



UNIVERSIDAD MICHOACANA DE SAN NICOLÁS DE HIDALGO

FACULTAD DE INGENIERÍA ELÉCTRICA

TESIS

IoT – INTERNET DE LAS PLANTAS
CULTIVOS HIDROPÓNICOS DE ALTA TECNOLOGÍA

PARA OBTENER EL TÍTULO DE INGENIERO EN ELECTRÓNICA

PRESENTA
JESÚS GIOVANNI ESTRADA CÁRDENAS

CON LA ASESORÍA DE
M. C. FÉLIX JIMÉNEZ PÉREZ

NOVIEMBRE 2019
Morelia, Michoacán

DEDICATORIA

Dedico este trabajo:

A mis padres, mis hermanos y mi familia. ¡Sí se pudo!

A todas mis amistades, que me han dado tantos momentos de diversión y su compañía.

A mis compañeros a lo largo de toda esta travesía de la vida, llamada educación.

A todo el profesorado, del cual, he tenido la dicha de ser alumno. Incluso a algunos de los que no.

A mi país, México, por brindarme la oportunidad de tener educación y cambiar toda mi cosmovisión.

A mí mismo. Sólo yo sé a ciencia cierta todo lo que ha significado atravesar este sinuoso camino de la educación, y todos los procesos por los que me he ido transformando para bien a lo largo de estos años. ¡Esto es por y para ti, campeón!

AGRADECIMIENTOS

A mis padres: por haber sido durante todos estos años, mi apoyo, mi soporte, el origen fuerte de mis convicciones y mis valores; por haberlo dado todo, incluso cuando no lo tenían, con tal de procurar mi bienestar y el de mis hermanos; por esas noches de desvelos y sacrificios; por esos abrazos en los momentos más necesarios; por jugar conmigo incluso cuando estaban cansados; por levantarme cuando me caía y me caía y me caía; por estar siempre dispuestos a responder mis preguntas y fomentar la curiosidad en mí, así como esos deseos de siempre seguir aprendiendo, descubriendo, superándome; por enseñarme lo que es una familia; por confiar en mí y permitirme confiar en ustedes plenamente. Tantas cosas me han dado que, si me pusiera a enlistarlas todas, necesitaría un documento aún más grande que esta tesis para agradecerse los. Soy muy afortunado teniéndolos como padres y amigos. Los amo.

A mis hermanos: por ser siempre mi fuente de júbilo y alegría; por aguantar esos momentos en los que, por tener otras ocupaciones, no pude jugar con ustedes o dedicarles algunas horas de charla; por acompañarme en la vida durante todos estos años. Les tengo un cariño inestimable, y quiero que sepan que siempre estaré ahí para ustedes si me necesitan.

A Evelyn: por estar ahí apoyándome y acompañándome desde hace ya 8 años; por impulsarme a no rendirme y perseguir mis sueños; por aceptarme con mis gustos, costumbres y manera de ser; por ser mi compañera e impulsarme en los momentos difíciles. Muchas gracias, pequeña.

Un especial agradecimiento a mi asesor, Félix: Por enseñarme tantas cosas dentro y fuera de las aulas; por ayudarme a entender mejor cómo funciona el mundo; por no dudar ni un momento en utilizar todos los recursos disponibles con tal de transmitir algún conocimiento; por convertirse en fuente de inspiración para mí y para tantos y tantos estudiantes más; por demostrarme que la vocación existe, con su ejemplo; por volverse mi amigo; por mostrarme que cualquier problema, por difícil que parezca, puede sortearse con esfuerzo, dedicación y hacer que sea al final “una babada”; por enseñarme que sí, claro, por supuesto que podemos aprender todo eso que queramos, con las herramientas que la universidad y la educación a lo largo de nuestras vidas, nos ha dado. Muchas gracias, “profe”.

A mi alma mater UMSNH: por formarme como profesionista y ofrecerme tantas oportunidades de crecer; por todas las herramientas que puso a mi alcance; por brindarme el acceso a la educación y poner a mi disposición sus instalaciones y equipo para mi desarrollo, y el de todos mis compañeros universitarios. ¡Siempre Zorros! ¡Siempre nicolaitas!

Agradezco también a la UNAM y a la Universidad Central de Colombia, por permitirme formar parte de su alumnado por un tiempo. Ambas aportaron elementos muy significativos a mi formación como ingeniero y como persona.

