



INSTITUTO TECNOLÓGICO DE TUXTLA GUTIÉRREZ

DEPARTAMENTO DE ELECTRICA Y ELECTRÓNICA

INGENIERÍA ELECTRÓNICA

Nombre del proyecto:

Sistema para Monitoreo y Control de un Animatronic Vía Wifi (Reportador)

Nombre:

Limberg Martín Díaz Vázquez

Asesor:

M.C. RAÚL MORENO RINCÓN

Período de realización:

Agosto-Diciembre 2018

Contenido

CAPÍTULO I.....	1
1.1 INTRODUCCIÓN	1
1.2 JUSTIFICACIÓN	2
1.3 OBJETIVO	2
OBJETIVO GENERAL	2
OBJETIVO ESPECÍFICO	2
1.4 PROBLEMAS A RESOLVER.....	3
1.5 ALCANCES Y LIMITACIONES	3
CAPÍTULO II	4
2.1 CARACTERIZACIÓN DEL ÁREA EN QUE SE PARTICIPÓ	4
2.2 ANTECEDENTES DE LA EMPRESA.....	4
2.3 ORGANIGRAMA DEL INSTITUTO TECNOLÓGICO DE TUXTLA GUTIÉRREZ	5
2.4 MISIÓN, VISIÓN Y VALORES	6
MISIÓN.....	6
VISIÓN	6
VALORES.....	6
2.5 DESCRIPCIÓN DEL ÁREA DONDE SE REALIZÓ EL PROYECTO.....	6
CAPÍTULO III	7
3.1 FUNDAMENTO TEÓRICO	7
3.2 AVANCES EN LA TECNOLOGIA.....	7
3.3 ANIMATRONIC.....	8
3.4 COMPONENTES DE UN AUTOMATISMO	10
3.5 IOT	14
3.6 ARDUINO	15
3.7 SERVIDOR WEB.....	20
3.8 SUBLIME TEXT	21
3.9 HTML.....	21
3.10 XAMPP.....	24
CAPÍTULO IV.....	25
4.1 PROCEDIMIENTO Y ACTIVIDADES REALIZADAS.....	25
4.2 CRONOGRAMA DE ACTIVIDADES	25
4.3 REALIZACIÓN DE UNA PÁGINA WEB	26

4.4 XAMPP	31
4.5 CONEXIÓN	33
4.6 PROGRAMACION DEL ARDUINO	35
CAPÍTULO V	38
5.1 RESULTADOS, PLANOS, GRAFICAS, PROTOTIPOS Y PROGRAMAS	38
CONCLUSIÓN	42
BIBLIOGRAFÍA	43
ANEXOS	44

Índice de Figuras

Fig. 1. Animatronic vulture in exhibition. Fuente: (Edward,2018).....	8
Fig. 2. Animatronic de Abraham Lincoln en Disneyland	9
Fig. 3. Bloques internos que componen un PLC. Fuente:(Daneri,2008)	11
Fig. 4. Cilindro de simple efecto	12
Fig. 5. Cilindro de doble efecto	12
Fig. 6. contacto reed:diagrama y simbolo	13
Fig. 7. block diagram	14
Fig. 8. Real Topology	15
Fig. 9. Arduino Ethernet Shield. Fuente: (Arduino, Arduino Ethernet Shield, 2015)	16
Fig. 10. CHIP WIZNET W5100. Fuente: (WIZnet Co, 2008).....	17
Fig. 11. COMUNICACIÓN SPI. Fuente: (López Pérez, S. F.).....	18
Fig. 12. Entorno de desarrollo en Arduino	19
Fig. 13. Servidor.....	20
Fig. 14. Sublime Text	21
Fig. 15. XAMPP	24

CAPÍTULO I

1.1 INTRODUCCIÓN

En la actualidad hay un alto potencial para los sistemas y las aplicaciones enfocados a mejorar la operatividad de los procesos que integren sensores con técnicas de comunicación vía internet, para el monitoreo, análisis y estadísticas. Gracias a los avances tecnológicos que permite mejorar considerablemente la calidad de los procesos en máquinas compactas o aplicaciones de pequeño alcance.

En este proyecto se pretende desarrollar un prototipo basado en PLC con comunicación al arduino, para la adquisición de datos vía Ethernet con referente al monitoreo de los sensores para el registro de los movimientos del animatronic.

El monitoreo del estado del animatronic puede beneficiarse del empleo de tecnología del Internet of Things (IOT). Con capacidad de conexión en red y procesamientos que permitirán que se comuniquen entre sí y con otros dispositivos a través de internet

El sistema en su conjunto contará con un eficiente sistema de adquisición de datos, que se obtendrá mediante la comunicación del PLC y arduino, con el animatronic que serán almacenados en una base de datos creada, que servirá para el monitoreo y control.

1.2 JUSTIFICACIÓN

Conocemos que en la actualidad existen siempre necesidades o problemas que surgen y que se pueden solucionar planteando diversas ideas de solución, en este caso se establece una solución para el monitoreo del animatronic. Que tendrá como objetivo el monitoreo de los resultados obtenidos.

El sistema de adquisición de datos está pensado para que se observe desde una página web, para monitorear los resultados obtenidos de los sensores a la hora que el animatronic realice una acción concreta y registrar los datos obtenidos durante su funcionamiento.

Se encargará de analizar la eficiencia y fiabilidad de la obtención de los datos, en la realización de su funcionamiento del sistema encargándose en mejorar en su funcionamiento con los análisis de los registros obtenidos.

1.3 OBJETIVO

Implementar la comunicación y el monitoreo de adquisición de datos de los sensores mediante el internet, para recabar información del funcionamiento del animatronic.

OBJETIVO GENERAL

Diseñar un sistema de recepción y registro de datos mediante la implementación de arduino y el PLC en un animatronic, mostrando los datos obtenidos de los sensores en una página web, monitoreando los registros obtenidos durante el funcionamiento del animatronic.

OBJETIVO ESPECÍFICO

- Establecer la comunicación del PLC y el arduino mediante Ethernet.
- Diseño e implementación de un sistema de adquisición de datos a través del arduino, para la medición de parámetros.
- Establecer la comunicación con el sitio web
- Analizar el funcionamiento de los sensores
- Evaluar el desempeño del animatronic con los datos recopilados.

1.4 PROBLEMAS A RESOLVER

Resolver la comunicación del arduino con el PLC para la obtención de datos con el fin de mostrar los registros obtenidos a través de una PC conectada a internet.

Realizar la conexión de los dispositivos a través de internet

Monitorear el funcionamiento de los sensores

Con la implementación del proyecto deberá solucionar algunos problemas adicionales en el funcionamiento de los sensores y del animatronic.

1.5 ALCANCES Y LIMITACIONES

Los alcances con los que cuenta el proyecto son en el mejorar el uso y la adaptabilidad que se le dé al animatronic, debido a que se podrá recolectar información de los sensores, al realizar una acción durante el funcionamiento del mismo recopilando la información en un sitio web.

El proyecto cuenta con ciertas limitaciones principalmente con la transmisión y recepción de internet, ya que esta puede ser afectada por factores muchas veces fuera de nuestro control, como las interferencias en las redes de internet.

CAPÍTULO II

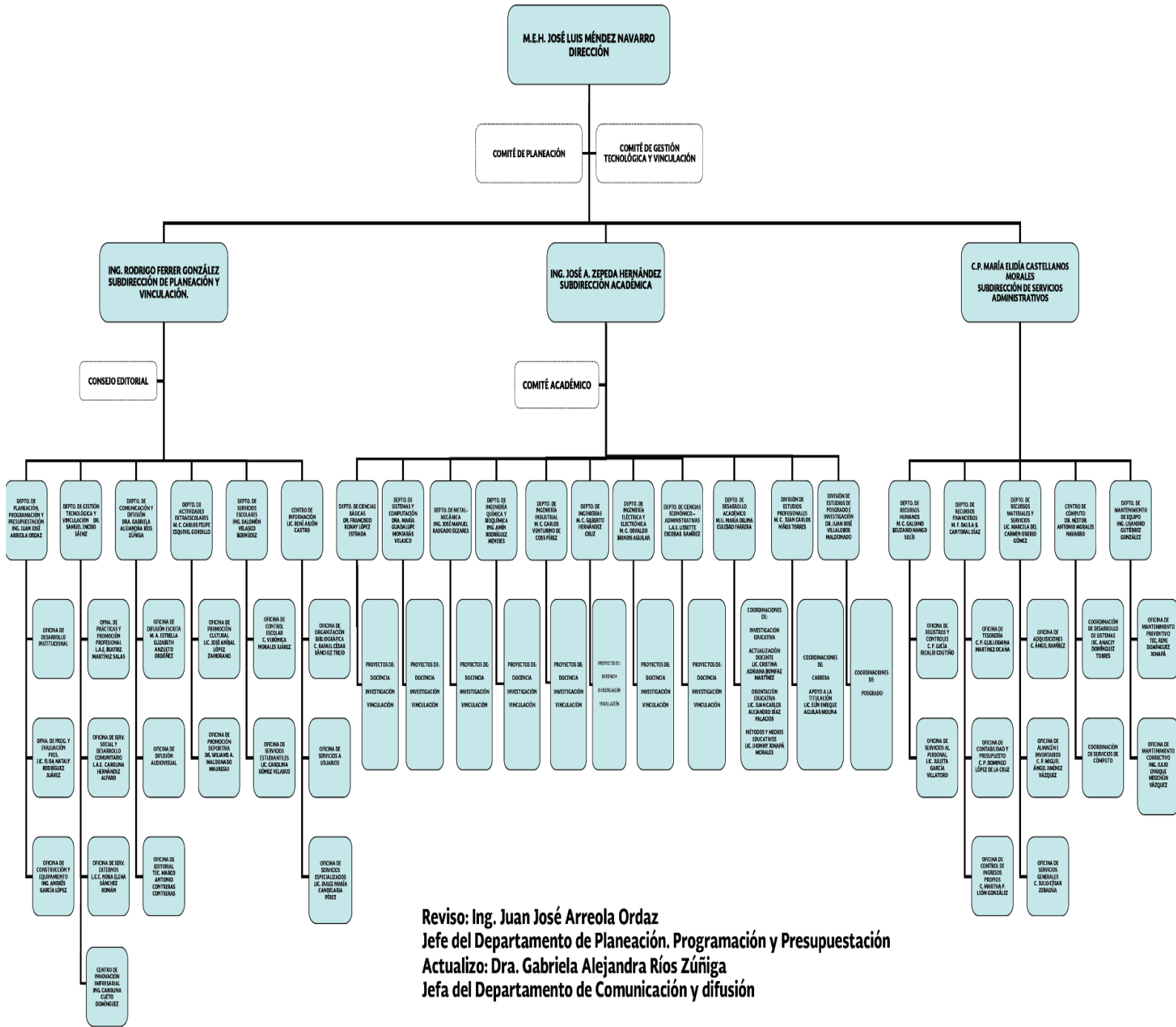
2.1 CARACTERIZACIÓN DEL ÁREA EN QUE SE PARTICIPÓ

En el área de Ingeniería Electrónica del Instituto Tecnológico de Tuxtla Gutiérrez; se pretende formar profesionales competentes en Ingeniería Electrónica con capacidad creativa, emprendedora, de análisis, liderazgo y capacidad de trabajo en equipo, que realicen actividades de diseño, innovación, adaptación y transferencia de tecnologías para resolver problemas del área de su competencia en forma competitiva atendiendo las necesidades del entorno globalizado, con una conciencia ética y social, asumiendo un compromiso con el desarrollo tecnológico y sustentable.

2.2 ANTECEDENTES DE LA EMPRESA

Nuestro Estado de Chiapas, fiel a sus deseos de progreso, más aún en la rama educativa media y superior, avalando con su potencial económica-social, logra sus más caros anhelos al contar ya con un Instituto Tecnológico para beneficio de la juventud estudiosa y acrecentar así la tecnología en todos sus aspectos. Es así como surge en 1972, el Instituto Tecnológico Regional de Tuxtla Gutiérrez por mandato del entonces Presidente de la Republica, Lic. Luis Echeverría Álvarez y colaboración ilimitada del Dr. Manuel Velasco Suárez, Gobernador del Estado, era el señor Ing. Víctor bravo Ahuja, secretario de Educación pública.

2.3 ORGANIGRAMA DEL INSTITUTO TECNOLÓGICO DE TUXTLA



Reviso: Ing. Juan José Arreola Ordaz
 Jefe del Departamento de Planeación, Programación y Presupuestación
 Actualizo: Dra. Gabriela Alejandra Ríos Zúñiga
 Jefa del Departamento de Comunicación y difusión

2.4 MISIÓN, VISIÓN Y VALORES

MISIÓN

Formar de manera integral profesional de excelencia en el campo de la ciencia y la Tecnología con actitud emprendedora, respeto al medio ambiente y apego a los valores éticos

VISIÓN

Ser una Institución de excelencia en la educación superior Tecnológica del Sureste, comprometida con el desarrollo socioeconómico sustentable de la región.

