o serigrafía para la ubicación de sellos de acuerdo al color de material, este sistema se encuentra compuesto de un Arduino Uno y un Relé programable Zelio que son los encargados de analizar las señales enviadas por los sensores para ejecutar las acciones sobre los actuadores, inicialmente por medio de un cilindro neumático el objeto es desplazado hacia la banda transportadora, una vez que el objeto se encuentra en el punto de desplazamiento, un motor de corriente continua se enciende para controlar el movimiento de la banda, hasta que el objeto llegue al punto de clasificación se encuentran ubicados sensores de colores, los cuales permiten identificar el tipo de elemento, automáticamente Arduino decide que tipo de sello debe ser ubicado, es importante considerar que Arduino procesa la señal del sensor de color para posteriormente enviar una señal al Zelio, el cual se encargará de controlar la entrada o salida de cilindros neumáticos que contienen los tipos de sellos.

PALABRAS CLAVE:

Arduino Uno, ZELIO, Cilindros Neumáticos, Sensores de color, Banda transportadora.

ABSTRACT

3. OBJETIVOS

3.1. Objetivo General

- Diseñar un sistema automatizado para el estampado de objetos según los colores, utilizando controladores lógicos programables en el Instituto Tecnológico Superior Vida Nueva.

3.2. Objetivos específicos

- Diseñar la estructura del proyecto, utilizando el software AutoCAD para dimensionar los elementos a utilizar.
- Elaborar los soportes mecánicos, para el movimiento de la banda y transportar distintos objetos mediante un motor.
- Implementar el gabinete en la estructura, para realizar las conexiones del PLC y las electroválvulas en el tablero de control.
- Realizar pruebas de funcionamiento, mediante el PLC para verificar el funcionamiento de los actuadores en la selección de colores para el estampado de los sellos de la institución.

4. DESARROLLO O CUERPO PRINCIPAL

4.1.Marco Teórico – Conceptual

4.1.1. Serigrafía

La palabra serigrafía del griego Serikós (seda) y Graphé (escribir), dibujar; se refiere al sistema de impresión, derivado de la antigua técnica de estarcido, que utiliza como matriz un marco con una malla abierta en ciertas zonas, que es la imagen a imprimir, y cerradas en otras.

Según Moreta, G. (2012), la serigrafía es una técnica de impresión empleada en el método de reproducción de documentos e imágenes sobre cualquier material, y consiste en transferir una tinta a través de una malla tensada en un marco, el paso de la tinta se bloquea en las áreas donde no habrá imagen mediante una emulsión o barniz, quedando libre la zona donde pasará la tinta.

4.1.2. Sistema Neumático

Según Valencia, J. (2010), un sistema neumático es todo aquel que funciona en base a aire comprimido, ósea aire a presión superior a una atmósfera, en los sistemas neumáticos, el aire comprimido se produce en un elemento llamado compresor, que es una bomba de aire comprimido accionada normalmente por un motor eléctrico, este aire se almacena en un depósito denominado receptor, desde éste, el aire es conducido a través de válvulas a los cilindros, que son los componentes encargados de realizar el trabajo, aquellas que los producen se llaman compresores.

Anteriormente se usaban pistones para comprimir el aire, ahora los compresores modernos utilizan dos tornillos giratorios para comprimirlos en un solo paso, obviamente

estas máquinas utilizaban el aire como su materia prima, aunque este puede ser tratado para una mayor pureza y mejor trabajo.

4.1.3 Cilindros Neumáticos

Los cilindros neumáticos son, por regla general, los elementos que realizan el trabajo tomando en cuenta que su función es la de transformar la energía neumática en trabajo mecánico de movimiento rectilíneo la cual consta de carrera de avance y carrera de retroceso.

Guillen, A. (2010), señala que generalmente el cilindro neumático está constituido por un tubo circular cerrado en los extremos mediante dos tapas, entre las cuales se desliza un émbolo que separa dos cámaras. Al émbolo va unido a un vástago que saliendo a través de ambas tapas, permite utilizar la fuerza desarrollada por el cilindro en virtud de la presión del fluido al actuar sobre las superficies del émbolo, como se muestra en la figura 1.



Figura 1. Cilindros neumáticos. Por Guillen A. (2010)

4.1.3.1. Tipos de Cilindros

Para Guillen, A. (2010), los cilindros neumáticos variarán de aspecto, tamaño y la función, no obstante hay también numeroso tipos de cilindros neumáticos, muchos de los cuales se diseñan para satisfacer funciones específicas y especializadas.

4.1.3.1.1 Cilindros de acción simple (SACO)

Utilizan la fuerza impartida por el aire para moverse en una dirección (generalmente hacia fuera), y un resorte a la vuelta “a casa” a la posición.

4.1.3.1.2 Cilindros dobles (DAC)

Utilizan la fuerza del aire para moverse el cual se extienden y contraen movimientos. Tienen dos puertos para permitir el aire adentro, uno para el outstroke y uno para el instroke.

4.1.3.1.3 Cilindros rotatorios del aire

Actuadores que utilizan el aire para impartir un movimiento rotatorio.

4.1.3.1.4 Cilindros del aire de Rodless

Los actuadores que utilizan un acoplador mecánico o magnético para impartir la fuerza, típicamente a la tabla o al otro cuerpo que se mueve a lo largo de la longitud del cuerpo del cilindro, pero no extienden más allá de él.

4.1.4. Electroválvulas

Aceves, A. (2011), dice que la válvula electromecánica, diseñada para controlar el paso de un fluido por un conducto o tubería. Generalmente no tiene más que dos posiciones: abierto y cerrado, o todo y nada. Las electroválvulas se usan en multitud de aplicaciones para controlar el flujo de todo tipo de fluidos. No se debe confundir la electroválvula con válvulas motorizadas, en las que un motor acciona el mecanismo de la válvula, y permiten otras posiciones intermedias entre todo y nada. Una electroválvula tiene dos partes fundamentales: el solenoide y la válvula, como se muestra en la figura 2.



Figura 2. Electroválvula Monoestable. Por Aceves, A. (2011)

4.1.4.1. Tipos de electroválvulas

Docplayer (2018) menciona 2 tipos de electroválvulas los cuales son:

4.1.4.1.1 Electroválvulas monoestables

Son aquellas que tienen una posición de reposo estable, que es en la que permanecerá la válvula de forma indefinida si no actúa sobre ella el dispositivo de mando. El regreso a la posición de reposo suele realizarse con un muelle; así en el caso de electroválvula monoestable de dos posiciones. Serán normalmente abiertas NA cuando en su posición estable dejen pasar a su través el fluido a presión hacia los elementos de consumo. Serán normalmente cerradas NC cuando en su posición estable no dejen pasar el fluido o lo dejen pasar desde los elementos actuadores hacia el escape.

4.1.4.1.2 Electroválvulas biestables

Son aquellas que no tienen una única posición de reposo estable; es decir, que aunque se anule la señal que provocó la posición en la que se encuentra, la válvula seguirá en esa misma posición hasta que se active la señal correspondiente a una nueva posición. En el caso de que se activen dos señales prevalece la más antigua.