A todos mis demás familiares y amigos: porque les aseguro que todos y cada uno de ustedes, han dejado algo en mí, que me ha ayudado a convertirme en lo que soy hoy. ¡Muchas gracias a todos! ¡Lo hicimos!

RESUMEN

Se estima que el 80% de la población mundial, vivirá en zonas urbanas para el 2050. Mientras que 4/5 partes de las tierras cultivables del planeta ya se utilizan. De ahí surge la necesidad de buscar fuentes alternativas de cultivo.

Una solución que responde a esa problemática es la hidroponía, la cual permite cultivar en zonas urbanas, ahorrar agua y obtener más y mejores cultivos que la agricultura tradicional, empleando la misma área. Si se combina la hidroponía con la posibilidad de supervisar y controlar parámetros del entorno de desarrollo de las plantas de manera remota, se tiene entonces un sistema diseñado para que cada hogar pueda tener sus cultivos hidropónicos en interiores, que ayude a su economía y contribuya al sustento familiar.

En esta tesis se expone el diseño y desarrollo de un sistema de monitoreo y control para cultivos hidropónicos en interiores, basado en un servidor y microcontroladores, que cuenta con los sensores y actuadores necesarios para el desarrollo de cultivos, tales como Iluminación, temperatura, pH, conductividad eléctrica y humedad. El sistema tiene, además, un registro en una base de datos del comportamiento de las variables físicas del sistema y los cambios históricos realizados en el mismo. El usuario interactúa con el sistema a través de una página web, desde la cual pueden visualizarse de manera gráfica los datos recabados por los sensores, así como también pueden programarse o realizarse modificaciones al estado de los actuadores del sistema. También cuenta con una cámara web que permite tomar fotografías de los cultivos para su visualización en esa misma página.

En conjunto, se tiene un sistema que incluye dispositivos de monitoreo y control, un servidor y una aplicación de gestión multiplataforma de diseño minimalista e intuitivo, con el cual es posible semi-automatizar y facilitar el cultivo de diversas plantas dentro de una casa-habitación.

PALABRAS CLAVE

ESP8266, Microcontrolador, MongoDB, NodeJS, Raspberry Pi.

ABSTRACT

It is estimated that 80% of the world's population will live in urban areas by 2050. While 4/5 parts of the planet's arable land is already used. Hence the need to find alternative sources of cultivation.

A solution to those problems is hydroponics. Which allows growing crops in urban areas, save water and obtaining more and better crops than traditional agriculture, using the same area. And if the possibility of monitoring and controlling remotely some parameters of the plant development environment is added, a system so that each household can have its hydroponic crops indoors, which helps its economy and contributes to family support, is born.

This thesis exposes the design and development of a monitoring and control system for indoor hydroponic crops, based on a server and microcontrollers, which has the necessary sensors and actuators for crop development, such as lighting, temperature, pH, electrical conductivity and humidity. The system also has a record in a database of the behavior of the physical variables of the system and the historical changes made to itself. The user interacts with the system through a web page, from which the data collected by the sensors can be displayed graphically, and from where changes can be programmed or made to the actuators of the system. It also has a webcam that allows taking pictures of the crops for viewing on that same page.

Together, there is a system that includes monitoring and control devices, a server and a multiplatform management application with a minimalist and intuitive design, with which it can be semi-automated and facilitated the cultivation of various plants within a houserroom.

KEYWORDS

ESP8266, Microcontroller, MongoDB, NodeJS, Raspberry Pi.

CONTENIDO

Lista de figuras	IX
Lista de tablas	XIII
Abreviaturas y acrónimos indexados.....	XIV
Capítulo 1. Introducción.....	1
1.1. Antecedentes	1
1.1.1. Breve historia de la agricultura.....	1
1.1.2. Invernaderos.....	3
1.2. Estado del arte	8
1.2.1. MiniFarm.....	8
1.2.2. Smartcultiva	9
1.3. Justificación	11
1.4. Objetivos	12
1.4.1. Objetivo general	12
1.4.2. Objetivos particulares.....	12
1.5. Metodología.....	12
1.6. Descripción de capítulos	12
Capítulo 2. Marco teórico	14
2.1. Introducción	14
2.2. Métodos de cultivo alternativos	14
2.2.1. Agricultura biodinámica.....	14
2.2.2. Acuaponia	15
2.2.3. Aeroponia	16
2.2.4. Hidroponia	17
2.3. Microcontroladores.....	18
2.3.1. Compilación	19
2.3.2. System on a Chip (SoC)	20
2.3.3. Placa de desarrollo.....	20
2.4. Ordenador de Placa Única (SBC)	21
2.5. Sensores	22