VALORES

El Ser Humano

El Espíritu de Servicio

El Liderazgo

El Trabajo en Equipo

La Calidad

El Alto Desempeño

Respeto al Medio Ambiente

2.5 DESCRIPCIÓN DEL ÁREA DONDE SE REALIZÓ EL PROYECTO

El proyecto de sistema para monitoreo y control de un animatronic vía wifi, se realizó en el área de ingeniería electrónica del Instituto Tecnológico de Tuxtla Gutiérrez, el cual cuenta con laboratorios que tiene equipos electrónicos a disposición del alumnado, tales como: osciloscopios, fuentes conmutadas y simétricas, variadores de frecuencias, procesadores lógicos programables, área de computo, área de proyecciones, entre otros. Además, cuenta con una caseta equipada con herramientas útiles para el desarrollo de proyectos, tales como: multímetros, cautines, fuentes variables, pinzas, destornilladores, entre otros. También se cuenta con el apoyo de catedráticos dispuestos a apoyar en lo que se requiera

CAPÍTULO III

3.1 FUNDAMENTO TEÓRICO

El alcance de los mecanismos robóticos o electrónicos actualmente están presente, en el desarrollo de la animatrónica que simula el aspecto y comportamiento de los seres vivos. Son creados para ser programados y controlados remotamente, para recrear movimientos ya sean sencillos o de gran complejidad.

3.2 AVANCES EN LA TECNOLOGIA

Los avances en la tecnología robótica han provocado una proliferación de aplicaciones más allá de los entornos de servicio tradicionales. Por ejemplo, los museos están utilizando más demostraciones de robótica social para atraer a sus clientes. El resultado es que estos robots sociales están siendo operados más allá del control directo y el soporte técnico de aquellos entrenados en tecnología robótica. Por lo tanto, es beneficioso incorporar el monitoreo de salud en tiempo real del sistema robótico para reconocer la necesidad de soporte técnico. A través de la incorporación de tecnología a menudo denominada Internet de las cosas, el soporte técnico puede monitorear las operaciones de forma remota para determinar las intervenciones necesarias. Este documento describe el desarrollo y despliegue de un buitre animatronic de 5 grados de libertad en una exposición en un museo distante. Además del seguimiento durante la exposición, Se utilizó tecnología para monitorear la frágil escultura durante el envío. La información se transmitió a los autores en tiempo real, lo que arrojó conclusiones valiosas sobre el bienestar remoto y el funcionamiento de la escultura.

Animatronics se ha empleado con éxito en películas, espectáculos y teatro durante más de 30 años, lo que ha dado como resultado un alto nivel de compromiso con el público. El resultado es que las personas tienen una gran expectativa de entretenimiento en otros lugares. En respuesta, los museos también buscaron emplear tecnología robótica para mejorar el compromiso de los clientes. Muchas de estas exhibiciones han sido en museos de tecnología o ciencia, como el Instituto Franklin de Filadelfia.

El personal en galerías de arte y museos rara vez tiene las capacidades técnicas para mantener estos sistemas. En respuesta, es beneficioso para estos sistemas hacer un monitoreo de salud personal para medir cuando el rendimiento varía. Idealmente, este desempeño puede ser monitoreado continuamente y las anomalías pueden ser reportadas a tecnólogos robóticos capaces para la determinación de las intervenciones apropiadas.

El monitoreo del estado de los sistemas animatronics en diversas configuraciones puede beneficiarse del empleo de tecnologías de Internet de las Cosas (IoT). IoT se

define en términos generales como “capacidades de identificación, detección, conexión en red y procesamiento que permitirán que (los objetos) se comuniquen entre sí y con otros dispositivos y servicios a través de Internet para lograr algún objetivo útil”. Se espera que decenas de miles de millones de dispositivos se conecten a Internet para 2020, lo que aporta beneficios a la industria como el pronóstico inteligente, la responsabilidad de fabricación y la adquisición de datos. Un buen ejemplo de IoT implementado e integrado con éxito es en medidores eléctricos inteligentes. De hecho, estos medidores se parecen mucho al sistema de monitoreo presentado en este documento, ya que informan el estado de una entidad varias veces por hora, lo que permite a los operadores de servicios públicos evitar la recuperación en persona de los datos. Más ampliamente, en los sistemas industriales, la conectividad por cable es la norma, aunque IoT presenta nuevas oportunidades para la transmisión inalámbrica de datos. Nuestra escultura demuestra la conectividad inalámbrica que no suele verse en la animatrónica profesional. [4]

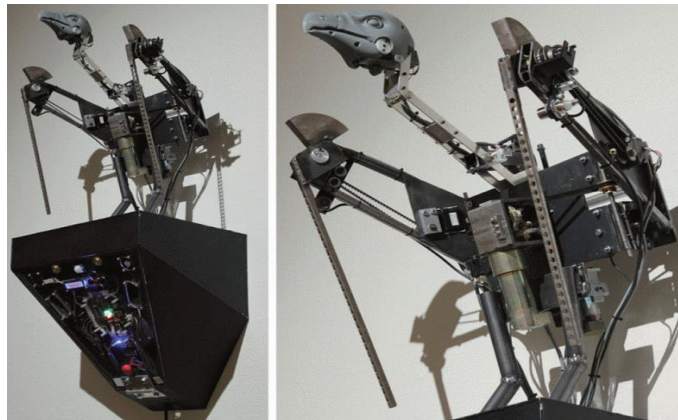


Fig. 1. Animatronic vulture in exhibition. Fuente: (Edward, 2018)

3.3 ANIMATRONIC

La animatrónica es la técnica capaz de dotar del aspecto y comportamiento de seres vivos a máquinas. Es decir, que es la parte de la ingeniería que se encarga de ‘humanizar’ a los robots. Los más básicos recrean simplemente el movimiento del cuerpo, pero los más complejos son capaces de simular el movimiento de cada músculo del cuerpo. La complejidad depende del uso al que se le vaya a dar a este, pues no siempre suele ser necesario que se recree por completo.

Históricamente se ha usado en la industria del espectáculo (parques temáticos, teatros, piezas cinematográficas y un largo etcétera). No obstante, también tiene otros usos que están llegando junto con la Inteligencia Artificial. Y justamente la unión de la IA junto con la animatrónica es lo que da el resultado de los androides, como se le conoce usualmente a los robots que imitan el comportamiento humano.

Una de las compañías más importantes detrás del desarrollo de la animatrónica es Disney. En los años 60 comenzó a desarrollar lo que patentó como Audio-animatronic, robots muy básicos que eran capaces de reproducir movimientos y sonidos ya previamente establecidos. Estaban destinados a ser una atracción de sus parques temáticos. [1]



Fig. 2. Animatronic de Abraham Lincoln en Disneyland

La principal utilidad de la animatrónica se da en el campo de la cinematografía y los efectos especiales, aunque también se emplea con frecuencia en los parques temáticos y en otras ramas de la industria del entretenimiento. Su principal ventaja respecto a las imágenes generadas por ordenador o la técnica de stop motion es que no consiste en una simulación de la realidad, sino que presenta ante la cámara auténticos objetos que se mueven en tiempo real. Con los años, la tecnología que da soporte a la animatrónica se ha ido haciendo más y más compleja, produciendo marionetas cada vez más realistas y, aparentemente, vivas. [2]

Para el diseño de un animatrónico es necesario el uso de técnicas y materiales que en su totalidad creen un robot que emule vida. Por lo tanto, es necesaria una investigación previa del personaje animado que se pretende crear. El análisis debe contemplar los movimientos del personaje, la movilidad de los músculos y la anatomía.

Una vez examinado el personaje se procede a las fases técnicas de creación: el elemento mecánico, el eléctrico y el de control.

- La mecánica se encarga de crear la estructura, y además, los movimientos que le corresponden.

- La electrónica dota de energía el animatrónico y define los elementos de entrada y el comportamiento de los de salida.
- El control otorga los comandos necesarios que se deben usar para poder manejar el personaje en la vida real.

Para la creación de un muñeco de estas características se han de tener en cuenta los puntos vitales que lo pueden humanizar. Estos elementos son: la boca, los labios, los ojos, el cuello, etc., en definitiva, el movimiento muscular. Los movimientos específicos se consiguen mediante motores eléctricos, cilindros neumáticos o hidráulicos y mecanismos controlados por cable. El tipo de elemento a utilizar se decide en función de los parámetros del personaje, los movimientos concretos que se requieren y las limitaciones del proyecto.

Finalmente, para que un animatrónico tenga una apariencia física real se utilizan materiales que simulen la piel de un ser vivo. De esta manera queda cubierto el esqueleto mecánico y eléctrico que se encarga de recrear los movimientos. Los materiales más usados son la silicona, el acrílico y la resina de nailon.[3]

3.4 COMPONENTES DE UN AUTOMATISMO

Los automatismos están compuestos por tres partes fundamentales, como son la obtención de señales mediante sensores, el procesamiento de dichas señales por lógicas de control y la ejecución de las respuestas mediante los actuadores (ver la fig.2). [5]

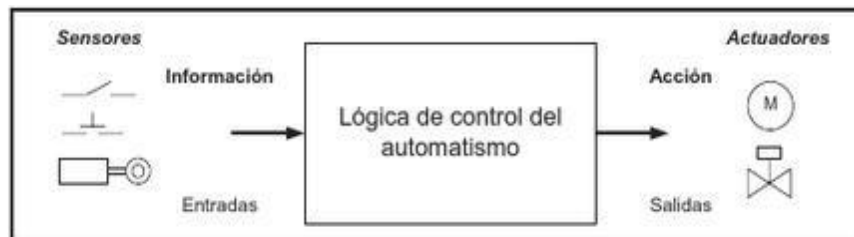


Fig. 3. Componentes de un automatismo

PLC

Un PLC permite controlar un proceso o realizar una secuencia de acciones de manera automática a partir de un programa definido por el usuario. Esto lo realiza ejecutando en forma cíclica una secuencia de instrucciones que, a partir de la información que llega a sus entradas desde los sensores, deciden cuando conmutar sus salidas, donde se encuentran conectados los actuadores. En este capítulo se describirán las partes que componen un PLC y sus principales características.

ESTRUCTURA INTERNA DEL PLC

El diagrama en bloques de la Fig.2 muestra la estructura interna de un controlador lógico programable (PLC).

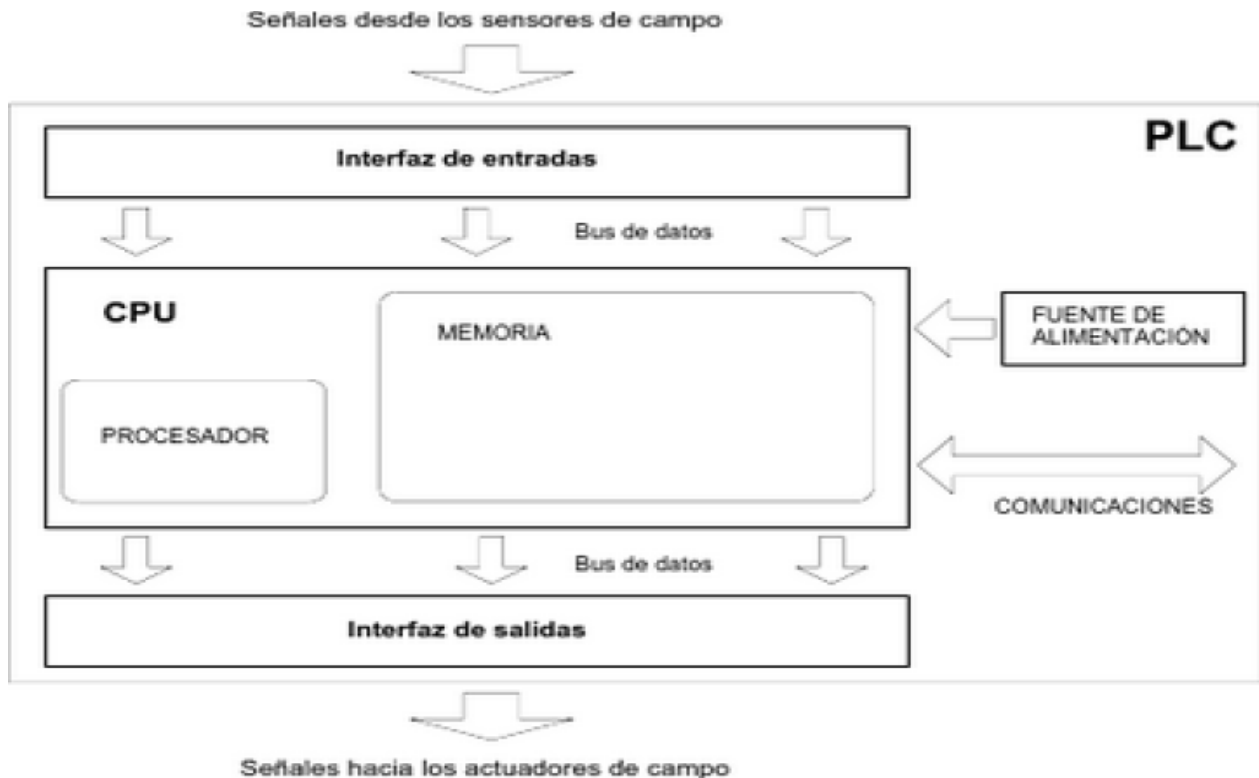


Fig. 3. Bloques internos que componen un PLC. Fuente:(Daneri,2008)

Sus partes fundamentales son la unidad central de procesos o CPU, y las interfaces de entrada y salida. La CPU es el cerebro del PLC y está formada por el procesador y la memoria. El procesador se encarga de ejecutar el programa escrito por el usuario que se encuentra almacenado en la memoria. Además, el procesador se comunica con el exterior mediante sus puertos de comunicación y realiza funciones de autodiagnóstico.

