

DISEÑO E IMPLEMENTACIÓN DE UN ENTORNO DE DESARROLLO DE DIAGRAMAS EN ESCALERA PARA ARDUINO

Inzunza Villagomez Héctor Iván¹, Pérez Arce Beatrice², Hernández Ruiz Sergio Iván³, López Corella José Alejandro⁴, Vázquez Cuevas Ignacio Javier⁵, Herrera Velarde Maribel⁶

Tecnológico Nacional de México / I.T. de Nogales, Metal Mecánica Ingeniería Mecatrónica
Av. Tecnológico 911, Nogales, Sonora, México
+52(631) 311 1881. Ext 1129.
sergiohernandez@itnogales.edu.mx

RESUMEN.

La finalidad del proyecto es presentar una alternativa para la programación de tarjetas Arduino a través de un entorno de desarrollo en diagrama en escalera. De igual manera, se busca ofrecer una nueva opción para el uso académico y de procesos de automatización que requieran de soluciones que no demanden muchos recursos. El producto de este objetivo es un prototipo de PLC basado en Arduino, el cual se distingue por ser compacto, económico y fácil de controlar. Es programado mediante una IDE, desarrollada en Visual Studio con el lenguaje C#, y cuenta con herramientas básicas e intuitivas para la construcción de programas, contando con una interfaz amigable y sencilla que facilita su uso.

Palabras Clave: Arduino, C#, Visual Studio, PLC, diagrama en escalera.

ABSTRACT.

The purpose of this project is to present an alternative for the Arduino boards programming through a ladder diagram development environment. Similarly, it seeks to provide a new option for both academic and automation process applications for effective solutions without the use of extensive resources. The product of this objective is a PLC prototype based in Arduino, which is distinguished by being compact, economical and easy to control. It's programmed through an IDE, developed by Visual Studio with the C# programming language and has basic and intuitive tools for program development and also has a single and friendly interface to ease its use.

Keywords: Arduino, C#, Visual Studio, PLC, diagrama en escalera.

1. INTRODUCCIÓN.

En la industria moderna, la automatización es uno de los pilares que sostiene la producción de la mayoría de los productos que regularmente consumimos y utilizamos, desde los jugos en lata hasta los automóviles en los que nos trasladamos [1].

Los procesos automatizados son sistemas que facilitan el control de las producciones en serie, debido a que agilizan el traslado y desarrollo de actividades, así como el ensamblado o el recubrimiento con pinturas de los productos. Asimismo, pueden disminuir costos a largo plazo, puesto que las

detecciones de errores humanos suelen ser fáciles de controlar y calibrar.

Los responsables de controlar los procesos de automatización son los sistemas de control, y estos últimos son tan importantes para el proceso como lo son los motores para los automóviles.

Con el avance de la tecnología los sistemas de control han evolucionado, aumentado su versatilidad para adaptarse a diversos procesos reduciendo su tamaño y complejidad. Los beneficios anteriormente mencionados han atraído a ciertos líderes o jefes de negocios que se interesan en tener un control más preciso al ofrecer o vender sus productos y/o servicios. Sin embargo, una limitante que se suele presentar para aquellos dueños de pequeños negocios que desean mejorar la producción y rendimiento procesos mediante la automatización, es el elevado costo que tienen los controladores, tales como los controladores lógicos programables (PLC).

Un PLC es una computadora que se utiliza para automatizar procesos electromecánicos. Se caracterizan por ser prácticas para programar y modificar, evitando costos adicionales y ahorrando tiempo en la elaboración de proyectos. El lenguaje en escalera, también conocido como lenguaje Ladder o diagrama de contactos, es el más popular en la industria para su programación [2,3].

2. MARCO DE REFERENCIA.

2.1. Arduino.

Arduino es una compañía que fabrica placas de desarrollo basadas en su mayoría en microcontroladores Atmel AVR. Dichas placas son circuitos impresos que facilitan el manejo de los periféricos de los microcontroladores.