2.5.1.	Sensores de temperatura	23
2.5.2.	pH-metros.....	27
2.5.3.	Medidor de conductividad eléctrica	28
2.5.4.	Sensor de intensidad luminosa.....	29
2.5.5.	Sensor de humedad	30
2.6.	Actuadores	32
2.6.1.	Bombas	33
2.6.2.	Relevadores	34
2.6.3.	Iluminación	35
2.7.	Periféricos	39
2.7.1.	Cámaras web.....	39
2.8.	Redes de computadoras.....	40
2.8.1.	Red de Área Local (LAN).....	40
2.8.2.	Red de Área Local Virtual (VLAN).....	41
2.8.3.	Internet	41
2.8.4.	Módem.....	42
2.8.5.	Ruteador	42
2.9.	Protocolos de comunicación	44
2.9.1.	Protocolo HTTP	45
2.9.2.	Protocolo IP.....	47
2.9.3.	Protocolo TCP.....	50
2.9.4.	SSH	54
2.9.5.	MQTT	56
2.10.	Base de datos	57
2.10.1.	Bases de datos relacionales	57
2.10.2.	Bases de datos no relacionales	58
Capítulo 3.	Diseño y descripción del hardware.....	62
3.1.	Introducción	62
3.2.	Instalación hidropónica PVC.....	63
3.2.1.	Solución nutritiva	64

3.2.2.	Semillas	65
3.3.	System on Chip (SoC)	65
3.3.1.	Cloud Build Service	70
3.3.2.	Escribiendo el firmware	71
3.4.	Cámara serial.....	72
3.5.	Sensores	75
3.5.1.	Sensor de temperatura y humedad DHT11	75
3.5.2.	Sensor de luminosidad.....	77
3.5.3.	Sensor de CE	79
3.5.4.	Sensor de pH.....	81
3.6.	Actuadores	85
3.6.1.	Bomba de agua	85
3.6.2.	Tira de LEDs.....	86
3.7.	Servidor web implementado en un SBC.....	88
3.8.	Otros circuitos integrados (CI)	88
3.8.1.	Multiplexor/Demultiplexor	88
3.8.2.	Transistor BC548B.....	90
3.8.3.	Driver L293B.....	92
Capítulo 4.	Diseño y descripción del software	94
4.1.	Introducción	94
4.2.	Flujo de datos.....	95
4.2.1.	Autenticación	95
4.2.2.	Captura de nueva fotografía	97
4.2.3.	Recepción y almacenamiento de datos de los sensores	99
4.2.4.	Graficación de datos de los sensores.....	100
4.2.5.	Obtención del estado de los actuadores	100
4.2.6.	Cambio en el estado de un actuador	102
4.3.	Servidor web	103
4.3.1.	NodeJS	103
4.3.2.	MongoDB	104

4.3.3.	Servicios	106
4.4.	System on Chip (SoC)	108
4.4.1.	ESPlorer.....	108
4.4.2.	Lenguaje Lua	109
4.4.3.	Programas de los microcontroladores.....	110
4.5.	Sitio web.....	117
4.5.1.	Página de autenticación.....	117
4.5.2.	Página principal.....	118
Capítulo 5.	Pruebas y resultados.....	125
5.1.	Pruebas con la cámara serial.....	125
5.2.	Uso de dos SoC.....	127
5.3.	Inicio de sesión.....	128
5.4.	Solicitud de nueva fotografía	129
5.5.	Visualización de datos en un gráfico	132
5.6.	Consulta y control del estado de los actuadores	134
5.7.	Cierre de sesión.....	135
	Conclusiones y trabajos futuros.....	136
	Apéndices.....	138
Apéndice 1.	HTTP.....	138
	Lista de comandos HTTP	138
	Encabezados HTTP	138
	Encabezados de respuesta HTTP	139
	Códigos de respuesta HTTP.....	139
Apéndice 2.	Especificaciones del ESP8266EX	141
Apéndice 3.	Diagramas de pines del NodeMCU V2 y V3	142
	Referencias	144