El PLC realiza las acciones de control mediante sus entradas y salidas. Las entradas vigilan las señales de los dispositivos de campo, por ejemplo, sensores e interruptores, mientras que las salidas comandan las bombas, motores u otros actuadores del proceso.

Los canales de entrada o salida se pueden clasificar de la siguiente manera:

- Digitales. También llamada on/off o discretas, pueden tomar solo dos estados, 0 y 1.
- Analógicos. Pueden tomar una cantidad de estados dentro de un cierto rango de tensión o corriente, por ejemplo, 4 a 20 mA, 0 a 20mA, 0a 10 v y -10 a 10 v.

- Especiales. Son variantes de los anteriores que se emplean en aplicaciones específicas, como por ejemplo el conteo de alta velocidad, etc.

ACTUADORES: Cilindros de simple y doble efecto. Los cilindros son los componentes neumáticos más difundidos, quienes, mediante el uso del aire comprimido, generan un movimiento rectilíneo de avance y de retroceso de un vástago. Se dividen en dos grandes grupos, los cilindros de simple efecto y los de doble efecto.

En los de simple efecto el aire a presión ingresa por la cámara trasera y desplaza al vástago. El retroceso se produce mecánicamente por la acción de un resorte.

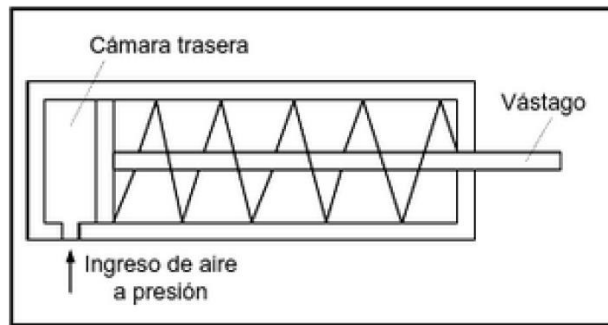


Fig. 4. Cilindro de simple efecto

En los cilindros de doble efecto el avance se realiza de la misma forma que en los de simple efecto, pero para el retroceso se debe hacer que el aire a presión penetre por el orificio de la cámara delantera. Cuando el vástago retrocede, el aire de la cámara trasera se libera al exterior a través del conducto correspondiente. Lo mismo sucede con el aire contenido en la cámara delantera cuando el cilindro avanza.

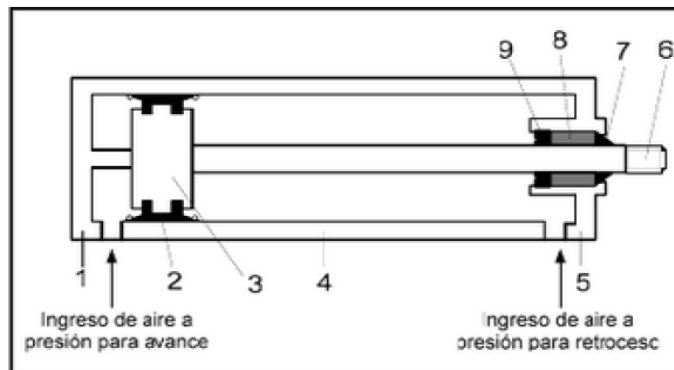


Fig. 5. Cilindro de doble efecto

Las partes fundamentales de un cilindro son la tapa trasera 1, el tubo o la camisa 4, el pistón 3, el vástago 6 y la tapa delantera 5. Para lograr la estanqueidad se necesita que tanto las tapas, como el pistón y el vástago, posean las correspondientes juntas de cierre. Por lo tanto, en las tapas se montan juntas

estáticas 8, en el pistón juntas estáticas y dinámicas 2, y en el vástago la dinámica 9. [5]

DETECTORES: Los detectores tienen la función de captar informaciones y de transmitir señales procesables a las unidades de evaluación. En numerosas aplicaciones se utilizan detectores de diversas formas y modos de funcionamiento. Considerando su gran variedad, es importante clasificarlos sistemáticamente. Los detectores pueden clasificarse de acuerdo con los siguientes criterios:

- Modo de funcionamiento (óptico, inductivo, mecánico, por fluidos, etc.)
- Magnitud de medición (recorrido, presión, distancia, temperatura, valor pH, intensidad de luz, presencia de piezas, etc.)
- Señal de salida (analógica, digital, binaria, etc.)

Los detectores más difundidos en la automatización industrial son los así llamados detectores de posición, con los que se comprueba la presencia (o la aproximación) de una pieza.

Detectores de posición: Los detectores de posición conmutan sin establecer contacto y, por lo tanto, sin que sea necesaria la presencia de una fuerza mecánica externa. Por ello tienen una larga duración y son muy fiables. Se puede distinguir entre los siguientes tipos:

- Detectores con contacto de conmutación mecánico
 - Contacto Reed

Los contactos Reed: son detectores de posición de accionamiento magnético. Estos detectores tienen dos lengüetas de contacto que se encuentran en un tubo de vidrio lleno de gas inerte. Por efecto de un imán se cierra el contacto entre las dos lengüetas, de modo que puede fluir corriente eléctrica.

Tratándose de contactos Reed normalmente cerrados, las lengüetas están pretensadas mediante un pequeño imán. Esta precarga se supera mediante un imán mucho más potente.

Los contactos Reed tienen una gran duración y su tiempo de respuesta es muy corto (aprox. 0,2 ms). Además, no precisan mantenimiento, aunque no deben utilizarse en zonas expuestas a campos magnéticos fuertes (por ejemplo, en las cercanías de máquinas de soldadura por resistencia o equipos de tomografía computarizada). [6]

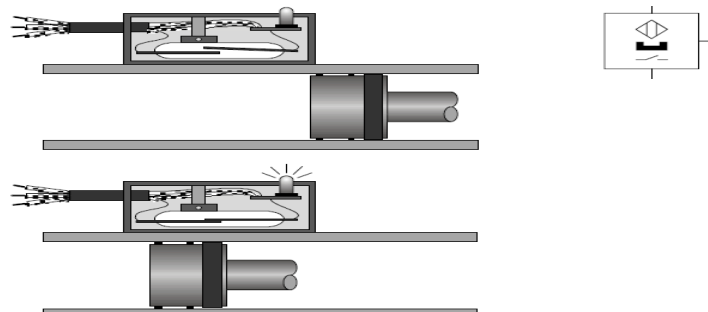


Fig. 6. contacto reed:diagrama y simbolo

3.5 IOT

El Internet de las cosas comenzó a imponerse en el sector empresarial, conocido como la Nueva Revolución Industrial y contribuyendo a la transformación de las prácticas comerciales en diferentes sectores. Hay un impacto positivo significativo de las iniciativas de transformación digital en los resultados y funciones del negocio, como el aumento de los ingresos, la mejora de los servicios y el lanzamiento más rápido de productos. La adopción de la tecnología de Internet puede garantizar que los negocios sean sostenibles y exitosos en esta etapa digital.

Las cosas de Internet "Internet Objects" tienen como objetivo hacer que todo lo que nos rodea se conecte a Internet para compartir datos asociados con él, así como el uso de inteligencia artificial para tratar con la gran cantidad de datos y explorar nuevas formas de desarrollar negocios y productos. Por ejemplo, los sensores de la planta que miden la temperatura, las emisiones gaseosas y la presión se pueden agregar a las plantas, y se pueden compartir con dispositivos que las registran y analizan grandes cantidades de ellas. Para observar cualquier fluctuación o fluctuación en el trabajo, para tomar acción. [7]

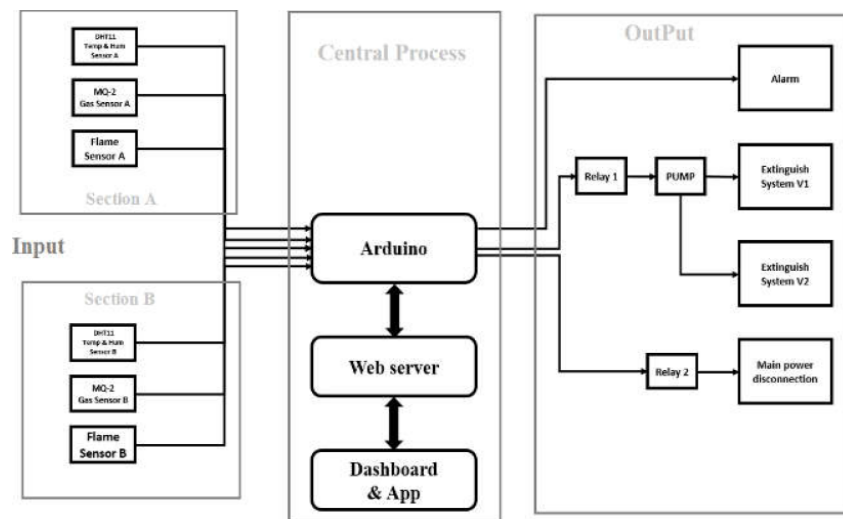


Fig. 7. block diagram

La idea principal de la IoT es integrar objetos en una red de comunicación. En otras palabras, representa un nuevo enfoque donde todas las cosas forman parte de Internet, trabajando entre ellas en tiempo real. Técnicamente, se basa en la integración de sensores, dispositivos y elementos del hogar que se conectan a Internet a través de redes cableadas o inalámbricas. Debido al despliegue mundial de Internet, la adopción de esta tecnología es factible y el costo de la implementación será económico.

Permitirá que los sensores se integren en hogares, lugares de trabajo, procesos de automatización, etc. De esta manera, cada objeto se puede conectar a un entorno

web (en este trabajo, la página web de Xively) para almacenar toda su información y mostrarla al trabajo real (si es conveniente). Muestra los datos recogidos de los sensores directamente a internet. Permite filtrar y trazar los datos para encontrar posibles tendencias o situaciones críticas. Los datos se pueden verificar en cualquier parte del mundo independientemente de la ubicación de los sensores.

La tendencia del mercado es que la información esté disponible independientemente del lugar o la ubicación geográfica. Por esta razón, en la actualidad, internet se utiliza para generar una interacción en tiempo real entre dispositivos que no será posible con otros medios. Simplemente, toda la información recopilada de los sensores debe estar disponible en la nube para ser administrada y controlada. Por lo tanto, existe un punto central de administración donde se almacena la información de los sensores distribuidos remotamente.

Además, los procesos industriales hacen necesario implementar sistemas de comunicación inalámbricos (debido al entorno hostil y al difícil acceso a los lugares) para transmitir las señales generadas por los sensores que forman el circuito de control. [8]

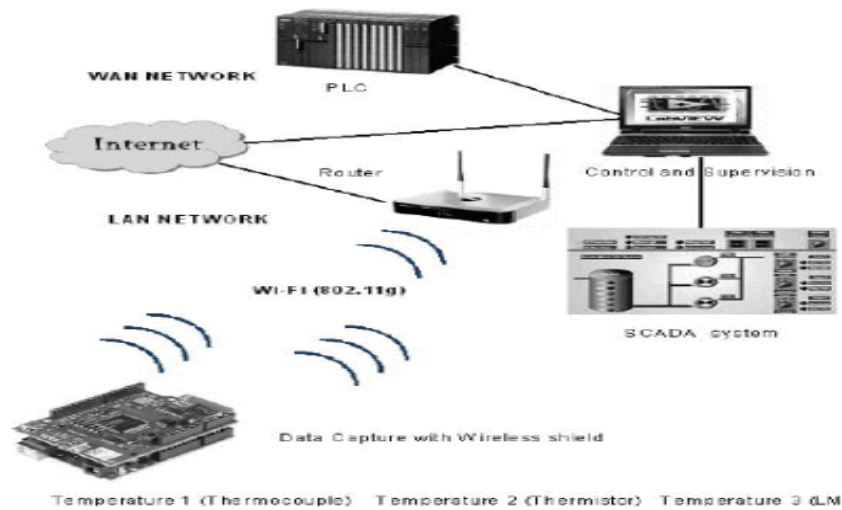


Fig. 8. Real Topology

3.6 ARDUINO

Arduino es una plataforma electrónica de código abierto basada en hardware y software fáciles de usar. Se creó para artistas, diseñadores, aficionados y cualquiera interesado en crear entornos u objetos interactivos. Arduino nació en el Instituto de Diseño de Interacción Ivrea como una herramienta fácil para la creación rápida de prototipos, dirigida a estudiantes sin experiencia en electrónica y programación. Tan pronto como llegó a una comunidad más amplia, la placa Arduino comenzó a cambiar para adaptarse a las nuevas necesidades y desafíos, diferenciando su oferta de tablas simples de 8 bits a productos para aplicaciones IoT, entornos integrados.

Todas las placas Arduino son completamente de código abierto, lo que permite a los usuarios construirlas de forma independiente y, eventualmente, adaptarlas a sus necesidades particulares. El software también es de código abierto y está creciendo a través de las contribuciones de los usuarios de todo el mundo.