De igual manera, la compañía Arduino ofrece un entorno de desarrollo (IDE) donde los usuarios pueden programar con una computadora el microcontrolador de la placa. Esta IDE es de código abierto y está basado en el entorno de processing y en la estructura del lenguaje de programación Wiring [4,5].

2.2. Visual Studio.

Microsoft Visual Studio es un entorno de desarrollo que ofrece herramientas y servicios para crear aplicaciones y sitios web

para cualquier plataforma. Esta IDE puede trabajar con múltiples lenguajes de programación, como lo son C++, C# y java. Las aplicaciones creadas en este entorno disponen una amplia gama de herramientas que permiten la manipulación, eliminación y creación de archivos y base de datos, la comunicación serial con distintos dispositivos y el intercambio de información con otros programas [6].

2.3. Controlador lógico programable (PLC).

Un PLC es un autómata programable diseñado para el control de procesos industriales siguiendo las instrucciones de un programa cargado en la memoria del dispositivo. A diferencia de una computadora comercial, estos dispositivos controlan múltiples periféricos donde se pueden conectar equipos, maquinaria y/o componentes eléctricos y electrónicos. Pero al igual que una PC, estos equipos pueden tener grandes capacidades de almacenamiento y procesamiento de información [2,7].

2.4. Diagrama en escalera.

Los diagramas en escalera también son conocidos como *ladder logic* y lenguaje *ladder* (figura 1). Es un lenguaje gráfico estandarizado para la programación de controladores lógicos y se basa en los esquemas eléctricos de control clásicos. Una de las ventajas del lenguaje *ladder* es su fácil entendimiento para un técnico o ingeniero. Para la construcción de estos diagramas se utilizan múltiples herramientas, tales como contactos abiertos, contactos cerrados, temporizadores, entre otros [2,7].

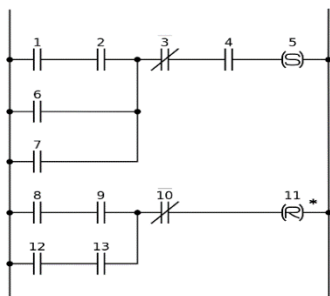


Figura 1. Ejemplo lenguaje ladder.

3. SOFTWARE DESARROLLADO.

3.1. HT-PLC (IDE).

HT-PLC es un software, elaborado en Visual Studio, que permite la creación de diagramas en escalera para después ser cargados y ejecutados en una tarjeta Arduino.

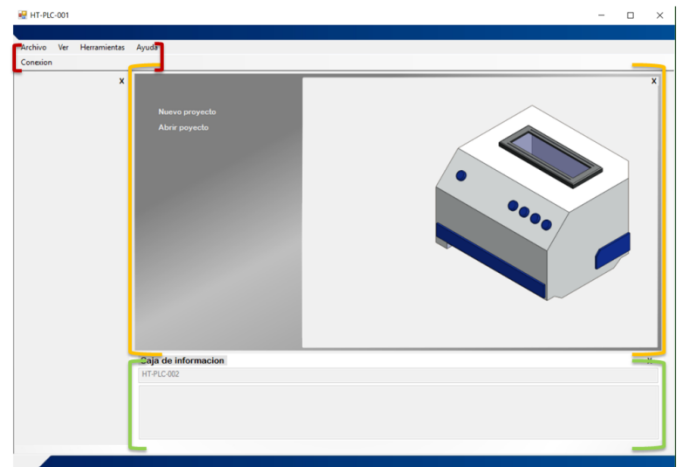
El programa contiene herramientas que permiten la comunicación del PLC mediante un puerto de comunicación serial de la computadora.