LISTA DE FIGURAS

Fig. 1-1. Primera sembradora del mundo, construida por Jethro Tull [6].	2
Fig. 1-2. Máquina cosechadora mecánica tirada por caballos, inventada por Cyrus McCormick [7].	3
Fig. 1-3. Orangerie ubicado en el Real Jardín Botánico de Kew, Londres, Inglaterra [10].	4
Fig. 1-4. Acolchado negro en cultivo de piña [13].	5
Fig. 1-5. Cortavientos semipermeable [12].	5
Fig. 1-6. Colocación mecanizada de cubierta flotante de material agrotexil [12].	6
Fig. 1-7. Túneles bajos [12].	6
Fig. 1-8. Túneles altos [12].	7
Fig. 1-9. Invernadero multitúnel de lámina plástica [12].	7
Fig. 1-10. Diseño modular MiniFarm de Replantable [15].	8
Fig. 1-11. Estructura del sistema Replantable [15].	9
Fig. 1-12. Película de pulpa reciclada con semillas y nutrientes [15].	9
Fig. 1-13. Aplicación y plataforma web de Smartcultiva para el monitoreo y control de variables de cultivo [16].	10
Fig. 1-14. Prestaciones de los sistemas y productos Smartcultiva para el análisis de datos [16].	10
Fig. 2-1. Desbalances más comunes en la acuaponia [20].	16
Fig. 2-2. Plantas cultivadas con aeroponía, y sus largas raíces [21].	17
Fig. 2-3. Pintura idealizada del siglo XIX de los Jardines Colgantes de Babilonia [24].	17
Fig. 2-4. Diagrama a bloques de un microcontrolador [25].	18
Fig. 2-5. Diagrama a bloques del funcionamiento del compilador [25].	20
Fig. 2-6. Estructura general de una placa de desarrollo [28].	21
Fig. 2-7. Ordenador de Placa Única (SBC) marca Raspberry Pi modelo 3B+ [30].	22
Fig. 2-8. Diagrama de bloques de un sensor [31].	22
Fig. 2-9. Cable termopar desnudo y con hebra cerámica [32].	23
Fig. 2-10. Sensor de temperatura por resistencia (RTD) [32].	24
Fig. 2-11. Sensor de temperatura bimetálico [32].	25
Fig. 2-12. Termómetros de cristal, de dilatación de mercurio [32].	25
Fig. 2-13. Etiquetas no reversibles de temperatura [32].	26
Fig. 2-14. Termómetro infrarrojo de mano [34].	26
Fig. 2-15. Medidor electrónico de pH portátil [35].	28
Fig. 2-16. Medidor de conductividad potenciométrica de sobremesa [38].	28
Fig. 2-17. Resistor Dependiente de la Luz (LDR) [40].	29
Fig. 2-18. Sensor analógico de temperatura y humedad [42].	30
Fig. 2-19. Sensor digital de temperatura y humedad [43].	31
Fig. 2-20. Bomba centrífuga [47].	33
Fig. 2-21. Principio de funcionamiento de una bomba centrífuga [47].	34
Fig. 2-22. Electroimán [48].	35
Fig. 2-23. Modelo de relevador sencillo [48].	35
Fig. 2-24. Partes de una bombilla [50].	36