El software Arduino es fácil de usar para principiantes, pero lo suficientemente flexible para usuarios avanzados. Se ejecuta en Mac, Windows y Linux. Los profesores y los estudiantes lo utilizan para construir instrumentos científicos de bajo costo, para probar los principios de química y física, o para comenzar con la programación y la robótica. El lenguaje se puede expandir a través de las bibliotecas de C ++, y las personas que deseen comprender los detalles técnicos pueden dar el salto de Arduino al lenguaje de programación AVR C en el que se basa. De manera similar, puede agregar código AVR-C directamente en sus programas Arduino si lo desea. [9]

ETHERNET SHIELD

Un shield es una placa de circuito modular que se montan unas encima de otras para dar funcionalidad extra a un Arduino. Esta Shields son apilables. Se basa en el chip de ethernet Wiznet W5100. El chip Wiznet W5100 ofrece una red (IP) capaz de usar TCP y UDP. Soporta hasta cuatro conexiones de socket simultáneas. Utilice la biblioteca de Ethernet para escribir bocetos que se conectan a Internet a través del escudo. El escudo de Ethernet se conecta a una placa Arduino usando largas cabeceras wire-wrap que se extienden a través del escudo. Esto mantiene la disposición de las clavijas intacto y permite que otro escudo pueda ser apilado en la parte superior.

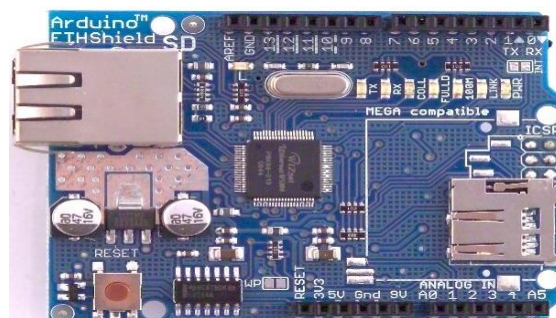


Fig. 9. Arduino Ethernet Shield. Fuente: (Arduino, Arduino Ethernet Shield, 2015)

El módulo Arduino Shield Ethernet tiene una conexión RJ-45 estándar, con un transformador de línea integrado. El botón de reset en la shield resetea ambos, el W5100 y la placa Arduino.

El módulo Arduino Shield Ethernet contiene una serie de LEDs informativos:

- **PWR**: indica que la placa y el módulo están encendidos.
- **LINK**: indica la presencia de un enlace de red y parpadea cuando el módulo transmite o recibe datos.
- **FULLD**: indica que la conexión de red es full duplex
- **100M**: indica la presencia de una conexión de red a Mb/s 100 (en contraposición a 10 Mb/s)
- **RX**: Parpadea cuando el escudo recibe datos
- **TX**: parpadea cuando el escudo envía datos
- **COLL**: parpadea cuando se detectan colisiones de red

El propio chip W5100 incluye el manejo de tarjetas SD. [9]

CHIP WIZNET W5100

Es un chip controlador de Ethernet, el que incluye la pila TCP/IP implementada por hardware, su principal campo de aplicación son sistemas embebidos con conectividad hacia internet. [10]



Fig. 10. CHIP WIZNET W5100. Fuente: (WIZnet Co, 2008)

Características:

- Implementa protocolos TCP, UDP, ICMP, IPV4, ARP, IGMPv2, PPPoE, Ethernet.
- Soporta 8 sockets independientes simultáneos
- 10/BaseT/100BaseTX Ethernet PHY
- Auto negociación (Full-duplex/Half-duplex)
- Ancho de bus 16 / 8 bit
- Memoria interna de 128 kbytes para comunicación de datos Tx/Rx
- Modo SPI (Serial Peripheral Interface)
- Opera a 3.3V

COMUNICACIÓN SPI

SPI (Serial Peripheral Interface) es una conexión de tres hilos síncrona, se usa para comunicar micro-controladores con periféricos u otros micro-controladores (López Pérez, S. F.). En la figura 3 se representa las líneas de comunicación, SPI es un estándar ampliamente utilizado, soporta los modos master y slave.

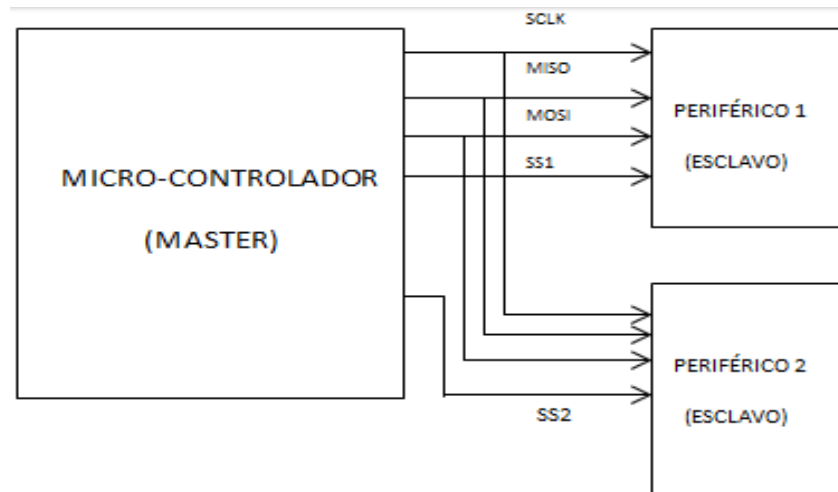


Fig. 11.COMUNICACIÓN SPI. Fuente: (López Pérez, S. F.)

Soporta paquetes de 8 bits, y la comunicación es full dúplex, dos hilos se utilizan para datos y la tercera línea es para el reloj.

ESPECIFICACIONES DEL BUS

- SCLK es el reloj generada por el maestro, sincroniza la transferencia de datos.
- MOSI (Master Out Slave In) comunica datos del maestro al esclavo.
- MISO (Master In Slave Out) comunica datos del esclavo al maestro.
- CS o SS (Chip Select o Slave Select) selecciona esclavo. [11]

ENTORNO DE DESARROLLO ARDUINO

El entorno de desarrollo en Arduino (IDE) es el encargado de la gestión de la conexión entre el PC y el hardware de Arduino con el fin de establecer una comunicación entre ellos por medio de la carga de programas. Como podemos ver en la figura 18, el IDE de Arduino se compone de: [12]

Un editor de texto: donde escribir el código del programa.

Un área de mensajes: a través del cual el usuario tendrá constancia en todo momento de los procesos que se encuentren en ejecución, errores en código, problemas de comunicación, etc.

Una consola de texto: mediante la que podremos comunicarnos con el hardware Arduino y viceversa.

Una barra de herramientas: donde podremos acceder a una serie de menús y a los botones con acceso directo a las principales funcionalidades de Arduino.

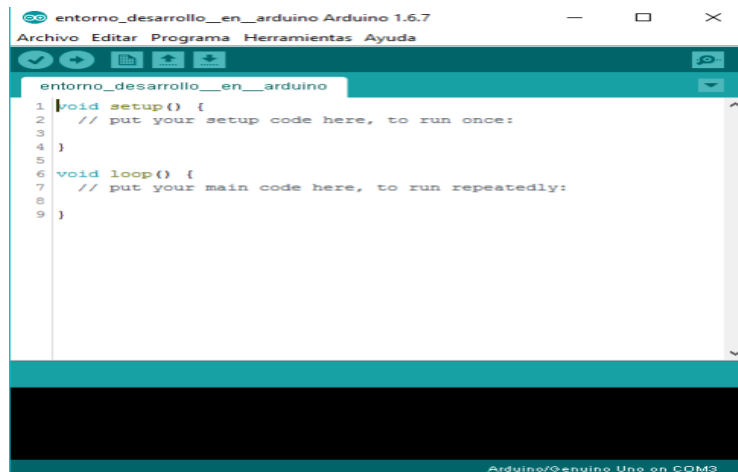


Fig. 12. Entorno de desarrollo en Arduino

ESTRUCTURA GENERAL DE UN SKETCH

Un programa diseñado para ejecutarse sobre un Arduino, un sketch siempre se compone de tres secciones:

La sección de declaraciones de variables globales: Ubicada directamente al principio del código.

La sección llamada Void Setup (): Delimitada por llaves de apertura y cierre, donde las instrucciones escritas dentro de esta sección se ejecutan una única vez, en el momento de encender la placa Arduino.

La sección llamada Void Loop (): Delimitada por llaves de apertura y cierre. Las instrucciones escritas dentro se ejecutan infinitas veces hasta que la placa se apague (o se resetee). Es en este punto donde está el programa en sí y que funciona continuamente.

Las instrucciones escritas dentro de la sección "void setup ()" se ejecutan una única vez, en el momento de encender (o resetear) la placa Arduino.

Las instrucciones escritas dentro de la sección "void loop ()" se ejecutan justo después de las de la sección "void setup ()" infinitas veces hasta que la placa se apague (o se resetee). Es decir, el contenido de "void loop ()" se ejecuta desde la 1ª instrucción hasta la última, para seguidamente volver a ejecutarse desde la 1ª instrucción hasta la última, para seguidamente ejecutarse desde la 1ª instrucción hasta la última, y así una y otra vez.

Por tanto, las instrucciones escritas en la sección “void setup ()” normalmente sirven para realizar ciertas preconfiguraciones iniciales y las instrucciones del interior de “void loop ()”. [13]

3.7 SERVIDOR WEB

Un servidor, como la misma palabra indica, es un ordenador o máquina informática que está al “servicio” de otras máquinas, ordenadores o personas llamadas clientes y que les suministran a estos, todo tipo de información.

Por tanto, un servidor en informática será un ordenador u otro tipo de dispositivo que suministra una información requerida por unos clientes (que pueden ser personas, o también pueden ser otros dispositivos como ordenadores, móviles, impresoras, etc.).

Básicamente tendremos el siguiente esquema general, en el denominado esquema “cliente-servidor” que es uno de los más usados ya que en él se basa gran parte de internet.

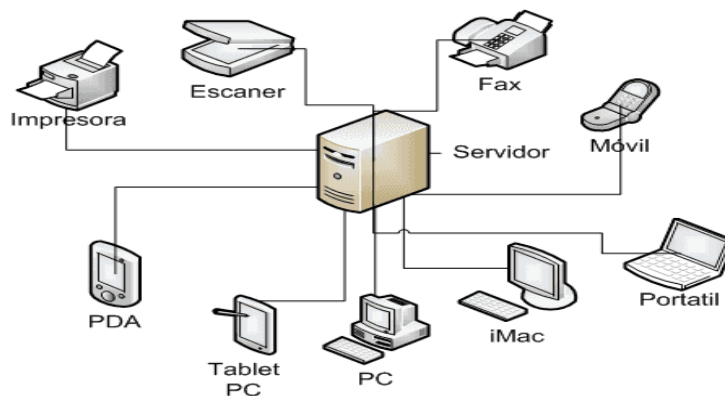


Fig. 13.Servidor

Como vemos, tenemos una máquina servidora que se comunica con variados clientes, todos demandando algún tipo de información. Esta información puede ser desde archivos de texto, video, audio, imágenes, emails, aplicaciones, programas, consultas a base de datos, etc.

Por regla general, las máquinas servidoras suelen ser algo más potentes que un ordenador normal. Sobre todo, suelen tener más capacidad tanto de almacenamiento de información como de memoria principal, ya que tienen que dar servicio a muchos clientes. Pero como todo, también depende de las necesidades, ya que podemos tener un servidor de menores prestaciones si vamos a tener pocos clientes conectados, o si los servicios que queramos en el servidor no requieren una gran capacidad servidora. A modo de ejemplo, podríamos hacer funcionar un ordenador en nuestra casa como si fuera un servidor, aunque esto no es lo más habitual. Por general, los servidores suelen estar situados en centros de datos de empresas (edificios con grandes salas dedicadas a alojar a los servidores).

DNS: son las siglas de Domain Name System. Es un sistema por el que se asocia una información con un nombre de dominio. El ejemplo más claro es cuando introducimos una ruta url en nuestro navegador de internet del tipo `http://www.aprenderaprogramar.com`. Una vez hemos introducido esta ruta, dicha información es enviada a un servidor DNS que lo que hace es determinar en qué lugar se encuentra esa página web alojada y nos conecta con ella.

WEB: el término web va asociado a internet, donde los usuarios utilizan sus navegadores web para visitar sitios web, que básicamente se componen de páginas web donde los usuarios pueden acceder a informaciones con texto, videos, imágenes, etc y navegan a través de enlaces o hipervínculos a otras webs. [14]

3.8 SUBLIME TEXT

Sublime Text es un editor de código multiplataforma, un editor de texto avanzado especialmente diseñado para desarrolladores que realmente sale de lo común y se destaca por sus funcionalidades e interfaz de usuario.

Sublime Text tiene resaltado a color de sintaxis para facilitar la lectura de código, así como numerado de línea. También muestra un pequeño índice gráfico a la derecha, por el cual podemos desplazarnos rápidamente a golpe de clic del ratón.