3.2. Características generales.

La IDE posee la opción de crear proyectos que pueden ser guardados en la memoria de la computadora, al igual que

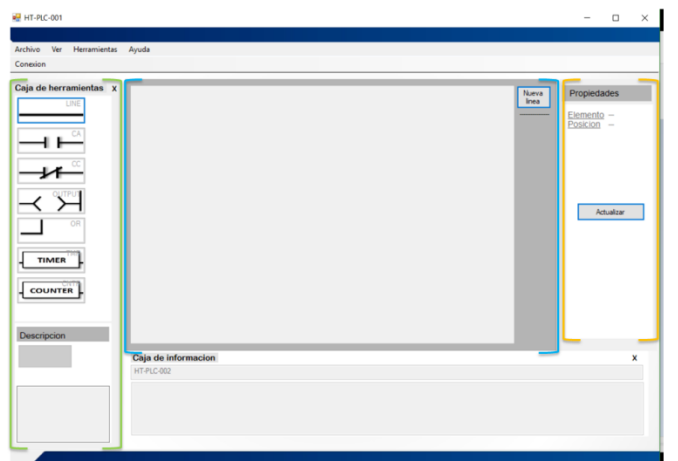
posibilita la modificación de archivos y enviarlos al PLC si así se desea.

Para la elaboración de proyectos, la IDE cuenta con herramientas, secciones de trabajo, y un entorno intuitivo (figura 2 y figura 3).



- Accesos directos, herramientas y ayuda
- Menú principal
- Caja de información

Figura 2. Menú de inicio.



- Caja de herramientas
- Sección de programación
- Caja de propiedades

Figura 3. Secciones de trabajo

3.3. Registros.

Un registro es un espacio de memoria donde se puede leer y/o escribir información binaria para interactuar con los periféricos del PLC. Existen 5 tipos de registros en la presente plataforma,

cada uno de estos se encargan de administrar distintas funciones (Tabla 1).

Tabla 1. Registros de programa.

Registro “A/a” salidas (L/E) (1 a 8)	Recibe la función de control de salidas.
Registro “B/b” entradas (L) (1 a 8)	Refleja los datos de las entradas.
Registro “C/c” bobinas internas (L/E) (1 a 77)	Es responsable de activar y desactivar las bobinas internas.
Registro “e” Temporizador (L)	Se encarga de leer los eventos de temporizador.
Registro “d” Contador (L)	Obtiene la información de los eventos contador.

3.4. Caja de herramientas.

La caja de herramientas es el apartado del programa en el que se ubican las herramientas para la construcción del diagrama en escalera. Aparece en pantalla automáticamente al seleccionar la opción de crear un nuevo proyecto o el de abrir uno ya existente, dejando a plena vista las herramientas que se pueden utilizar (Tabla 2).

Tabla 2. Clasificación de herramientas.

Herramientas de control	
Contacto Abierto(CA) / Contacto Cerrado(CC)	Los contactos sirven para la lectura específica de la posición de un bit de un registro. Si el bit en cuestión es igual a “1” el contacto cerrado se abrirá, y si es un contacto abierto, este se cerrará. Si el bit en cuestión es igual a “1” el contacto cerrado se abrirá y/o contacto abierto se cerrará. En caso contrario, en el que reciban un bit con valor “0”, se mantendrán en su estado normal, abierto y cerrado respectivamente.
Salida (OUT)	El elemento de salida se ocupa de escribir la posición de un bit de un registro.
Herramientas de unión	
Línea (LINE)	La línea es una herramienta que puede borrar cualquiera de los otros recursos colocados en la sección de programación.
Unión “OR” (OR)	La unión “OR” habilita el vínculo

	de una o más líneas a otra.
Herramientas de eventos	
Temporizador (TMR)	El temporizador es aquel que activa momentáneamente la posición del registro “e” cuando pasa un tiempo determinado, en el momento que el evento se ha repetido una cierta cantidad de veces, se activa la otra posición del mismo registro. El usuario es quien define la duración y las repeticiones del temporizador.
Contador(CNTR)	El contador envía un bit “1” en el registro “d” cuando se ha activado un número determinado de ciclos, definido por el usuario.