El dispositivo de mando y la forma en que se activa cada posición en la válvula se representa simbólicamente añadiendo en la parte lateral del cuadrado el símbolo del accionamiento correspondiente. En ocasiones, también a las válvulas biestables se les dice NA o NC. Cuando así ocurra, será NA cuando lo sea en la posición de la derecha, caso de tener dos posiciones o en la posición central, caso de tener tres posiciones. Análogamente para NC.

4.1.5. Zelio PLC

Según Prieto, P. (2007), Zelio es una gama de relés inteligentes o pico-PLC (Controlador Lógico Programable) autónomos que incluyen entradas digitales o analógicas para la conexión de sensores, finales de carrera, pulsadores, interruptores, etc. una serie de salidas para la conexión de contactores, lámparas, válvulas, variadores de velocidad, etc. y una pantalla para mostrar textos al usuario, los PLC's Zelio contiene temporizadores, contactores y diferentes funciones internas que gracias a estos se puede controlar sistemas automatizados sencillos.

Los relés programables Zelio Logic están diseñados para realizar pequeñas aplicaciones de automatismos. Se utilizan en actividades industriales y del sector terciario; como se muestra en la figura 3.



Figura 3. Zelio PLC. Por ZELIO (2007)

4.1.5.1. Beneficios del ZELIO PLC

ZELIO PLC cuenta con los siguientes beneficios:

- Automatismos de máquinas de acabado pequeñas, de confección, de ensamblaje o de embalaje.
- Automatismos descentralizados en los anexos de las máquinas grandes y medianas en los ámbitos textiles, del plástico, de la transformación de materiales, etc.
- Automatismos para máquinas agrícolas (irrigación, bombeo, invernaderos...).
- Por ser compacto y fácil de instalar, supone una solución competitiva frente a otras de lógica cableada o de tarjetas específicas (Schneider Electric, 2006).

4.1.5.2. Programación.

Para Prieto, P. (2007), su programación es sencilla, debido al carácter general de los lenguajes y cumple las exigencias del diseñador de automatismos y las expectativas del electricista, la programación se puede realizar:

- De forma independiente utilizando el teclado del módulo Zelio Logic (lenguaje de contactos).
- En el PC con el software “Zelio Soft 2”.
- En PC, la programación se puede efectuar bien en lenguaje de contactos (LADDER), bien en lenguaje de bloques de función.

4.1.5.3. Partes principales del ZELIO PLC

El Zelio Logic compacto poseen las siguientes partes que se describe a continuación:

- Dos patas de fijación retráctiles
- Dos bornes de alimentación.
- Bornes de conexión de las entradas.

- Un visualizador LCD retroiluminado de 4 líneas de 18 caracteres.
- Un emplazamiento para cartucho de memoria y conexión al PC o interface de comunicación por módem.
- Un teclado de 6 teclas para la programación y el parametraje.
- Bornes de conexión de las salidas.

En la figura 4 se muestran los elementos del panel frontal del módulo logic para distinguir cada una de ellas.

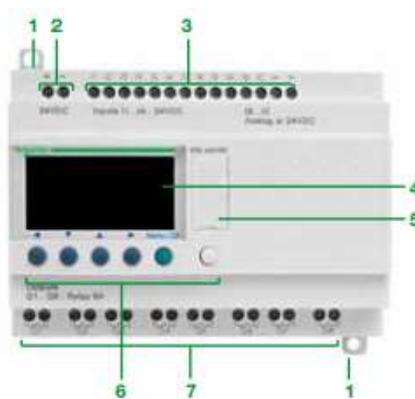


Figura 4. Partes de Zelio PLC. Por ZELIO (2007)

4.1.6. Arduino Mega

Es una placa de micro controlador basada en ATmega1280. Tiene 54 pines digitales de entrada / salida (de los cuales 14 se pueden usar como salidas PWM), 16 entradas analógicas, 4 UART (puertos serie de hardware), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP, y un botón de reinicio Arduino.

Por otro lado Hernández, L. (2018), propone que este microprocesador contiene todo lo necesario para soportar todo tipo de programaciones, para poder comenzar a programar se debe conectar a una computadora con un cable USB o con un adaptador de CA a CC o

batería El Mega es compatible con la mayoría de los escudos diseñados para el Arduino Duemilanove o Decimilla, como se muestra en la figura 5.



Figura 5. Tarjeta Arduino Mega. Por Hernández, L. (2018),

4.1.6.1. Características del Arduino Mega

El Arduino Mega tienen las siguientes características que se detalla a continuación:

- El microprocesador ATmega1280
- 128 Kbyte de memoria flash
- 8 Kbyte de memoria RAM
- 4 Kbyte de EEPROM
- 16 MHz
- 54 pines para entradas/salidas digitales (programables)
- 16 pines para entradas analógicas
- 16 pines para salidas PWM
- Completamente autónomo: Una vez programado no necesita estar conectado al PC
- Microcontrolador ATmega1280
- Voltaje de operación 5V
- Voltaje de entrada (recomendado) 7-12 V
- Voltaje de entrada (límite) 6-20 V
- Digital I/O pines 14 (con 6 salidas PWM)

- DC corriente I/O Pin 40 mA
- DC corriente 3.3V Pin 50 mA
- Memoria flash 128 KB (4 KB para el bootloader)
- SRAM 8 KB
- EEPROM 4 KB
- Velocidad de reloj 16 MHz

4.1.7. Sensor GY-31

Electrónica Caldas, M. (2016) explica que es un detector de color que incluye un chip sensor TAOS TCS3200 RGB y 4 Leds que ayuda a medir una gama ilimitada de colores visibles como pueden ser: rojo, verde, azul y blanco sin filtro convirtiendo el color a frecuencia en un circuito integrado CMOS. La salida es una onda cuadrada con una frecuencia directamente proporcional a la intensidad de luz. La frecuencia de salida que se obtiene puede escalarse por uno de tres valores preestablecidos a través de dos pines de entrada de control S2 y S3, como se muestra en la figura 6.

Las aplicaciones incluyen clasificación por color, sensor de luz ambiente, calibración y coincidencia de colores, entre otras.