Autocompletado. Esta funcionalidad nos ayudará a recordar cómo se escriben ciertos comandos, funciones o textos del lenguaje o marcado en el que nos encontramos.[15]



Fig. 14.Sublime Text

3.9 HTML

HTML (hypertext mark-up language). Se trata de un lenguaje de marcas (se utiliza insertando marcas en el interior del texto) que nos permite representar de forma rica el contenido y también referenciar otros recursos (imágenes, etc.), enlaces a otros documentos (la característica más destacada del WWW), mostrar formularios para posteriormente procesarlos, etc.

El lenguaje HTML actualmente se encuentra en la versión 4.01 y empieza a proporcionar funcionalidades más avanzadas para crear páginas más ricas en contenido. Además, se ha definido una especificación compatible con HTML, el

XHTML (extensible hypertext markup language) que se suele definir como una versión XML validable de HTML, proporcionándonos un XML Schema contra el que validar el documento para comprobar si está bien formado, etc.

La estructura de un documento HTML

Un documento HTML comienza con la etiqueta **<html>**, y termina con **</html>**.

Dentro del documento hay dos zonas principales: el encabezamiento, delimitado por las marcas **<HEAD>** y **</HEAD>**, que sirve para definir algunos valores válidos para todo el documento, y el cuerpo, delimitado por las etiquetas **<BODY>** y **</BODY>**, donde reside la información del documento.

El elemento **<TITLE>** contenido dentro del encabezamiento permite especificar el título de un documento HTML. Este título no forma parte del documento en sí pues no se ve en la pantalla principal, sino que sirve como título de la ventana del programa que la muestra. Existen muchos otros elementos que se engloban dentro del encabezamiento, pero para la estructura básica del lenguaje HTML en su nivel básico no son necesarios.

La estructura de un documento HTML es la siguiente:

```
<HTML>
<HEAD>
<TITLE>Título de la página</TITLE>
</HEAD>
<BODY>
[Aquí se sitúan otras etiquetas que hacen posible visualizar la página]
</BODY>
</HTML>
```

CSS: CSS es un lenguaje que describe el estilo de un documento HTML.

CSS describe cómo se deben mostrar los elementos HTML.

¿Qué es CSS?

CSS significa: **C**ascading **S**tyle **S**heets

- CSS describe cómo se mostrarán los elementos HTML en la pantalla, en papel o en otros medios
- CSS ahorra mucho trabajo. Puede controlar el diseño de varias páginas web a la vez.
- Las hojas de estilo externas se almacenan en archivos CSS

Hoja de estilo externa

Con una hoja de estilo externa, puede cambiar el aspecto de un sitio web completo cambiando solo un archivo. Cada página debe incluir una referencia al archivo de la hoja de estilo externa dentro del elemento <link>. El elemento <link> va dentro de la sección <head>.

PHP: PHP es un lenguaje de scripts de servidor y una poderosa herramienta para crear páginas web dinámicas e interactivas. es una alternativa ampliamente utilizada, gratuita y eficiente para competidores como el ASP de Microsoft.

¿Qué es PHP?

- PHP es un acrónimo de "PHP: preprocesador de hipertexto"
- PHP es un lenguaje de scripting de código abierto ampliamente utilizado
- Los scripts PHP se ejecutan en el servidor.
- PHP es gratuito para descargar y usar

¿Qué es un archivo PHP?

- Los archivos PHP pueden contener texto, HTML, CSS, JavaScript y código PHP
- El código PHP se ejecuta en el servidor, y el resultado se devuelve al navegador como HTML plano
- Los archivos PHP tienen extensión ".php"

¿Qué puede hacer PHP?

- PHP puede generar contenido de página dinámico
- PHP puede crear, abrir, leer, escribir, eliminar y cerrar archivos en el servidor
- PHP puede recopilar datos de formulario
- PHP puede enviar y recibir cookies
- PHP puede agregar, eliminar, modificar datos en su base de datos
- PHP puede ser usado para controlar el acceso de usuarios
- PHP puede cifrar datos

Con PHP no estás limitado a generar HTML. Puede generar imágenes, archivos PDF e incluso películas Flash. También puede generar cualquier texto, como XHTML y XML.

SQL: SQL es un lenguaje estándar para almacenar, manipular y recuperar datos en bases de datos.

Tipos de datos SQL: Tipos de datos y rangos para Microsoft Access, MySQL y SQL Server.

¿Qué es SQL?

- SQL significa lenguaje de consulta estructurado
- SQL le permite acceder y manipular bases de datos
- SQL se convirtió en un estándar del American National Standards Institute (ANSI) en 1986 y de la Organización Internacional de Normalización (ISO) en 1987. [16]

¿Qué puede hacer SQL?

- SQL puede ejecutar consultas contra una base de datos
- SQL puede recuperar datos de una base de datos
- SQL puede insertar registros en una base de datos
- SQL puede actualizar registros en una base de datos
- SQL puede eliminar registros de una base de datos
- SQL puede crear nuevas bases de datos
- SQL puede crear nuevas tablas en una base de datos
- SQL puede crear procedimientos almacenados en una base de datos
- SQL puede crear vistas en una base de datos
- SQL puede establecer permisos en tablas, procedimientos y vistas

3.10 XAMPP

Es el entorno de desarrollo PHP más popular, XAMPP es una distribución Apache completamente gratuita y fácil de instalar que contiene MariaDB, PHP y Perl. El paquete de código abierto XAMPP se ha configurado para que sea increíblemente fácil de instalar y usar.

XAMPP es una compilación de software libre (similar a una distribución de Linux). Es gratuita y puede ser copiada libremente de acuerdo a la licencia GNU GPL. Únicamente la compilación de XAMPP está publicada bajo la licencia GPL. Cada uno de los componentes incluidos tiene su propia licencia y deberías consultarlas para conocer qué es posible y que no. En el caso de uso comercial deberás consultar las licencias individuales, en particular MySQL. Desde el punto de vista de XAMPP como compilación, el uso comercial es gratuito.

XAMPP proporciona un entorno de desarrollo local ideal, pero no está diseñado para implementaciones de producción. Queremos que el alojamiento de aplicaciones PHP creadas con XAMPP sea lo más fácil posible. Comenzamos por ofrecer tutoriales detallados sobre cómo ejecutar XAMPP en la nube con Microsoft Azure , Google Cloud Platform , 1&1 y Amazon Web Services. [17]



Fig. 15.XAMPP

CAPÍTULO IV

4.1 PROCEDIMIENTO Y ACTIVIDADES REALIZADAS

Las actividades realizadas en el proyecto, son realizadas en acorde a lo que se requiere cumpliendo los procesos establecidos en el cronograma de actividades, y organizando cada proceso obtenido durante su elaboración.

Todo el procedimiento realizado se puede ver en los siguientes subtemas, en las que se describen cada paso realizado.

4.2 CRONOGRAMA DE ACTIVIDADES

Actividad	Semana															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Búsqueda Bibliográfica	x	x	x	x	x											
Diseño de la página web		x	x	x	x											
Desarrollo de la programación de arduino				x	x	x	x									
Diseño e implementación de la comunicación wifi						x	x	x	x	x						
Conexión del PLC y el arduino							x	x	x	x	x	x	x			
Pruebas del sistema y monitoreo													x	x	x	x

Investigación bibliográfica

En el diseño e implementación del monitoreo de un Animatronic, donde se dio a la tarea básica de la investigación recopilando información sobre la creación y diseño de una página web.

Estructura de conexión



4.3 REALIZACIÓN DE UNA PÁGINA WEB

Primer paso

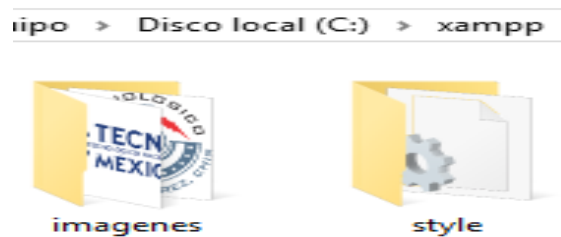
Descargar el editor de texto desde el sitio <https://www.sublimetext.com/>

Instalar en la maquina e instalar las librerías más óptimas para HTML, CSS y JavaScript.

- Emmet. Automatiza la creación de bloques de HTML utilizando abreviaciones.
- HTML-CSS-JS Prettify. Maqueta el código.
- ColorPicker. Se ahorra mucho tiempo al no tener que cambiar de ventana para buscar el código del color por otros medios.

Segundo paso

Se crea una carpeta, dentro del mismo se crea otras dos carpetas, con el nombre de style y el otro de imágenes.



En la carpeta se crea un documento con el nombre de index.php, el cual contendrá la programación principal para la página web, el archivo se muestra a continuación.

m.php (garduino) - Sublime Text (UNREGISTERED)

v Goto Tools Project Preferences Help

Guardar como

« xampp » htdocs » ejemplo

Organizar Nueva carpeta

Nombre	Fecha de modifica...	Tipo
imagenes	09/10/2018 07:19 ...	Carpeta de archivos
style	09/10/2018 07:39 ...	Carpeta de archivos
ejem	24/10/2018 01:21 ...	Archivo PHP

Nombre: ejem

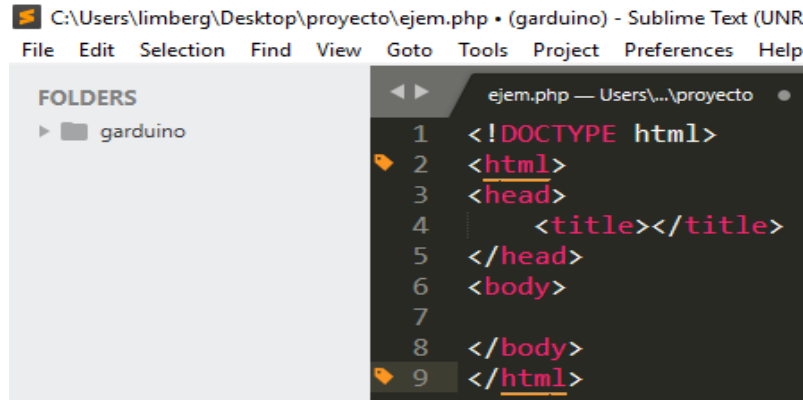
Tipo: PHP (*.php;*.php3;*.php4;*.php5;*.php7;*.phps;*.phpt;*.phtml)

Guardar Cancelar

equipo » Disco local (C:) » xampp » htdocs » ittg

Nombre	Fecha de modifica...	Tipo
imagenes	06/01/2019 02:35 ...	Carpeta de archivos
style	09/10/2018 07:39 ...	Carpeta de archivos
index	19/12/2018 05:49 a...	Archivo PHP

Cuando se crea el archivo index.php, se realiza el código para la creación de la página web que contendrá las siguientes líneas de códigos como se muestra a continuación. Y servirán para empezar a organizar el contenido principal del programa.



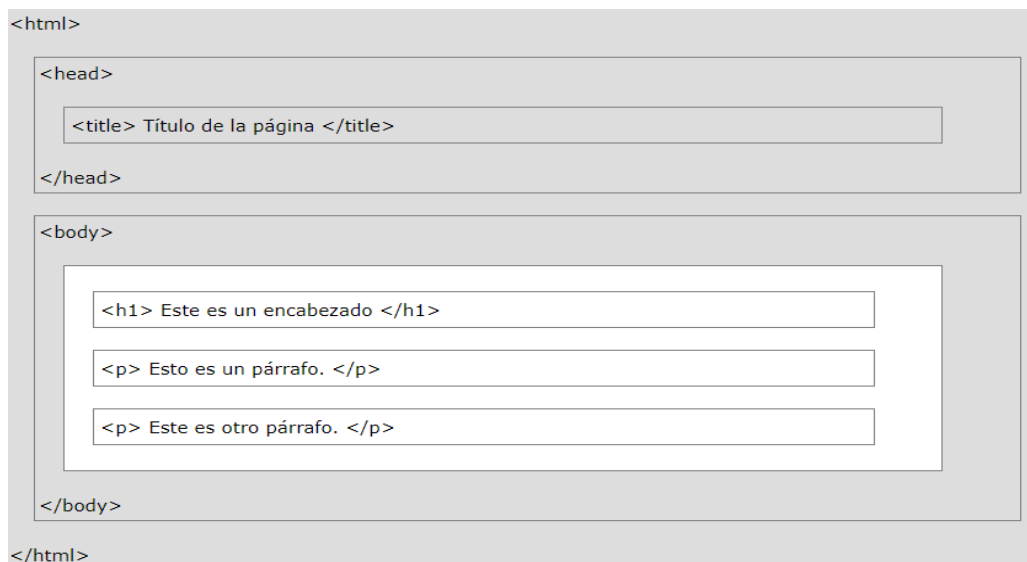
The screenshot shows a code editor window titled 'ejem.php' with the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5 </head>
6 <body>
7
8 </body>
9 </html>
```

El cual son código de html para crear páginas web.

- La declaración `<!DOCTYPE html>` define este documento como HTML5
- El elemento `<html>` es el elemento raíz de una página HTML
- El elemento `<head>` contiene información sobre la página.
- El elemento `<title>` especifica el título de la página.
- El elemento `<body>` contiene el contenido visible de la página.
- El elemento `<h1>` especifica el encabezado.
- El elemento `<p>` define los párrafos.

A continuación, se muestra una estructura de una página HTML:

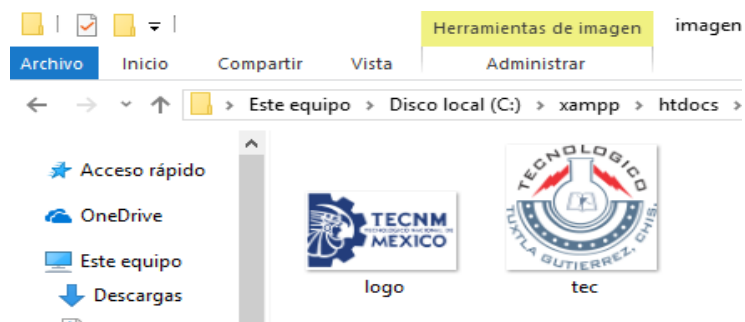


Dentro de la estructura de código, se ingresan los datos necesarios para organizar los datos necesarios que se requiera la página web, como se muestra a continuación.

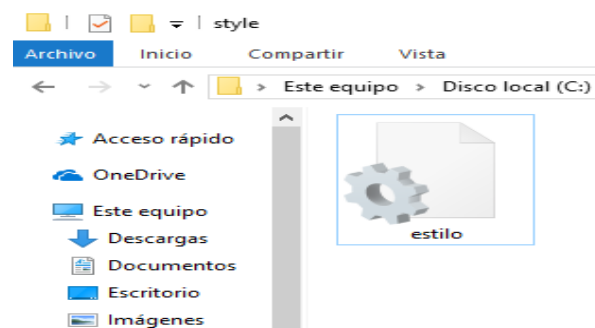
Como sería el caso del título principal de la página y el encabezado.