3.5. Sección de programación.

La sección de programación (figura 4) es aquella que está dedicada a la construcción de diagramas en escalera. Al iniciar un nuevo proyecto el área aparecerá en blanco, y para empezar a trabajar es necesario presionar el botón “Nueva línea” cada vez que se desee agregar una línea nueva.

El apartado se divide en líneas compuestas de 8 bloques en las que se les puede asignar uno de los objetos disponibles. A su vez, los bloques se dividen en dos grupos: 7 de lectura ubicados en la parte izquierda, pudiéndose asignar solamente los recursos de CA, CC, LINE y OR; y el último de escritura que está situado en el lado derecho, en el que solo se pueden insertar las herramientas OUT, TMR y CNTR.

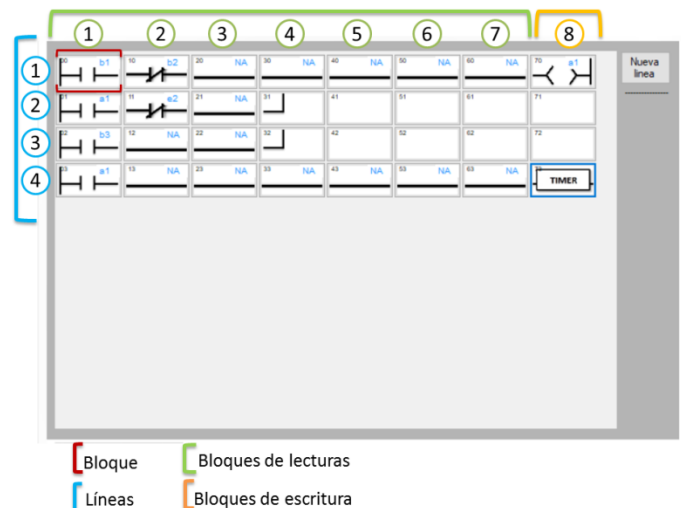


Figura 4. Sección de programación

3.6. Caja de propiedades.

La caja de propiedades asigna la información con la que trabajará cada bloque, dependiendo del elemento seleccionado. Las herramientas de control escribirán o leerán una posición específica de un registro y de igual modo pueden ser designadas a una bobina, si es permitido por el objeto elegido. Para indicar dicho registro es obligatorio introducir una letra seguida de un número en la caja de texto denominada "Bobina", localizada en el apartado de propiedades. La letra representa el registro y el número declara la posición de lectura. En el caso a las herramientas de unión, no se les es posible introducir o editar ningún tipo de información.

Por otro lado, las herramientas de eventos aceptan la introducción de datos, cuyas opciones o variables cambian por el tipo de evento. En el temporizador es necesario establecer dos valores: el valor del intervalo, que define el tiempo en el que el primer bit del registro "e" tendrá el valor de "1"; y el valor de repeticiones, el cual establece el número de ocasiones en el que se repite el intervalo del temporizador para poner el segundo bit del registro "e" en "1". Con el contador solo se especifica la cantidad de veces que debe ser activado momentáneamente para poner en "1" el primer bit del registro "d".

3.7. Caja de información.

La caja de información se encuentra en la parte inferior de la pantalla. Se encarga de brindar información de las operaciones y errores que se pueden producir, tales como las fallas en la conexión, la actualización de datos, entre otros. Con el fin de asistir al usuario en la aparición de errores, se ha colocado en la sección de ayuda una opción llamada "Lista de errores" que contiene la descripción y los motivos por los que se producen.