Figura 6. Sensor GY-31 RGB. Por Caldas, M. (2017)

4.1.7.1. Pines de la unidad de control

Se utilizan para recibir señales que el sensor recibe y procesa. El pin LED se usa para encender y apagar el LED, es mejor mantener los leds encendidos en todo momento debido a la precisión del sensor en ausencia de oscilación. El pin OUT sirve para recibir todos los valores totales del sensor, el sensor lee el reflejo de la luz a través de rojo, verde, azul y sin filtro lo registra como datos, es decir, en el programa se convierte el valor RGB, como se muestra en la figura 7.

```
const int s0 = A3;
const int s1 = A4;
const int s2 = A1;
const int s3 = A2;
const int out = A0;
// values
int red = 0;
int green = 0;
int blue = 0;
```

Figura 7. Declaración e inicialización de variables y constantes. Por Hackster.io (2018)

La programación de la función principal son los pines s0, s1, s2, s3 y el modo 'OUT' está configurado en OUTPUT (salida de datos), como muestra la figura 8.

```
void setup()
{
  Serial.begin(9600);

  pinMode(s0, OUTPUT);
  pinMode(s1, OUTPUT);
  pinMode(s2, OUTPUT);
  pinMode(s3, OUTPUT);
  pinMode(out, INPUT);

  digitalWrite(s0, HIGH);
  digitalWrite(s1, HIGH);
}
```

Figura 8. Declaración de los pines de salida. Por Hackster.io (2018)

Los pines S0 y S1 son programados con la ayuda de la herramienta digital Write configurado en pulsos altos. Hay varias maneras de leer el reflejo de los colores en

diferentes porcentajes como: 2%, 20% y 100% de reflexión. Para poder ubicar el porcentaje correcto dependerá del entorno y las condiciones en que funciona el sensor como se ve observa en la tabla 1 y 2 que muestran opciones seleccionales de los pines del sensor.

Tabla 1.

Porcentajes de frecuencia de los colores.

S0	S1	OUTPUT FREQUENCY SCALING
L	L	Power down
L	H	2%
H	L	20%
H	H	100%

Nota: Tomada de Hackster.io (2018)

Tabla 2.

Asignación en los pines para los colores

S2	S3	OUTPUT FREQUENCY SCALING
L	L	Red
L	H	Blue
H	L	Clear
H	H	Green

Nota: tomada de Hackster.io (2018)

4.1.7.2. Características del sensor GY-31

El módulo sensor de color tiene 4 canales de detección de objetos por ello tiene las siguientes características:

- 3-5v Alimentación
- Resistencia a la interferencia de la luz
- LED blanco se puede controlar encendido, apagado.

- Puede detectar el color del objeto no luminoso
- Mejor detección de distancia de 1 cm.

4.1.8. Sensor de obstáculos TCRT5000

Es un dispositivo que detecta la presencia de un objeto mediante el reflejo que produce la luz. Es un sensor sencillo que dispone de un LED emisor de luz infrarroja y de un fotodiodo que recibe la luz reflejada por un posible obstáculo, como se observa en la figura 9.

Para obtener una lectura como valor digital y poder sincronizarle con el Arduino este sensor, en su placa contiene un comparador LM393, que convierte la señal del sensor en una digital para poder usarla en cualquier actuador.



Figura 9. Sensor de obstáculos. Por Llamas, L. (2001)

Llamas, L. (2001), afirma que este tipo de sensores actúan a distancias cortas, típicamente de 5 a 20mm. Además la cantidad de luz infrarroja recibida depende del color, material, forma y posición del obstáculo, por lo que no disponen de una precisión suficiente para proporcionar una estimación de la distancia al obstáculo.

Pese a esta limitación son ampliamente utilizados para la detección de obstáculos en pequeños vehículos o robots. Su bajo coste hace que sea frecuente ubicarlos en el perímetro, de forma que detectemos obstáculos en varias direcciones.

También son útiles en otro tipo de aplicaciones como, por ejemplo, detectar la presencia de un objeto en una determinada zona, comprobar una puerta si está abierta o cerrada, o si una máquina ha alcanzado un cierto punto en su desplazamiento. En lo que respecta a al esquema eléctrico se hace referencia que el montaje es sencillo puesto que se necesita alimentar el módulo a través de Vcc y GND conectándolos, respectivamente, a la salida de 5V y GND en Arduino, como se puede observar en la figura 10.

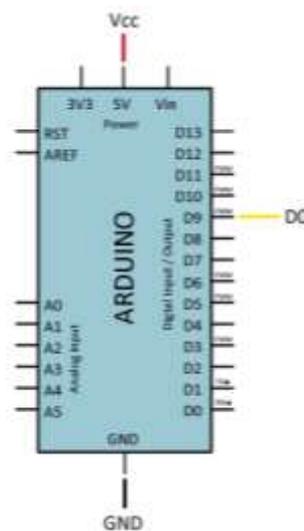


Figura 10. Esquema eléctrico del Sensor de obstáculos. Por Llamas, L. (2017)

Finalmente, se conecta la salida digital del sensor a una entrada digital para leer el estado del sensor, como se muestra en la figura 11.

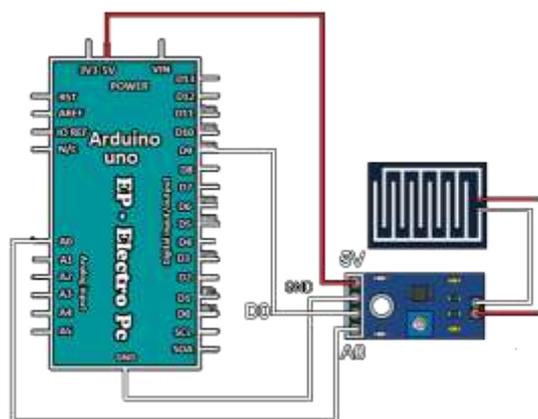


Figura 11. Conexión de Arduino con el Sensor de obstáculos. Por Llamas, L. (2017)

Opcionalmente, se calibra el paso de disparo cuando se acerque un objeto al detector de obstáculos para regular la salida digital con el potenciómetro del mismo.

4.1.8.1. Especificaciones

El Sensor de obstáculos TCRT5000 debe tener las siguientes descripciones como lo detallamos a continuación:

- Voltaje de alimentación de 5V listo para conectarlo al Arduino.
- Rango de detección 1mm – 8mm.
- Tiene dos salidas una digital y una analógica.
- La salida digital está en nivel 1 cuando un objeto es detectado.
- Led indicador de estado.
- Potenciómetro para ajustar sensibilidad

4.1.9. Motor DC 12V

Sainz, B. (2011), dice que el motor de corriente continua (motor DC) es una máquina que convierte la energía eléctrica en mecánica, provocando un movimiento rotatorio. En algunas modificaciones, ejercen tracción sobre un riel. Estos motores se conocen como motores lineales.

Una máquina de corriente continua (generador o motor) se compone principalmente de dos partes, un estator que da soporte mecánico al aparato y tiene un hueco en el centro generalmente de forma cilíndrica. En el estator además se encuentran los polos, que pueden ser de imanes permanentes o devanados con hilo de cobre sobre núcleo de hierro, como se muestra en la figura 12.