```
ejem.php — Users\...\proyecto  ejem.php — xampp\
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>proyecto jaguar</title>
5 </head>
6 <body>
7 <hr>
8   <h1>ITTG JAGUAR</h1>
9 </hr>
10  <h2>REGISTROS DE DATOS</h2>
11 </body>
12 </html>
```

En la carpeta de imágenes, se guardan las imágenes necesarias a utilizar para que se muestre en la página web. Organizando el contenido de imágenes que se utiliza para el diseño de la página.



Dentro de la carpeta de style se crea un documento llamado estilo.css que es creado en el editor de texto de sublime text, para la creación de los estilos de la página web.



En el documento estilos.css se realiza las siguientes líneas de códigos para los estilos de la página web. Con el cual, se le da un fondo de color a la página y color al texto con sus respectivas características.


```

1  body{
2      background-color:lightblue;
3      margin: 0;
4      height: 5000px
5      width: 100%
6  }
7  h1{
8      color:blue;
9      text-align:center;
10 }
11 h2{
12     color:red;
13     text-align:center;
14 }

```

Para que el código del documento estilos.css realice los cambios en el documento principal que es el index.php, donde se encuentra la programación principal de la página web se ingresa unas líneas de códigos necesarias que permitirán realizar los cambios de diseño.

`<meta charset="UTF-8">` Especifique la codificación de caracteres para el documento HTML.

`<meta name="viewport" content="width=device-width, initial-scale=1.0">`Le da instrucciones al navegador sobre cómo controlar las dimensiones y la escala de la página.

La etiqueta `< link rel="stylesheet" type="text/css" href="theme.css">` define un enlace entre un documento y un recurso externo.

La etiqueta `` define una imagen en una página HTML.

Los códigos se implementan en el programa principal organizando los datos.

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>proyecto jaguar</title>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <link rel="stylesheet" type="text/css" href="css/estilos.css">
8      
9      
10 </head>
11 <body>
12 <hr>
13     <h1>ITTG JAGUAR</h1>
14 </hr>
15     <h2>REGISTROS DE DATOS</h2>
16 </body>
17 </html>

```

Con la actualización de los códigos en el programa principal, la página web se aprecia de la siguiente forma, con los cambios realizados.



En la página es necesario mostrar los datos almacenados, con esto se crea una tabla que mostrara los datos obtenidos del arduino, este se crea con las siguientes líneas de código.

La etiqueta `<div id="main-container"></div>` es un divisor de contenido.

La etiqueta `<table>` define una tabla HTML.

El elemento `<tr>` define una fila de la tabla.

El elemento `<td>` define una celda de tabla.

El cual se utilizará para crear la tabla, para agrupar los elementos que conformaran a la tabla, creando una división del contenido, en fila y celda.

Nombrando la división del contenido de la celda con los siguientes nombres, ID, Fecha y Hora, y Sensor Serial.

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>proyecto jaguar</title>
5 <meta charset="UTF-8">
6 <meta name="viewport" content="width=device-width, initial-scale=1.0">
7 <link rel="stylesheet" type="text/css" href="css/estilos.css">
8 
9 
10 </head>
11 <body>
12 <hr>
13 <h1>ITTG JAGUAR</h1>
14 </hr>
15 <h2>REGISTROS DE DATOS</h2>
16 <div id="main-container">
17 <table>
18 <tr>
19 <td class="table_titles">ID</td>
20 <td class="table_titles">Fecha y Hora</td>
21 <td class="table_titles">Sensor Serial</td>
22 </tr>
23 </table>
24 </div>
25 </body>
26 </html>
```

Y en el documento estilo.css se realiza la modificación para el diseño de la tabla, establecido con el divisor de contenido (#main-container).

```

16 #main-container
17 {
18   margin:5px auto;
19   width:500px;
20 }
21 table{
22   background-color: white;
23   text-align: left;
24   border-collapse: collapse;
25   width:100%;
26 }
27 td{
28   border:solid 1px black;
29   padding:0;
30 }
31 .table_title{
32   background-color:#246355;
33   border-bottom: solid 5px #0F362D;
34   color:white;
35 }
36 body{
37   margin:0;
38   padding: 0;
39   overflow: scroll;
40 }
41 footer{
42
43   height: 50px;
44   background-color: #333;
45   color: #fff;
46   bottom: 0;
47   position: fixed;
48   text-align: center;
49   width: 100%;
50   clear: both;
51 }
30 }
31 .table_title{
32   background-color:#246355;
33   border-bottom: solid 5px #0F362D;
34   color:white;
35 }
36 body{
37   margin:0;
38   padding: 0;
39   overflow: scroll;
40 }
41 footer{
42
43   height: 50px;
44   background-color: #333;
45   color: #fff;
46   bottom: 0;
47   position: fixed;
48   text-align: center;
49   width: 100%;
50   clear: both;
51 }

```

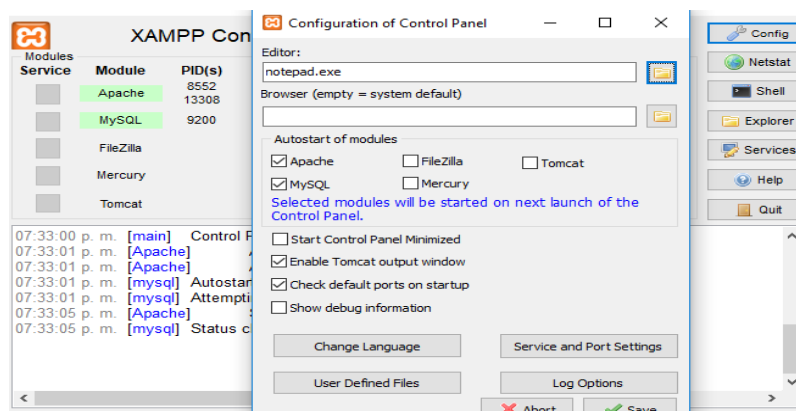
Con la actualización de la programación, con todos los datos requeridos para el diseño. En el localhost de la página web se puede observar los cambios realizados.



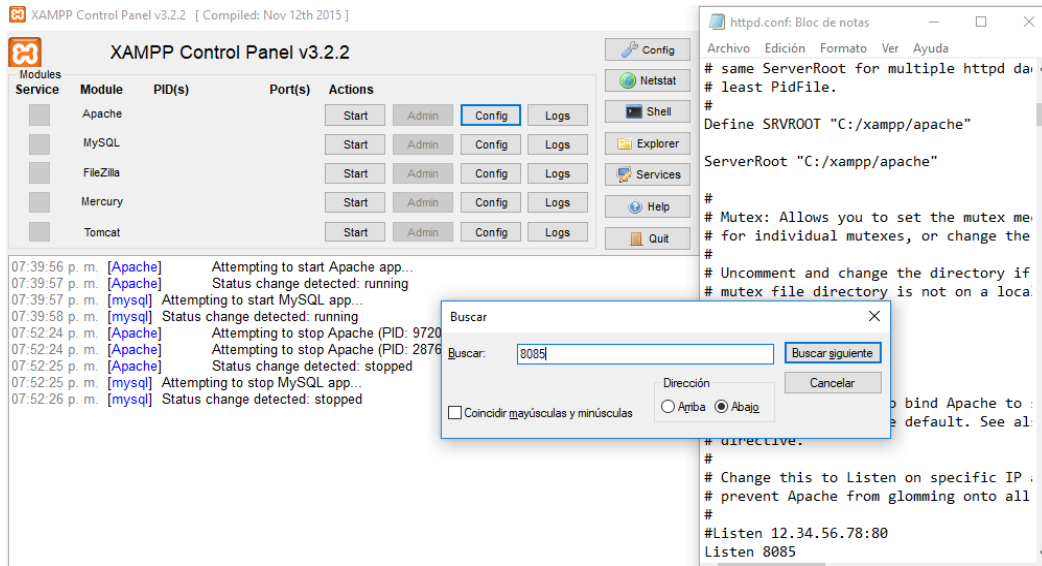
4.4 XAMPP

El programa se descarga desde el siguiente sitio <https://www.apachefriends.org/index.html> y se instala en la máquina.

Instalado el programa Xampp, se abre y se configura para que apache y MySQL inicie en automático al abrir el panel de control de Xampp.

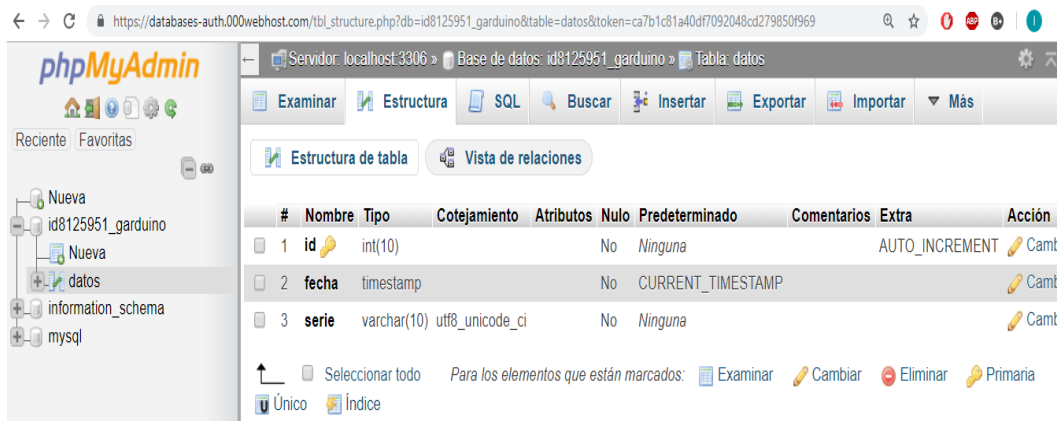


Y también se configura el puerto del que trae que es el 80 por otro que es el 8085, para realizar el cambio del puerto, se realiza la configuración de Apache el cual abre un block de notas, en donde se busca el 80 y se realiza el cambio en todas donde aparezca ese número, por el 8085. Esto se realiza con el fin de no utilizar un puerto que podría estar siendo utilizado por otro programa y así evitar problemas posteriores.



Realizados los cambios necesarios para adaptarlas para nuestro uso, se inicia apache y mysql. Para abrir el Localhost de phpmyadmin en la ventana de internet. Mostrando el phpmyadmin que es el programa principal para crear base de datos. Creando una base de datos, se crea una base de datos con el nombre de "garduino" dentro del cual se crea una tabla estableciendo su estructura y parámetros de resultados que almacenara la base de datos.

La tabla creada estará compuesta por, id, fecha y serie, que serán lo que se identifiquen los datos recolectados en cada columna.



4.5 CONEXIÓN

También se realiza un código en php, el cual será el encargado de comunicar la base de datos con la página web y con el arduino.

1. Primero se crea un archivo con el nombre de conec.php, en donde se realizará la conexión con la base de datos. Se pone el usuario, contraseña, host y el nombre de la base de datos. Como se muestra en la siguiente imagen.

```
data.php x estilo.css x index.php x conec.php
1 <?php
2 $usuario = "ittg" ;
3 $contraseña = "ittg" ;
4 $host = "localhost" ;
5 $base = "arduino" ;
6 $conn = mysqli_connect ( $host, $usuario, $contraseña );
7 $db=mysqli_select_db ( $conn, $base );
8 mysqli_set_charset($conn,"SET NAMES utf8");
9
10 ?>
11
```

En la programación se utiliza las siguientes funciones:

`mysqli_connect ()`: abre una nueva conexión con el servidor MySQL.

Sintaxis: `mysqli_connect(host, username, password, dbname);`

`mysqli_select_db ()`: se usa para cambiar la base de datos predeterminada para la conexión.

Sintaxis: `mysqli_select_db (connection, dbname);`

`mysqli_set_charset ()`: especifica el conjunto de caracteres predeterminado que se utilizará al enviar datos desde y hacia el servidor de la base de datos.

Sintaxis: `mysqli_set_charset (connection, charset);`

2. Se crea otro archivo con el nombre de data.php, el que se comunica con el arduino para registrar los datos y comunicarse con el archivo conec.php. que sirve para comunicarse con la base de datos y la página web.