3.8. Construcción del diagrama en escalera.

La elaboración de un diagrama de contactos consiste en una serie de pasos muy sencillos: lo primero es abrir un nuevo proyecto desde el menú principal o en la pestaña de "Archivo"; lo siguiente es colocar una nueva línea presionando el botón "Nueva línea" en la sección de programación; luego se introducen las herramientas en los bloques, lo cual se logra haciendo un clic en el elemento y otro clic en el espacio que se prefiera; lo que sigue es determinar la información del bloque seleccionándolo con el cursor, mostrando la caja de propiedades que da acceso a los aspectos editables del objeto; por último, se presiona el botón "Actualizar" en el mismo apartado y los valores son reflejados en la caja de información. Posteriormente a la conclusión del diagrama, el programa puede ser guardado en la memoria de la computadora como archivo de texto, para ello se selecciona en la pestaña "Archivo" donde dice "Guardar". De modo que si se quiere abrir un programa guardado se va de nuevo a la pestaña de "Archivo" y se elige la opción "Abrir", o como alternativa se puede ir al menú principal y cargar el archivo.

3.9. Conexión de con el Hardware.

La conexión de la computadora al Arduino se realiza mediante el uso de un cable USB (Tipo A-B) que se conectan a sus respectivas entradas. Al hallarse en contacto, en la sección de "Puerto" se busca la pestaña "Conexión" y se hace clic para escoger el puerto serial de Arduino. Existe la posibilidad de que no aparezca y, de ser así, se deberá presionar el botón de "Actualizar". En caso de que se haya establecido el puerto serial, solo se hace clic en "Conectarse" y completar exitosamente la conexión. Si el programador decide cargar la programación al hardware después de haber hecho lo anterior, únicamente se precisa que el botón "Cargar proyecto" sea presionado.

4. HARDWARE.

4.1. HT-PLC-001.

Con el objetivo de facilitar el uso de una tarjeta Arduino como controlador lógico programable, se ofrece un prototipo de PLC basado en este microcontrolador, denominado "HT-001". Aunque el programa puede adaptarse a cualquier placa de la marca, este prototipo es controlado por un Arduino uno y un Arduino nano con el propósito de expandir los periféricos. Los programas del software pueden ser cargados a este controlador, los cuales son guardados en la memoria EEPROM del microcontrolador, esto quiere decir que una vez cargado el programa, aunque el PLC se reinicie el último programa cargado, no se perderá pero se borrará al cargar uno nuevo.

4.2. Funcionamiento.

El HT-001 cuenta con un módulo "puente H" utilizado como regulador de voltaje para alimentar un Arduino uno, un Arduino nano, una pantalla LCD 20x4 y un módulo de 8 relevadores (Figura 4). El Arduino uno imprime la información del menú en la pantalla LCD, recibe la información de las entradas digitales y manda la información del estado de los relevadores al Arduino nano, mientras este responde con la información sobre los estados de los botones.

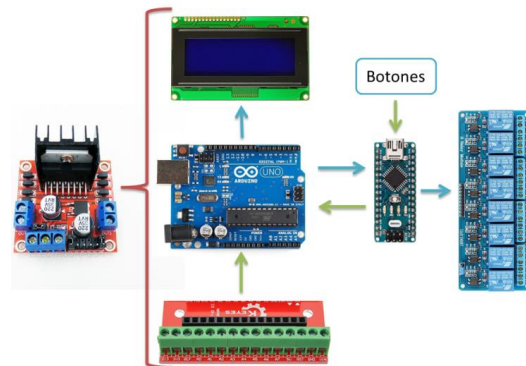


Figura 5. Estructura interna del PLC 1.

4.3. Estructura y control.

El PLC cuenta con un total de 8 entradas digitales, 8 salidas de relevador, 5 botones para el control del menú, un par de luces led, un botón de encendido/apagado, una entrada para la alimentación de 12V y una pantalla LCD (figura 6).