Figura 12. Motor de corriente continúa. Por Sainz, B. (2011),

4.1.9.1 Especificaciones del motor

Un motor dc 12v tiene las siguientes especificaciones como se muestra acontinuacion:

- Tensión Nominal 12V
- Velocidad Sin carga 100 rpm
- Consumo sin carga 140 mA
- Consumo nominal < 800 mA
- Velocidad nominal 73 rpm
- Fuerza nominal 3,5 kgf·cm / 0,34 Nm
- Fuerza a máxima eficiencia 11 kgf·cm / 1,08 Nm
- Consumo en frenado 3000 mA
- Potencia de salida 2,6 W
- Diámetro Máximo 37 mm
- Diámetro Eje 6 mm
- Rosca fijación 4 x M3

4.1.9.2. Sentido de giro

El sentido de giro de un motor de corriente continua depende del sentido relativo de las corrientes circulantes por los devanados inductor e inducido así como lo explica Sainz, B. (2011), la inversión del sentido de giro del motor de corriente continua se consigue

invirtiendo el sentido del campo magnético o de la corriente del inducido. Si se permuta la polaridad en ambos bobinados, el eje del motor gira en el mismo sentido. Los cambios de polaridad de los bobinados, tanto en el inductor como en el inducido se realizarán en la caja de bornes de la máquina, y además el ciclo combinado producido por el rotor produce la fuerza magneto-motriz, como se muestra en la figura 13.

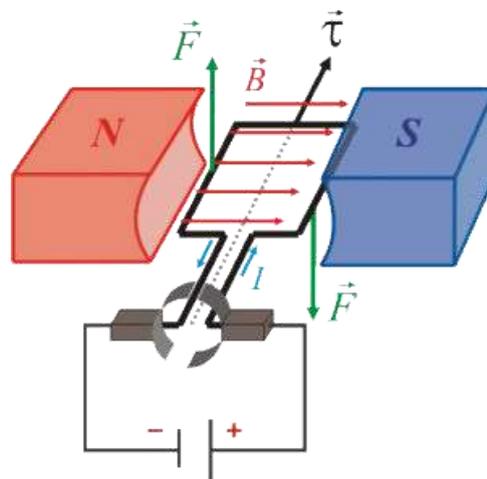


Figura 13. Sentido de giro de un motor DC. Por Sainz, B. (2011),

4.1.10. Módulo de relés de 8 canales

Según Galindo, A. (2018) refiere que el módulo de relés sirve para la comunicación de cargas de potencia, los contactos soportan cargas hasta corriente de 10A y una tensión hasta 250 VAC por lo general, se añade además que es recomendable trabajar por debajo de los límites de tensión. Esta placa funciona con una alimentación de 5V.

Un relé es un dispositivo diseñado para control remoto de otro dispositivo tomando en cuenta que un relé usa bajo voltaje y poca corriente para causar la conmutación de dispositivos de alto voltaje o alta corriente, usualmente se coloca a distancias remotas de estos. Las entradas de control se encuentran aisladas con optoacopladores que son dispositivos emisores y receptores que se activan mediante una luz que emita un diodo

LED. Las señales de control pueden ser enviadas desde cualquier microcontrolador o Arduino, como se observa en la figura 14.



Figura 14. Módulo de relés de 8 canales. Por Galindo, A. (2018)

4.1.10.1. Especificaciones:

El módulo relé para elaborar el proyecto tendrá las siguientes especificaciones:

- Voltaje entrada: 5 V
- Voltaje de control: 3.3 ~ 9 V
- Voltaje de salida: 250 VCA o 30 VDC
- Corriente a la salida: 10 A
- Dimensiones: 139 x 55 x 17 m

4.1.10.2. Características:

El módulo relé tiene las siguientes características como lo detallamos a continuación:

- 8 canales independientes protegidos con optoacopladores
- 8 relés (relays) de 1 polo 2 tiros
- El voltaje de la bobina del relé es de 5 VDC
- Led indicador para cada canal (enciende cuando la bobina del relé esta activa)

- Activado mediante corriente: el circuito de control debe proveer una corriente de 15 a 20 mA
- Puede ser controlado directamente por circuitos lógicos
- Terminales de conexión de tornillo (clemas)
- Terminales de entrada de señal lógica con headers macho.

4.1.11. Fuente de alimentación

Son las encargadas de convertir la corriente alterna en corriente continua, con salidas de 5 a 12 v cuenta conectores estándar que podrán ser utilizados para alimentar cualquier dispositivo o circuito electrónico, estas fuentes de alimentación son muy utilizadas, ya que se consigue reducir el tamaño del equipo notablemente y sobre todo reduce el calor del interior de los aparatos a utilizar, como se puede observar en la figura 15.



Figura 15. Fuente de alimentación. Por Martínez, M. (2018),

4.1.11.1. Etapas de las fuentes de alimentación

Según lo citado por Martínez, M. (2018), afirma que las etapas de las fuentes de alimentación son:

- La transformación es la etapa donde se reduce la tensión de (240v o 120v) a otra tensión baja, acorde para ser tratada y poder trabajar con corrientes alternas, es decir que la corriente de entrada sea igual a la de salida.
- El Rectificado es el encargado de convertir la tensión alterna en tensión continua para que no existan fluctuaciones de voltaje, es decir que el voltaje se mantenga sobre los 0v.
- El filtrado es la etapa donde se nivela la señal al máximo utilizando condensadores, reteniendo la corriente dejándole pasar poco a poco.
- La estabilización en esta etapa es donde se cuenta con una señal continua y casi del todo plana y solo es necesario estabilizarla.

4.2. Procedimiento – Metodología

4.2.1 Diseño

El sistema de clasificación automatizado por colores, está diseñado de forma que se puedan realizar prácticas, toda la caja de control donde se encuentran los elementos pueden ser manipulados por otro autómata con el fin de mejorar o reprogramar todo el sistema.

La máquina se encuentra constituida por sistemas eléctricos y mecánicos que interactúan entre sí para poder realizar el proceso deseado.

4.2.1.1 Sistema Eléctrico

La alimentación principal del circuito es de 110V y para los componentes electrónicos se utiliza una fuente de 9V a 2 amperios, los elementos que necesitan de un voltaje para cumplir su función, se realizan las distintas conexiones desde las borneras principales.

El circuito está diseñado con todas las medidas de seguridad correspondiente, los cables de conexión están internamente conectados aislados con taípe y extraíble con el fin de proteger las conexiones. El sistema eléctrico del sistema de clasificación está conformado de la siguiente forma, como se muestra en el figura 16.



Figura 16. Sistema eléctrico.

Fuente: Elaboración propia (2018)

4.2.1.1.1 Panel de Control y Energización

De manera esquemática, el panel de control de nuestra maqueta constará de los siguientes elementos:

- Selector dos posiciones ON / OFF para energizar todo el sistema
- Pulsador tipo hongo de emergencia color rojo
- Luz piloto verde
- Luz piloto roja.

De esta forma se ve conformado el panel de control que a continuación, se puede observar en la figura 17.



Figura 17. Panel de control

Fuente: Elaboración propia (2018)

4.2.1.1.2 Control Sensores y motores

Son las conexiones de cada uno de los aparatos a controlar, el cual son salidas, entradas y alimentaciones de los siguientes elementos:

- Sensor GY-31 de color para Arduino
- Sensor de obstáculos TCRT5000
- Motor DC 12V.
- Un motor a pasos Nema 17 bipolar para la clasificación y almacenamiento de los objetos.