```
data.php x estilo.css x index.php x conec.php
1 <?php
2 include("conec.php");
3
4 $serie= mysqli_real_escape_string($conn,$_GET['serie']);
5 $sql_insert ="INSERT INTO datos (serie) VALUES('".$serie."'";
6 if(mysqli_query($conn,$sql_insert))
7 {
8     echo "Done";
9     mysqli_close($conn);
10 }
11 else
12 {
13     echo "error is".mysqli_error($conn);
14 }
15
16 ?>
```

En la programación del archivo se utiliza las siguientes funciones:

`mysqli_real_escape_string ()`: escape caracteres especiales en una cadena para usar en una declaración SQL.

Sintaxis: `mysqli_real_escape_string(connection,escapestring);`

`mysqli_query ()`: realiza una consulta en la base de datos.

Sintaxis: `mysqli_query(connection,query,resultmode);`

`mysqli_close ()`: cierra una conexión de base de datos abierta previamente.

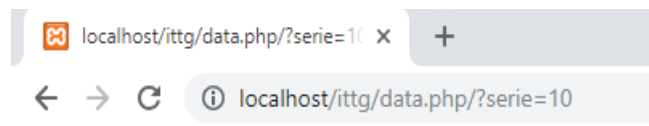
Sintaxis: `mysqli_close(connection);`

Con lo realizado, se realiza una prueba para determinar el funcionamiento de la conexión con la programación y la base de datos sea correcta. Donde se ingresa a una nueva venta de Google, y se ingresa el siguiente enlace:

<http://localhost/ittg/data.php/?serie=10>

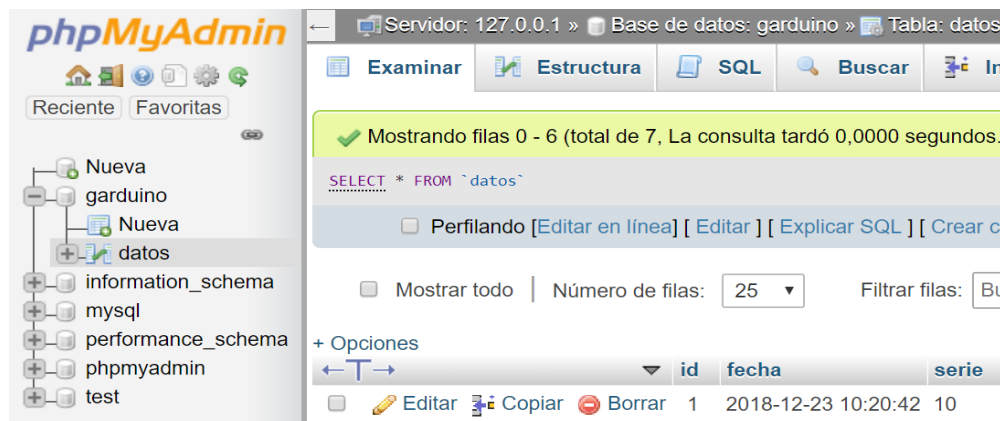
El cual lee los archivos creados, para ingresar el dato de serie que puede ser un número o letra dependiendo de lo que se quiera mostrar en la base.

Al ingresar al enlace se visualiza lo siguiente:



Done

Donde se observa un mensaje que dice Done, el cual indica que el dato se ingresó correctamente a la base de datos. El cual se podrá observar en el phpmyadmin del localhost, sin ningún problema.



En este caso se ingresó el número 10 y se pudo observar el id que es la identificación, la fecha y hora en que se ingresó el dato.

Ese mismo dato se pudo observar en la página web, al actualizar el localhost en el internet. Esto indica que el funcionamiento de la base de datos con la página web es la correcta.



ID	Fecha y Hora	Sensor Serial
1	2018-12-23 10:20:42	10

4.6 PROGRAMACION DEL ARDUINO

En esta parte se diseña la programación del arduino, que realiza la comunicación entre la página web. Para la programación es necesario instalar las librerías para que el ethernet shield funcione correctamente.

Las librerías son las siguientes:

- Ethernet
- SPI

Con las librerías instaladas se realiza la programación de la siguiente forma:

```
#include<SPI.h>
#include<Ethernet.h>
byte mac[] = {0x00, 0xAA, 0xBB, 0xCC, 0xDE, 0x02};
//byte servidor[]={192, 168, 0, 22};
char servidor[]="http://tecjaguar.000webhostapp.com";
#define puertoHTTP 80
EthernetClient cliente;
```

```

//=====
void setup(){
  Serial.begin(9600);
  Ethernet.begin(mac);
  //pinMode(sensor,INPUT);
  if(Ethernet.begin(mac)==0){
    Serial.println("no se pudo conectar a la red");
    Ethernet.begin(mac);
  }
  Serial.print("conectado a la red, no ip:");
  Serial.println(Ethernet.localIP());
}
void loop(){
  if(cliente.available()){
    char datosRetornados=cliente.read();
    Serial.print(datosRetornados);
  }
  if(!cliente.connected()){
    cliente.stop();
  }
  char comando=Serial.read();
  if (comando=='1'){

    Serial.println("conectando al servidor...");

    if (cliente.connect(servidor,puertoHTTP)){

      cliente.println("GET /data.php HTTP/1.0");
    }
  }
}

```



```
cliente.println("Host: tecjaguar.000webhostapp.com");
cliente.println("Connection: close");
cliente.println();

}else{
  Serial.println("falla la conexion con el servidor");
}
}
//delay(5000);
}
```

CAPÍTULO V

5.1 RESULTADOS, PLANOS, GRAFICAS, PROTOTIPOS Y PROGRAMAS

En base a todos los pasos realizados, El prototipo realizado de la página web se puede visualizar desde una computadora o un teléfono móvil que cuente con conexión a internet, y se puede acceder ingresando el siguiente link:

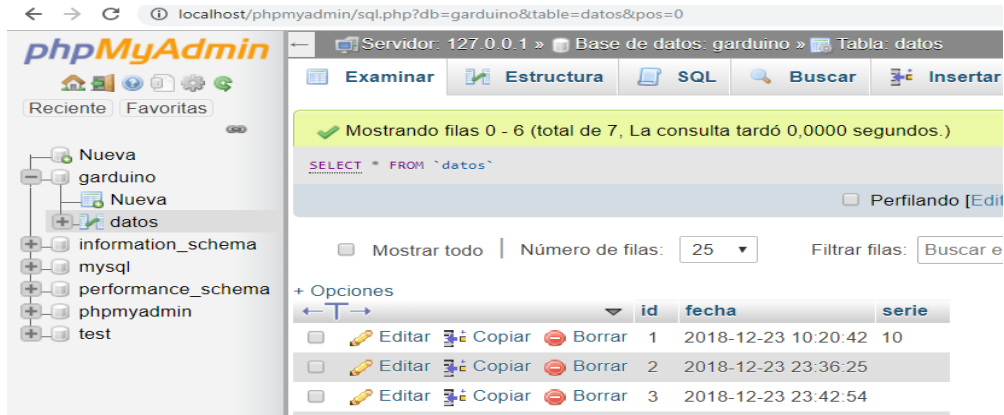
<https://tecjaguar.000webhostapp.com>

Esto permite que la pagina pude ser visible en otros dispositivos desde diferentes partes, con solo contar con conexión a internet.



Lo que permite que los datos sean visualizados en tiempo real, porque se puede visualizar la información que manda la comunicación del arduino en conjunto con el ethernet shield.

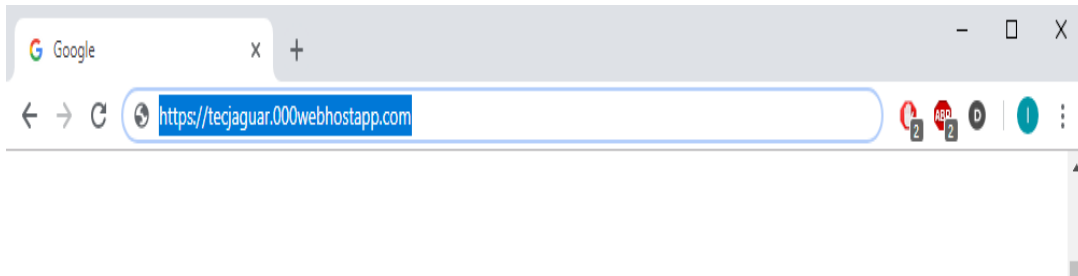
La base de datos creada es la que permite que los datos sean almacenados y visualizados mediante la página web, en este caso los datos que manda el arduino se agrupa de acuerdo, a la hora y fecha que se ingresa.



Permitiendo que al actualizar la página se visualicen los datos.



Para poder ingresar al servidor web, se puede hacer desde cualquier lugar con servicio de internet. Ingresando en Google y abrir una ventana, escribiendo el link de la página web: <https://tecjaguar.000webhostapp.com> en la ventana abierta. Realizado este pasa se podrá visualizar la página web creada.



El siguiente paso es comprobar en el cmd o símbolo del sistema de la máquina que se esté utilizando la dirección ip, este paso se realiza lo siguiente:

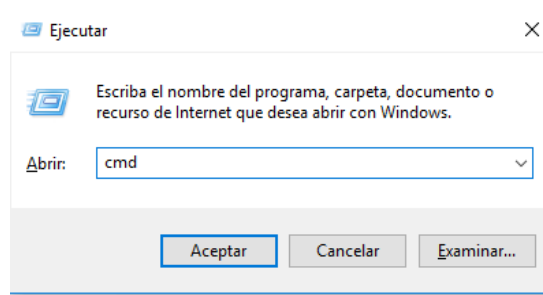
Paso 1

Pulsamos las teclas



Paso 2

Nos aparecerá la ventana de Ejecutar, donde simplemente escribimos CMD y pulsamos en aceptar y aparecerá la ventana de comandos.



Paso 3

Abierto el CMD, se ingresa ipconfig para verificar la dirección ip a la que está conectado la computadora.

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.16299.904]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.
C:\Users\limberg>ipconfig

Configuración IP de Windows

Adaptador de Ethernet Ethernet:
```

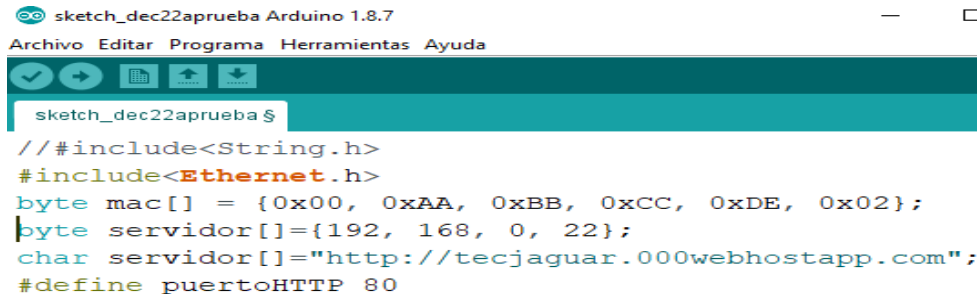
Paso 4

Verificamos la dirección IPv4 que es la dirección ip a la que estamos conectados. Y se escribirá en la programación de arduino. La IPv4 depende de la red de internet en la que se está conectado, en todo caso se tendrá que realizar los pasos anteriores para verificar la dirección ip. Para no tener problemas con la conexión.

```
Dirección IPv4. . . . . : 192.168.0.14
Máscara de subred . . . . . : 255.255.255.0
```

Paso 5

En este paso se abre el sketch de arduino, donde se tiene la programación del programa realizado. Y se ingresa la dirección IPv4 para realizar la conexión del arduino con el sitio web.

A screenshot of the Arduino IDE interface. The title bar reads "sketch_dec22aprueba Arduino 1.8.7". The menu bar includes "Archivo", "Editar", "Programa", "Herramientas", and "Ayuda". Below the menu bar is a toolbar with icons for check, back, forward, upload, and download. The main text area shows the following code:

```
sketch_dec22aprueba $
//#include<String.h>
#include<Ethernet.h>
byte mac[] = {0x00, 0xAA, 0xBB, 0xCC, 0xDE, 0x02};
byte servidor[]={192, 168, 0, 22};
char servidor[]="http://tecjaguar.000webhostapp.com";
#define puertoHTTP 80
```

En este paso también se ingresa la dirección de nuestro sitio web y se define el número del puerto en este caso es el 80, esto se realiza para establecer a que sitio web nos queremos comunicar.

Paso 6

Se ingresa el método del envío de los datos, que será por el método GET y el nombre del archivo que es data.php el cual se comunica con el sitio web. También se ingresa nuevamente el nombre del sitio web. Para que la conexión se pueda realizar sin problemas.

```
if (cliente.connect(servidor,puertoHTTP)) {

    cliente.println("GET /data.php HTTP/1.0");
    cliente.println("Host: tecjaguar.000webhostapp.c
```

Realizando todos los pasos anteriores se puede establecer la comunicación del arduino y el sitio web creado.