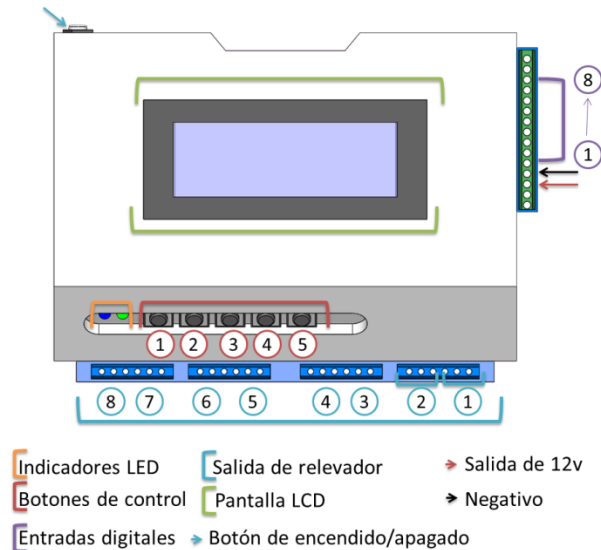


Figura 6. Estructura física del PLC.

Si se desea encender el controlador es necesario conectar el PLC a una fuente de alimentación de 12V y después accionar el botón de encendido que se encuentra en su parte posterior. Si el led verde se enciende y apaga cuatro veces en intervalos iguales, indica que el PLC está funcionando correctamente, y si el led azul permanece encendido indica que se le está otorgando una alimentación correcta.

4.4. Pantalla.

El controlador dispone de una pantalla LCD 20x49 (figura 7) que está dividida en 4 líneas: La primera, ubicada en el extremo derecho, informa si un programa está en ejecución o no; la segunda y tercera, localizadas en el centro, contienen las opciones del menú; y en la cuarta se obtiene la información de los botones.

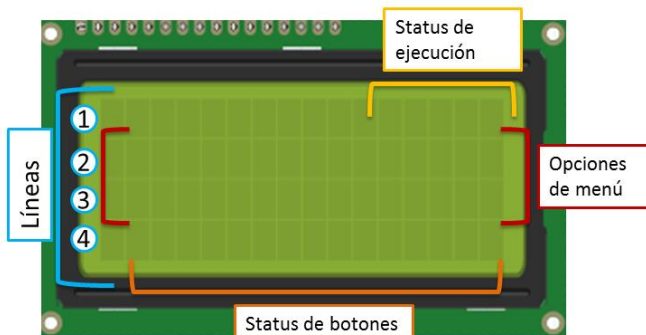


Figura 7. Pantalla LCD de PLC 1.

La interfaz está dividida en el menú principal (menú 1) y el menú secundario (menú 2), contando cada uno con dos opciones.

El PLC siempre inicia con el menú 1, donde se encuentran las opciones de “Leer programa” y “Status”, mientras que en el menú 2 están las opciones de “Borrar programa” y “Ejecutar programa”.

4.5. Botones.

El botón 1 sirve para seleccionar la opción deseada. El botón 2 y 3 permiten al usuario desplazarse entre el menú 1 y el menú 2, Y por último los botones 4 y 5 sirven para subir y bajar el cursor de selección.

4.6. Opciones.

La opción de “Leer programa” hace que el PLC cargue el proyecto hecho en la PC, teniendo como único requisito que el PLC esté conectado antes a la computadora. Automáticamente el programa se guarda en la memoria EEPROM de Arduino.

La opción de “Status” sirve para ver en tiempo real el estado de algunos registros, tales como los registros de salidas, entradas, entre otros.

La opción de “Borrar programa”, que se encuentra en la segunda línea del menú 2, borra el último programa cargado de la memoria EEPROM, dejándola completamente vacía.

Para ejecutar un programa que ya ha sido cargado es necesario dirigirse a la opción de “Ejecutar programa”. Una vez seleccionado, el programa se ejecuta automáticamente. En caso de querer detenerlo solo se tiene que volver a elegir la misma opción.

5. RESULTADOS.

Se diseñó e implementó la plataforma HT-PLC con el fin de ser una opción tanto académica como industrial, y contar con una opción robusta y económica para el funcionamiento de Arduino como PLC.