4.2.1.1.3 Parte interna de la caja de control.

Dentro de la parte interna de la caja del control está formado por:

- Alimentación 110 v
- Fuente de 12v y 5A
- Relays 110v 8 pines
- Breaker bipolar 32A

Finalmente, el panel de control interno y externo ya implementado, como se muestra en la figura 18.



Figura 18. Panel de control interna

Fuente: Elaboración propia (2018)

4.2.1.2 Sistema Neumático

4.2.1.2.1 Electroválvulas

Está compuesto por tres electroválvulas monoestables, las mismas que al ser activadas eléctricamente deja pasar el aire para su respectiva aplicación. Es necesario tomar en cuenta que se necesita una presión mínima de 1,5-8 bar, por lo tanto desde 21,75 PSI hasta 116,03 PSI, como se puede observar en la figura 19.



Figura 19. Electroválvulas Monoestables

Fuente: Elaboración propia (2018)

4.2.1.2.2 Cilindros doble efecto

Los cilindros de doble efecto son capaces de producir trabajo útil en dos sentidos, ya que disponen de una fuerza activa tanto en avance como en retroceso. Se construyen siempre en formas de cilindros de embolo y poseen dos tomas para aire comprimido, cada una de ellas situada en una de las tapas del cilindro por lo que consta de tres cilindros neumáticos doble efecto para su funcionamiento, como se muestra en la figura 20.



Figura 20. Cilindro doble efecto.

Fuente: Elaboración propia (2018)

4.2.1.2.3 Unión Neumática tipo Y

Son elementos periféricos que tienen la función de interconectar elementos principales de los circuitos neumáticos, se utiliza una unión tipo Y de esta forma se obtiene una alimentación de aire para la otra electroválvula, como se muestra en la figura 21.



Figura 21. Unión Neumática tipo Y

Fuente: Elaboración propia (2018)

4.2.1.2.4 Compresor

Puesto que es una máquina diseñada para tomar el aire del ambiente, almacenarlo y comprimirlo dentro de un tanque llamado calderil y con este aire darle potencia a la máquina, se utiliza un compresor básico para que realicen su función las electroválvulas, como se muestra en la figura 22.



Figura 22. Compresor.

Fuente: Elaboración propia (2018)

4.2.1.2.5 Manguera Neumática

Son fabricadas en poliuretano y se utilizan para realizar la comunicación entre los diferentes elementos de un sistema neumático de tal forma que ayudará a que a través de la manguera neumática número 6, lleve aire por medio de ella, como se muestra en la figura 23.



Figura 23. Manguera Neumática

Fuente: Elaboración propia(2018)

4.2.1.3 Sistema Mecánico

4.2.1.3.1 Estructura

Para diseñar la estructura de la máquina se utiliza un tubo metálico el mismo que está elaborado de acero con medidas de 25*25 mm, como se puede observar en la figura 24.



Figura 24. Tubo cuadrado de acero

Fuente: Elaboración propia (2018)

4.2.1.3.2 Ejes cilíndricos

Se ocupan dos cilindros hechos de acero, utilizados para movilizar la banda desde el inicio hasta el final del proceso, es necesario destacar que los materiales para realizar los cilindros fueron obtenidos del reciclaje, para esto se aplica conocimientos obtenidos en el área de soldadura con el fin de diseñar cada uno de estos, como se aprecia en la figura 25.



Figura 25. Ejes cilíndricos

Fuente: Elaboración propia (2018)

4.2.1.3.3 *Chumaceras*

Se ocupó cuatro chumaceras, dos para cada uno de los ejes cilíndricos, con el fin de proporcionar apoyo al movimiento del motor con la banda y eliminar el contacto de metal con metal. Debido a que no existían unas chumaceras con las medidas para el eje, se tuvo que diseñar unas chumaceras utilizando tubo de acero, platinas y rodamientos, como se observa en la figura 26.



Figura 26. Chumaceras.

Fuente: Elaboración propia (2018)

4.2.1.3.4 *Banda transportadora*

La banda transportadora está diseñada en tela de gabardina, tomando las dimensiones necesarias para que esté precisa y no haya fallos al momento de templar la banda.

La función principal de la banda es soportar directamente el material a transportar y desplazarlo desde el punto de carga hasta el de descarga, razón por la cual se le puede considerar el componente principal de las cintas transportadoras, como se muestra en la figura 27.

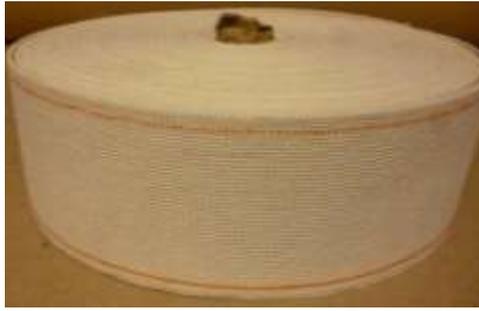


Figura 27. Banda transportadora.

Fuente: Elaboración propia (2018)

4.2.1.3.5 Acople para motor

El acople está diseñado en el torno utilizando un material de plástico resistente llamado duralon, este permitirá que el motor a pasos con el plato de almacenamiento emitiendo la fuerza para que gire, como se muestra en la figura 28.

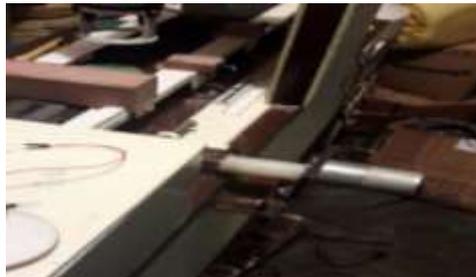


Figura 28. Acople de duralon.

Fuente: Elaboración propia (2018)

4.2.1.3.6 Pernos prisioneros

Los prisioneros son utilizados para la fijación del eje del motor a pasos, con el eje del plato de almacenamiento haciendo que todos los giros que realice el motor sean vistos al posicionarse en el color que le asigne el sensor, como se puede observar en la figura 29.



Figura 29. Prisioneros

Fuente: Elaboración propia (2018)

4.2.2 Solución del problema

El presente proyecto se enfoca principalmente en la automatización de estampado de objetos según sus colores, esto permitirá un trabajo eficiente, a través de máquinas sistematizadas e inteligentes reduciendo la carga laboral generando ahorro en las empresas.

La elaboración de este proyecto responde a una necesidad particular de una automatización para las industrias, pues actualmente los mercados altamente competitivos hacen que las empresas se vean obligadas hacer cambios respecto al enfoque de su trabajo. Ya que desean optimizar su sistema de producción, aumentando la productividad y mejorando la calidad del producto, la fabricación de estos modelos se sustentará a una maqueta previamente elaborada que cumpla con las funciones requeridas en el campo del estampado en madera, con el desarrollo del proyecto solo bastará que una persona accione el sistema para que comience a operar automáticamente. Todo el sistema es aplicable en diversas industrias, ya que mejorará la productividad de la empresa, destacando la calidad de la misma y reduciendo costos de producción.