CONCLUSIÓN

Al inicio de la elaboración del proyecto se tenía claro que se carecía del conocimiento en el diseño de una página web y el uso de base de datos, pero con la iniciativa se adquirieron los conocimientos en dicho ramo, se logró superar estos inconvenientes y se llevó a buen término esta residencia.

Se logró desarrollar la página web del animatronic con las características necesarias que lleva una página web en su desarrollo, organizando el entorno de la página que se desea mostrar incluyendo imágenes, el texto y la tabla donde se almacenan los datos.

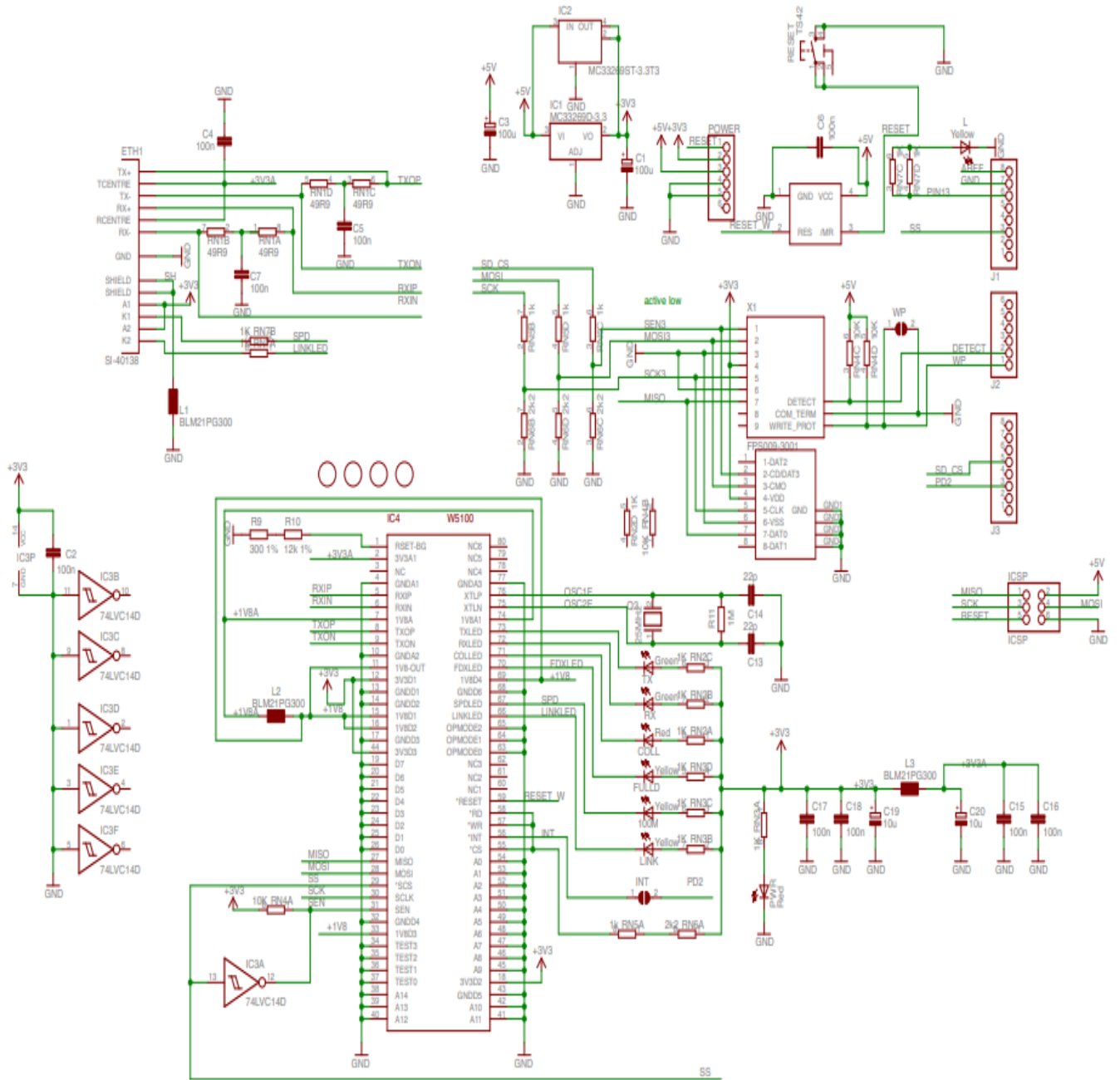
Entorno a trabajar con lo que es el internet de las cosas (IOT), tiene un gran alcance en lo que respecta al avance de la tecnología y la comunicación. El cual se puede continuar con su desarrollo para darle un mejor enfoque, optimizando su desarrollo para implementar opciones de control.

BIBLIOGRAFÍA

- [1] <https://www.omicrono.com>
- [2] <http://amrproducciones.blogspot.mx/2013/03/animatronics.html>
- [3] Hernández, C. Animatrónica, una aproximación a partir de su diseño y construcción. Licenciatura. San Buenaventura University (2012).
- [4] Edward R. Prescott, Steven B, Joe Meiser., “Persistent tracking and monitoring of animatronics using IoT capabilities”, IEEE (N° 8446618, agosto 27, 2018).
- [5] Daneri, Pablo A. PLC Automatización y Control Industrial (1ª Edición), Hasa- Hispano Americana, Argentina, Buenos Aires: 2008.
- [6] Ebel F. Ider S. Prede G. Scholz D. Fundamentos de la técnica de automatización, Festo Didactic GmbH & Co.KG, Alemania 2008.
- [7] H.Eltom Rammah, A.Hamood Enan, A.Mohammed AbdAlrahman. ” Early Warning Firefighting System Using Internet of Things”, IEEE (N° 18229361, noviembre 18, 2018).
- [8] Corso Felipe, Camargo Yeison, Ramírez Leonardo.,” Wireless Sensor System According to the Concept of IoT -Internet of Things-”, International Conference on Computational Science and Computational Intelligence (N° 14333756, Mayo 29, 2014).
- [9] <https://www.arduino.cc/>
- [10] WIZnet. W5100, Datasheet Versión 1.1.6, WIZnet Co., 2008.
- [11] López Pérez, E. Ingeniería en microcontroladores protocolo spi (serial Peripheral interface) ingeniería en microcontroladores, México Distrito Federal.
- [12] <https://www.arduino.cc/en/Guide/Environment>
- [13] Torrente Artero Oscar, Arduino. Curso práctico de Formación, (1° Edición), Alfaomega Grupo Editor, Mexico:2013.
- [14] <http://tecnologiaunida2015.blogspot.com>
- [15] <https://www.sublimetext.com/>
- [16] <https://www.w3schools.com>
- [17] <https://www.apachefriends.org/index.html>

ANEXOS

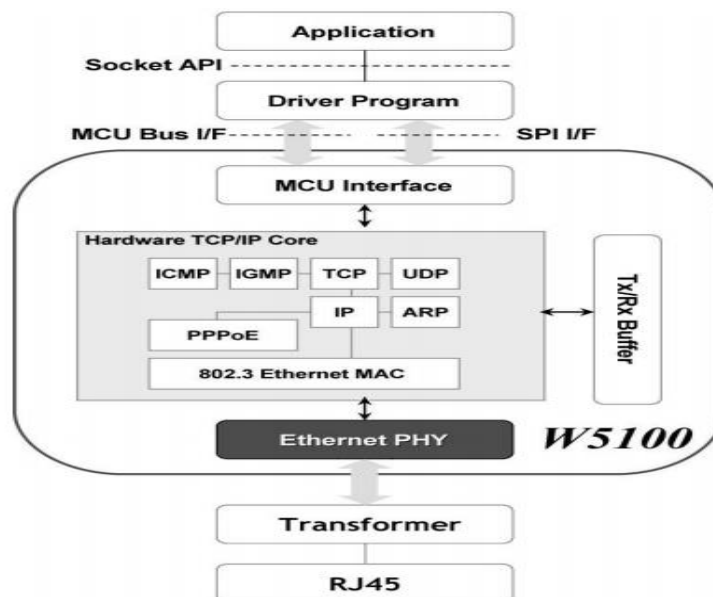
Modulo Ethernet



Document History Information

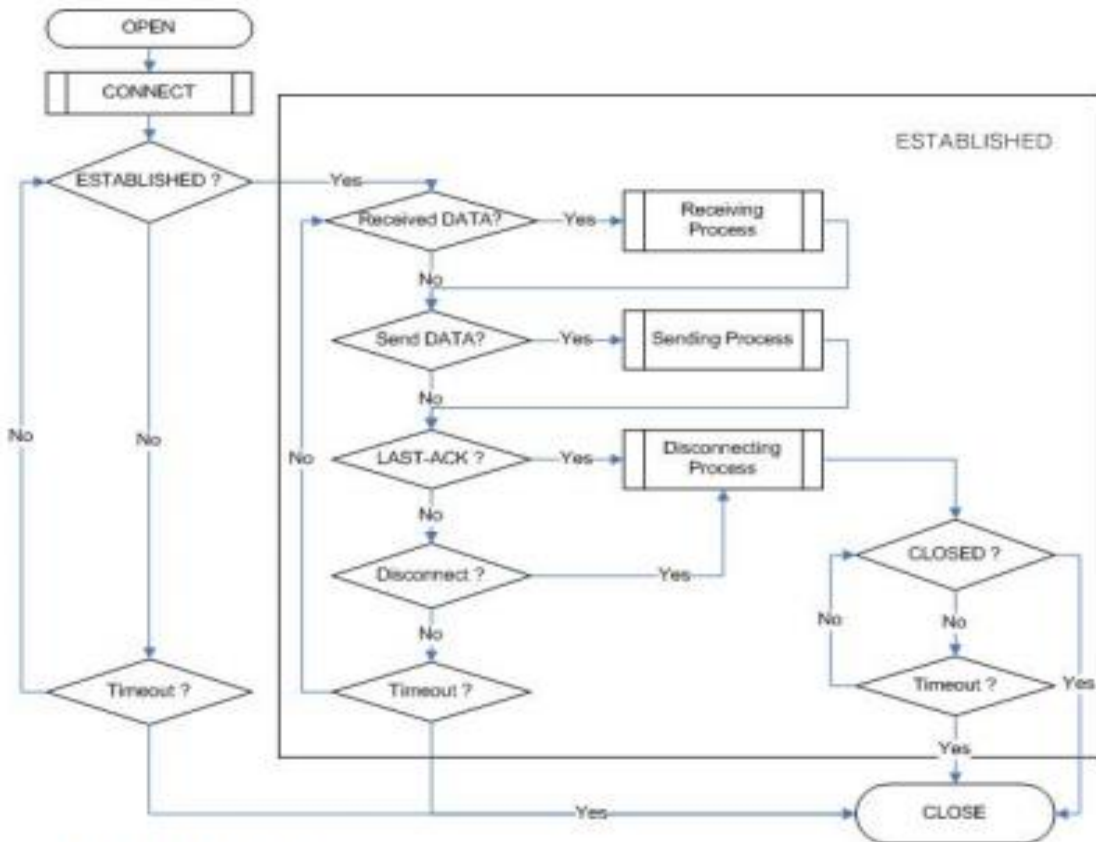
Version	Date	Descriptions
Ver. 1.0.0	Dec. 21, 2006	Released with W5100 Launching
Ver. 1.0.1	Jan. 10, 2006	LB bit in Mode register is not used . W5100 is used only in Big-endian ordering.
Ver. 1.1.1	Jun. 19, 2007	Modified the OPMODE2-0 signals descriptions (P. 10) Modified the TEST_MODE3-0 signals description (P.11) Modified the Colck signals description (P.12) Modified the LINKLED signal description (P.12) Modified the explanation of RECV_INT in Sn_IR register (P. 27) Replaced the reset value of Sn_DHAR register (0x00 to 0xFF, P. 30) Modified the explanation of Sn_DIPR, Sn_DPORT register(P. 31) Replaced the reset value of Sn_MSS register (0xFFFF to 0x0000, P. 31)
Ver. 1.1.2	Sep. 28, 2007	Modified the Operating temperature (P. 63)
Ver. 1.1.3	Oct. 8, 2007	Changed the wrong word "MISO signal" (P. 10) Modified the SPI Timing diagram and description (P. 66)
Ver. 1.1.4	Oct. 18, 2007	Modified the diagram (P. 40)
Ver. 1.1.5	Nov. 1, 2007	Modified the Crystal Characteristics value (P. 67)
Ver. 1.1.6	Jan. 30, 2008	Modified the SEN signals description (P.10) Changed the wrong word "SCLK" (P. 66)

Block Diagram



5.2.1.2 CLIENT mode

Whole process is shown as below.



■ **Socket Initialization**

Refer to "5.2.1.1 SERVER mode" (The operation is same as SERVER).

■ **CONNECT**

Send connection request to remote HOST(SERVER) is as below.

```

[
/* Write the value of server_ip, server_port to the Socket n Destination IP Address
   Register(Sn_DIPR), Socket n Destination Port Register(Sn_DPORT). */
Sn_DIPR = server_ip;
Sn_DPORT = server_port;
/* set CONNECT command */

```