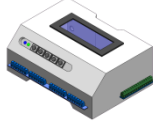


Para que el diagrama en escalera (un entorno de programación gráfico) pueda ser interpretador por el lenguaje de Arduino, se hace uso de una String (cadena de caracteres) que es enviada al PLC mediante el puerto de comunicación serial. Esta cadena está compuesta por secciones las cuales representan las líneas de programación y a su vez, cada sección contiene un total de 16 caracteres (2 caracteres por cada bloque). Cuando se está construyendo un diagrama en escalera en el software y se actualiza la información de un bloque se sobrescriben los dos caracteres correspondientes a este. Una vez que el programa ya es enviado a la memoria del PLC y es ejecutado, el Arduino empieza a leer de dos en dos los caracteres de una sección. Cada par de caracteres indica una dirección y el registro de un

bit en específico que se va a leer (por ejemplo: el registro B en la dirección/entrada 5) para posteriormente ser comparado con los siguientes bits y para producir un resultado que se sobrescribirá sobre una dirección y un registro específico determinado por los últimos dos caracteres de la sección en cuestión (los cuales representa los bloques de escritura).

HT-PLC facilita el control de los periféricos y no necesita una configuración previa para su funcionamiento, lo único que requiere es tener un programa cargado para la lectura y ejecución de los programas creados en el software.

En la tabla 3 se hace una comparación con dos controladores lógicos programables que se encuentran actualmente en el mercado y que poseen recursos básicos similares.

Tabla 3. Comparación de PLC 1.

Modelo	HT-PLC-001	LOGO! 0BA7	PLC Micro810
			
Entradas	8	12	8
Salidas	8	8	4
Herramientas básicas	5	8	
Herramientas especiales	2	30	
Costos de hardware (dólares)	450	2700	4700

Este trabajo explica el funcionamiento general y la estructura del software y del hardware que permiten la construcción de diagramas en escalera y su ejecución en un PLC basado en una tarjeta Arduino y, como resultado, se ha obtenido un controlador práctico y un software amigable que permite la programación de este PLC.

6. CONCLUSIONES.

El objetivo de este trabajo fue desarrollar una plataforma (software-hardware) que permitiera la programación de tarjetas Arduino en lenguaje *ladder* para la posterior aplicación de este microcontrolador como PLC. Se presenta de manera general el contexto y los requerimientos mínimos necesarios para el funcionamiento y comunicación del sistema, cumpliendo satisfactoriamente.

A futuro se pretende integrar al dispositivo la tarjeta *Wi-Fi Shield*, para dotar a la plataforma con capacidades de conexión a red y así una mayor adaptabilidad a las necesidades y a los procesos de producción industrial, en específico la industria 4.0.

7. REFERENCIAS

- [1] Enrique Mandado Pérez, “Autómatas programables y sistemas de automatización” Ed. Marcobo, 2009, ISBN-13: 978-84267-1575-3.
- [2]<https://www.idec.com/ca/en/products/Catalogs/Software/AO/overview.html>
- [3] López Corella José, Automatización del proceso de moldeo por inyección de plásticos de la maquina BOY 25D, Congreso Internacional en Ingeniería Electrónica Electro 2017, Vol. 39, pp. 11-16, ISSN 1405-2172.
- [4] <https://www.arduino.cc/>
- [5] Hernández R. Sergio, Sistema Distribuido con Arduino, wifi y LabView, utilizando el protocolo UDP, congreso Internacional en Ingeniería Electrónica Electro 2017, Vol. 39, pp. 22-26, ISSN 1405-2172.
- [6]<https://www.visualstudio.com/?rr=https%3A%2F%2Fwww.google.com%2F>
- [7] <https://ab.rockwellautomation.com/>
- [8] Hernández R. Sergio, Sistema Distribuido con Arduino y LabView, utilizando el protocolo UDP, congreso Internacional en Ingeniería Electrónica Electro 2014, Vol. 36, pp. 352-355, ISSN 1405-2172.