4.2.3 Proceso de control

El proceso industrial de estampado dispondrá de un selector, que en una posición hará que el sistema esté apagado y en la otra forma empezará a funcionar haciendo activar una luz piloto de color verde como señal de funcionamiento.

Todo el proceso comienza sin ningún objeto, porque al momento de funcionar el sistema, también se activa una electroválvula empujando al primer objeto para que caiga en la banda transportadora y pase a su vez al sensor donde detectará el color mandando una señal al Arduino Mega, y este a su vez accione un relé que dará la señal al Zelio PLC para que se accione la electroválvula dando paso al aire para que funcione el cilindro que le corresponde al color que le haya asignado el sensor. En el transcurso que viaja el objeto por la banda tiene otro proceso interno que es el posicionamiento del motor a pasos donde va a almacenarse el objeto que está siendo estampado, después de un lapso de tiempo un último sensor da la señal para que se vuelva a repetir todo el proceso.

4.3 Construcción.

4.3.1 Estructura

La construcción de la estructura está diseñada con el fin de implementar los diversos elementos que conforman el sistema mecánico y neumático, que permitirá el control del movimiento del motor DC y de los actuadores neumáticos. La estructura tiene una medida reducida por motivos de reducir peso, también por la comodidad para realizar las conexiones correspondientes, como se muestra en la figura 30.



Figura 30. Estructura terminada

Fuente: Elaboración propia (2018)

4.3.2 Sistemas de control

Todo el sistema eléctrico se ubicó en la caja de control, utilizando canaletas para que los cables no queden a la vista y protegiéndolos del contacto directo e indirecto.

El Arduino Mega y el bloque de relés son ubicados encima de una base de acrílico, con el fin de aislar el contacto que puede tener con el metal, como muestra la figura 31. También es ubicado de esta manera para que los usuarios puedan acceder a la manipulación de estos elementos.



Figura 31. Sistema de control

Fuente: elaboración propia (2018)

El sistema de control cuenta con un breaker bifásico de 20 amperios directamente a las líneas de alimentación que dará el paso a la corriente, para que al momento de posicionar

al selector en encendido comience a funcionar todo el sistema incluyendo al Zelio PLC, como se observa en la figura 32.



Figura 32. Protecciones del Panel de control

Fuente: Elaboración propia (2018)

4.3.3 Programación en el software de Arduino

4.3.3.1 Programación de sensor de color GY-31

Para la programación del sensor se necesita asignar los pines de conexión con el Arduino, dándoles nombres a cada uno utilizando #define, en las siguientes líneas de programación se debe también definir las constantes para los colores, lo cual se establece con int, para que la variable tenga una lectura analógica, como se muestra en la figura 33.

```

#define s0 8
#define s1 9
#define s2 12
#define s3 11
#define out 10
// Variables
int red = 0;
int green = 0;
int blue = 0;
String Color;
int contColor=0;
int Sens=0;

void setup()
{
  pinMode(s0, OUTPUT);
  pinMode(s1, OUTPUT);
  pinMode(s2, OUTPUT);
  pinMode(s3, OUTPUT);
  pinMode(out, INPUT);
  digitalWrite(s0, HIGH);
  digitalWrite(s1, HIGH);
  Serial.begin(9600);
}

```

Figura 33. Programación sensor GY-31

Fuente: Elaboración propia (2018)

La función `setup ()` sirve para realizar las configuraciones iniciales, es decir debe contener la declaración de las variables y se utiliza para configurar o inicializar `pinMode` (modo de trabajo de las entradas y salidas), configuración de la comunicación en serie y otras.

Aquí se asigna a los pines que vienen del sensor al Arduino como salidas y al pin `out` como entrada de datos, este será el que lleve la información sobre qué color ha sido sentido emitiendo pulsos altos según el color para enviar los pasos respectivos que debe girar para posicionarse en el color que senso.

La función `loop ()` es el núcleo de todo el programa, porque contiene el código que se ejecutará continuamente, esta función es la que realiza la mayor parte del trabajo; en este caso se programa los valores de comparación de colores, ya que el sensor emite una frecuencia de 0 a 200, haciendo que el Arduino compare los valores de las frecuencias y proporcione un color fijo, como se muestra en la figura 34.

```

void loop()
{
  pinMode(
    Serial.println(" ");
    Serial.println(reyn, DEC);
    Serial.println(" ");
    Serial.println(reyn, DEC);
  }

  //Bucle inicio REINICIO
  if (reyn<10 || reyn>40 || reyn<20 || reyn>120 || reyn<40 || reyn>90)
  {
    Serial.println(" Rezo");
    V_salida = 4;
  }

  //Bucle inicio REINICIO
  if (reyn<10 || reyn>140 || reyn<20 || reyn>180 || reyn<40 || reyn>100)
  {
    Serial.println(" Rezo");
    V_salida = 4;
  }

  //Bucle inicio REINICIO
  if (reyn<140 || reyn>180 || reyn<20 || reyn>20 || reyn<20 || reyn>180)

```

Figura 34. Void setup y loop del sensor GY-31.

Fuente: Elaboración propia (2018)

4.3.3.2 Programación del sensor de obstáculo TCRT5000 (reinicio)

Al igual que la programación anterior incluye las mismas funciones (setup, loop). Para la programación del sensor TCRT5000 de obstáculos se necesita declarar nombres a las constantes, para lo cual la salida del sensor que ingresa al Arduino se le denomina reinicio y el pin que energizará al actuador se le denominará final; también se debe ubicar una constante que se denominará ‘valor’.

En el void setup se debe utilizar pinMode para indicar el modo de trabajo que tendrá los pines de reinicio (input) y final (output).

Para la parte del bucle (loop) se debe indicar al microprocesador que lea el pin de entrada y eso le iguale al constante valor; dicho esto, según el valor que emita el sensor si es un pulso alto este enviará a energizar al pin de salida del Arduino es decir al pin final haciendo funcionar la electroválvula que comanda al cilindro para poder tener un reinicio de todo el sistema, como se muestra en la figura 35.

```

sensor_de_obstulos$
const int REINICIO = 22;
const int FINAL = 23;
int valor;

void setup() {
  Serial.begin(9600);
  pinMode(FINAL, OUTPUT);
  pinMode(REINICIO, INPUT);
}

void loop() {
  valor = digitalRead(REINICIO);
  digitalWrite(FINAL, !valor);
  delay (100 );
}

```

Figura 35. Programación sensor de obstáculos.

Fuente: Elaboración propia (2018)

4.3.4 Programación en el software Zelio Soft

4.3.4.1 Programación del Zelio PLC

Para el diseño de la programación se debe tomar en cuenta primero las características del Zelio para poder seleccionar el módulo en el software y poder empezar a programar, como se muestra en la figura 36.



Figura 36. Programación Zelio PLC

Fuente: Elaboración propia (2018)

Después de haber seleccionado el módulo empezamos a ubicar las cuatro primeras entradas del Zelio que representan a cada uno de los colores las cuales ejecutará a cada una

de las salidas que se dirigen a las electroválvulas que darán paso al aire para accionar a los cilindros donde se encuentran los sellos, como se muestra en la figura 37.

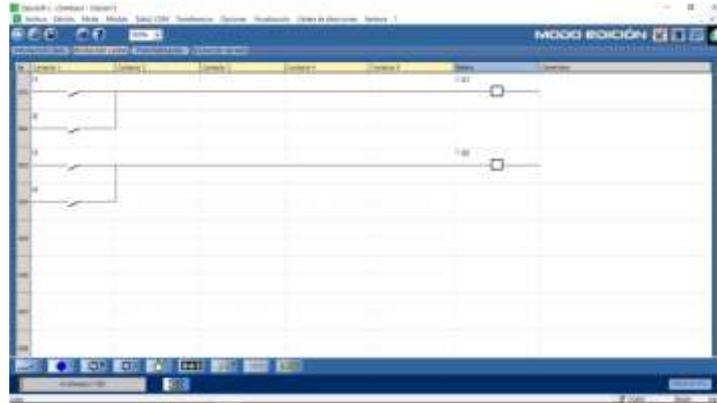


Figura 37. Líneas de Programación PLC

Fuente: Elaboración propia (2018)

4.4 Implementación.

El diseño de la implementación de el proyecto, tendrá una programación de lazo abierto, ya que no tendrá ninguna restricción por el programador, con el fin de que algún usuario pueda mejorar el mecanismo. Todo el sistema consta de cuatro etapas: mecánica, control, electrónica y neumática.

La parte mecánica, consta de la ubicación de los ejes cilíndricos y la unión del motor DC, para que se puedan transportar los objetos. Para templar la banda se coloca un perno, que le permitirá empujar la chumacera haciendo que se vaya apretando y templando a su vez.

La etapa de control parte desde la alimentación que ingresa directamente al breaker bifásico y desde aquí se toma una línea y pasa por el pulsador de paro tipo hongo para después pasar por el selector con el fin de energizar y desenergizar el sistema, también se usa dos relays el primero es usado para la alimentación y control de las luces piloto, la luz piloto verde servirá como aviso para saber que el sistema esta energizado y la luz roja es para saber que el sistema está paralizado.

La etapa electrónica, consiste en todos los sensores que se ubicaron y en la parte para el control del motor a pasos en esta etapa también, constituye la colocación del sensor en una posición en la que ya se vayan a quedar fijos para que no existan variaciones al momento de recibir las señales y tener errores. Para lo cual se ha realizado unas bases en MDF con una madera aislante y fácil de manipular; los sensores están internamente conectados y todos sus cables de conexión están cubiertos por canaletas. Haciéndolos llegar a la caja de control donde se conectan al Arduino Mega para que después realice las demás funciones. El Arduino es alimentado con una fuente de 9 voltios a 2 amperios para que el voltaje y amperaje sean abstraídos sin ningún problema.

La parte neumática consta de la alimentación de aire en donde se conectara el compresor en la parte interna se ubicó un acople tipo Y para la distribución a las electroválvulas y a partir de ahí conectar también a los cilindros neumáticos, todas las conexiones neumáticas se realizó con manguera azul número seis.

5 CONCLUSIONES

- El sensor GY-31 tiene la facilidad de programar mediante la frecuencia de los colores, para que al detectar el valor de cada frecuencia, agilite el proceso de estampado, además su costo es accesible comparado con otros sensores de las mismas características.
- El proceso de la selección de colores se realizará de mejor manera cuando los tazos que se desee sensor son claros, esto se debe que si los colores son débiles el sensor detectará como color negro.
- El diseño del proceso de estampada de colores funciona de mejor manera cuando el sensado de los colores de los objetos se les hace uno por uno, esto se debe a que si se hace seguido uno tras otro el sensor tiende a confundir las frecuencias haciendo que no de un color fijo y existirán fallas al momentos de sellar cada uno de los objetos.
- La sincronización de Arduino y PLC ayuda a controlar de mejor manera todo el proceso que se desee realizar aplicando los sensores requeridos de forma precisa.
- La aplicación de electroválvulas monoestables ayuda al mejor accionamiento de cada uno de los cilindros y tener una mayor precisión al estampar los objetos.

6 RECOMENDACIONES

- Para la programación del sensor de color, se debe tener en una base estática el sensor para realizar la calibración del sensor y así realizar una correcta comparación de frecuencias y obtener un valor correcto.
- Al conectar los cables hacia el Arduino se tiene que asignar colores de cable para diferenciar su polaridad en caso de que tenga el elemento una polaridad y para no perder los cables de las señales de los distintos sensores.
- Tener establecidos los colores que se desee sensar para poder fijar las frecuencias de los colores y así tener un buen desempeño del sensor, al momento de que emita las señales.
- Para el accionamiento de las electroválvulas es necesario que las salidas del PLC estén bien establecidas para que al momento de programación no se efectúen algún problema y no se estampen en los colores que no son los establecidos.
- Utilizar una presión de 60 psi que llegue a las válvulas, para que se accione correctamente y no tenga ningún retraso de accionamiento al momento de empezar el proceso con otro objeto y de sellarlos
- Es necesario que la banda este bien templada para que al momento de pasar los objetos por el sensor no exista movimientos bruscos y pueda realizar bien su trabajo de sensado.

7 FUENTES

7.2 BIBLIOGRAFÍA

- Aceves, A. (2011). Electroneumática. Recuperado en Marzo del 2018 de <https://dokumen.tips/documents/electroneumatica-55a232cdc8f3f.html>.
- Andrés, A. (2016). Diseño E Implementación De Un Sistema. Sangolqui: San Jose primera edicion.
- Arduino. (2018). Arduino. Recuperado el 18 de enero del 2018 de <https://store.arduino.cc/arduino-mega-2560-rev3>.
- Armesto, M. (s.f.). Autómatas programables y sistemas de automatización. Barcelona: MARCOMBO, S.A. 1º Edición.
- Burbano, P. (13 de Mayo de 2013). Didactica. Recuperado el 11 de Marzo de 2018, de <http://www.didactica.com/recursos/reciclaje>.
- Caldas, M. (Noviembre de 2016). hackster.io. Recuperado en Noviembre del 2017, de <https://www.hackster.io/MakerRobotics/color-sensor-gy-31-6b5f75>.
- Carbajal, M. (28 de Julio de 2014). Reportero Industrial. Recuperado el 20 Mayo del 2018 de <http://www.reporteroindustrial.com/temas/Evolucion-de-la-automatizacion-industrial+98784>.
- Docplayer. (2018). Circuitos neumáticos y oleohidraulicos. Recuperado en Marzo del 2018 de <http://docplayer.es/39288831-Bloque-iv-circuitos-neumaticos-y-oleohidraulicos.html>.
- Electronica Caldas, M. (2017). Sensores de luz visible. Recuperado en Diciembre del 2017, de <http://www.electronicoscaldas.com/sensores-de-luz-visible/947-gy-31-modulo-con-sensor-de-color-tcs3200.html>.
- Galindo, A. (2018). sumecan. Recuperado el 22 Julio del 2018, de http://sumecan.com/portfolio_page/modulo-rele-8-canales-5v-arduino/.
- Guillén, A. (2010). Ingeniería en soporte. Recuperado en Diciembre del 2017, de <http://www.guillesime.galeon.com/index.htm>.

- Hernández, L. (2018). programafacil. Recuperado el 20 de Febrero del 2018 de <https://programafacil.com/blog/breve-introduccion-al-arduino/>.
- Laguna, L. (2009). ESPOSH Tesis de grado. ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO.
- Llamas, L. (2018). wikipedia. Recuperado en marzo del 2018 de https://es.wikipedia.org/wiki/Sensor_infrarrojo.
- Martínez, M. (2018). ocw.usal. Recuperado en Mayo del 2018 de http://ocw.usal.es/enseñanzas_tecnicas/electronica/contenido/electronica/Tema4_F_alimentac.pdf.
- Moreta, G. (2012). repositorio. (Tesis de grado) Universidad Tecnica Del Norte Facultad De Educacion Ciencia Y Tecnologia.
- Prieto, P. (08 de octubre de 2007). recursostic. Recuperado en Enero del 2018 de <http://recursostic.educacion.es/observatorio/web/gl/component/content/article/502-monografico-lenguajes-de-programacion?start=2>.
- Porras, J. (2008). urp. Recuperado el 20 de Mayo del 2018 de http://www.urp.edu.pe/pdf/ingenieria/electronica/CAP_1_Taller_de_Electronica_I_V_b.pdf.
- Sainz, B. (2011). repositorio. Recuperado el 25 de Marzo del 2018 de http://repositorio.umsa.bo/bitstream/handle/123456789/15140/EG_1092_Ram%C3%ADrez%20Sainz%2C%20Beatriz.pdf?sequence=1&isAllowed=y.
- Schneider Electric. (2006). Relés programables compactos Zelio Logic.
- Valencia, J. (marzo de 2010). Repositorio. Recuperado en Febrero del 2018 de <http://repositorio.utc.edu.ec/bitstream/27000/604/1/T-UTC-1070.pdf>.
- varios. (2012). Libro básico de Arduino: electrónica y programación. quito: tienda de robotica.
- Vera, C. (2013). Sistema por sensor de color. Riobamba.

8 ANEXOS

8.2 Pintada de la estructura



Figura 38. Pintada de la estructura

Fuente: Elaboración propia (2018)

8.3 Implementación de los elementos de control



Figura 39. Elementos de control

Fuente: Elaboración propia (2018)

8.4 Sensor de obstáculo



Figura 40. Elementos de control

Fuente: Elaboración propia (2018)

8.5 Implementación de los elementos en la caja de control



Figura 41. Caja de control

Fuente: Elaboración propia (2018)

8.6 Cableado interno de la caja de control



Figura 42. Cableado de la Caja de control

Fuente: Elaboración propia (2018)

8.7 Implementación del motor DC con la banda



Figura 43. Motor DC conectado a la banda

Fuente: Elaboración propia (2018)

8.8 Unión de cables del sensor con el cable UTP



Figura 44. Soldadura con cables de conexión

Fuente: Elaboración propia (2018)

8.9 Tarjeta para controlar los PMW del motor DC



Figura 45. Tarjeta para controlar los PMW del motor DC

Fuente: Elaboración propia (2018)

8.10 Prueba de programación de sensor de obstáculos

```
sensor_ds_obstaculos
const int REINICIO = 22;
const int FINAL = 23;
const int REDONDO = 24;
const int LED2 = 25;
const int CUADRADO = 26;
const int LED = 27;
int valor;
int valor1;
int valor2;

void setup() {
  Serial.begin(9600);
  pinMode(FINAL, OUTPUT);
  pinMode(REINICIO, INPUT);
  pinMode(LED2, OUTPUT);
  pinMode(REDONDO, INPUT);
  pinMode(LED, OUTPUT);
  pinMode(CUADRADO, INPUT);
}

void loop() {
  valor = digitalRead(REINICIO);
  digitalWrite(FINAL, !valor);
  delay(100);
  valor1 = digitalRead(CUADRADO);
  digitalWrite(LED, !valor1);
  delay(100);
  valor2 = digitalRead(REDONDO);
  digitalWrite(LED2, !valor2);
  delay(100);
}
```

Figura 46. Programación sensor de obstáculos

Fuente: Elaboración propia (2018)

8.11 Prueba de programación de unión de sensores



```
union_de_sensores Arduino 1.8.3
Archivo Editar Programa Herramientas Ayuda

union_de_sensores
const int a1 = 0;
const int a2 = 9;
const int a3 = 10;
const int a4 = 11;
const int out = 10;
// LED pins connected to Arduino
int redLed = 7;
int greenLed = 6;
int blueLed = 5;
int ledPin1 = 3; // pin de un LED
int ledPin2 = 3; // pin de otro LED
int infraPin = 4; // pin del infrarrojo utilizado como entrada digital
int valorInfra = 0; // Valor inicial de la lectura digital del infrarrojo.
// Variables
int red = 0;
int green = 0;
int blue = 0;
void setup() {
  Serial.begin(9600);
  pinMode(a1, OUTPUT);
  pinMode(a2, OUTPUT);
  pinMode(a3, OUTPUT);
  pinMode(a4, OUTPUT);
  pinMode(out, OUTPUT);
  pinMode(greenLed, OUTPUT);
  pinMode(redLed, OUTPUT);
  pinMode(blueLed, OUTPUT);
  digitalWrite(a1, HIGH);
  pinMode(ledPin1, OUTPUT); // Inicializa el pin del LED1 como salida digital.
}
```

Figura 47. Programación de unión de sensores

Fuente: Elaboración propia (2018)



```
union_de_sensores Arduino 1.8.3
Archivo Editar Programa Herramientas Ayuda

union_de_sensores
pinMode(infraPin, INPUT); // Inicializa el pin
}

void loop() {
  valorInfra = digitalRead(infraPin); // Lee el va
  digitalWrite(ledPin1, valorInfra); // Escribe e
  Si el infrar
  Si el infrar
  // Se asigna
  digitalWrite(ledPin2, valorInfra); // Escribe e
  Si el infrar
  Si el infrar

  color();
  Serial.print("R Intensity:");
  Serial.print(red, DEC);
  Serial.print(" G Intensity: ");
  Serial.print(green, DEC);
  Serial.print(" B Intensity : ");
  Serial.print(blue, DEC);
  //Serial.println();

  if (red < blue && red < green && red < 20)
  {
    Serial.println(" - (Red Color)");
    digitalWrite(redLed, HIGH); // Turn RED LED ON
    digitalWrite(greenLed, LOW);
    digitalWrite(blueLed, LOW);
  }

  else if (blue < red && blue < green)
```

Figura 48. Programación de unión de sensores

Fuente: Elaboración propia (2018)