Fig. 2-25. Lámparas incandescentes de halógeno [50].....	36
Fig. 2-26. Lámparas fluorescentes [50].....	37
Fig. 2-27. Componentes de un LED [50].	38
Fig. 2-28. Cámara web para computadora [53].....	39
Fig. 2-29. El módem convierte señales digitales a analógicas y viceversa [59].	42
Fig. 2-30. Fotografía de un ruteador común [61].	43
Fig. 2-31. Etapas de comunicación entre navegador y servidor [63].	45
Fig. 2-32. Ejemplo esquematizado de un datagrama [64].....	48
Fig. 2-33. Multiplexado realizado por el protocolo TCP [66].....	50
Fig. 2-34. Modelo de la aplicación cliente-servidor [67].	51
Fig. 2-35. Formación de los segmentos en TCP [66].....	52
Fig. 2-36. Tres etapas en el inicio de una comunicación TCP [66].	54
Fig. 2-37. Flujo de configuración simplificado de una conexión de SSH [68].	55
Fig. 2-38. Arquitectura de MQTT con topología de estrella [69].	56
Fig. 2-39. Estructura de una base de datos SQL por medio de tablas [71].....	58
Fig. 2-40. Diagrama de ejemplo de una base de datos key-value [72].....	59
Fig. 2-41. Ejemplo en diagrama de una base de datos documental [72].	60
Fig. 2-42. Diagrama de base de datos en grafo [73].	60
Fig. 3-1. Diagrama a bloques del hardware del sistema.	62
Fig. 3-2. a) Acotaciones de Fig. 3-1. b) Composición interna del SoC representado en Fig. 3-1.	63
Fig. 3-3. Diseño de la estructura de PVC utilizada para el sistema hidropónico.	63
Fig. 3-4. Vista transversal del modelo de la tubería PVC que permite la aireación de las raíces.	64
Fig. 3-5. Semillas de lechuga italiana marca Hortaflor.	65
Fig. 3-6. Composición de la placa de desarrollo NodeMCU [28].	66
Fig. 3-7. Diagrama funcional del SoC ESP8266EX [75].	67
Fig. 3-8. Diagrama de pines del SoC ESP8266EX [75].	67
Fig. 3-9. Módulo ESP-12E [28].	68
Fig. 3-10. Variantes en el mercado del NodeMCU [28].	69
Fig. 3-11. Pines digitales recomendados para utilizarse en el NodeMCU V2 y V3 [28].	70
Fig. 3-12. Listado de módulos disponibles para crear el firmware del NodeMCU desde el Cloud Build Service.	71
Fig. 3-13. Comando en terminal de Ubuntu para cargar el firmware en el NodeMCU.	72
Fig. 3-14. LinkSprite LS-Y201-TTL [79].	73
Fig. 3-15. Medidas de la LinkSprite JPEG Color Camera [82].	73
Fig. 3-16. Pines disponibles en la cámara LinkSprite modelo LS-Y201 [82].	74
Fig. 3-17. Vista frontal del sensor de temperatura y humedad DHT11 [83].....	76
Fig. 3-18. Dimensiones en mm del DHT11 y numeración de pines [83].	76
Fig. 3-19. Circuito resistivo para medir de luminosidad del sistema.	77
Fig. 3-20. Luxómetro Hioki LUX Tester 3421.	78
Fig. 3-21. Gráfico de luxes por unidad del ADC obtenidos con el muestreo y el luxómetro.	78
Fig. 3-22. Medidor de TDS y CE portátil [84].	79

Fig. 3-23. Circuito diseñado para medir la conductividad eléctrica de la solución nutritiva.	80
Fig. 3-24. Gráfico de $\mu\text{S/cm}$ por unidad del ADC obtenidos con el muestreo y el medidor de TDS y CE empleado.	81
Fig. 3-25. Módulo de monitoreo y sensor de pH líquido modelo PH-4502C [85].	81
Fig. 3-26. Electrodo sensor de PH con conector tipo BNC [86].	82
Fig. 3-27. Gráfico de valores de pH por unidad de voltaje, obtenidos durante la calibración del sensor de pH utilizado.....	84
Fig. 3-28. Gráfico de relación entre las mediciones en unidades del ADC y su correspondencia con unidades pH reales.	85
Fig. 3-29. Bomba Yosoo zjchao-6935824c de 12v de CD [87].	86
Fig. 3-30. Tira de LED de 5m modelo TMU5SOLA de la marca laser&led [88].	86
Fig. 3-31. Caja receptora de IR y control remoto marca laser&led.	87
Fig. 3-32. Ejemplo de señal PWM con 75% de ciclo de trabajo.....	87
Fig. 3-33. Diagrama de conexión del CI CD4051BCN [90].....	89
Fig. 3-34. Circuito de instrumentación entre la salida del MUX y la entrada del ADC.	90
Fig. 3-35. Gráfica de señales lógicas invertidas de 3.3v, en colores rojo (señal original) y azul (señal invertida con transistor).	91
Fig. 3-36. Diagrama de terminales del transistor BC548B [91].....	91
Fig. 3-37. Circuito de transistor utilizado para obtener una señal lógica invertida.	92
Fig. 3-38. Diagrama de bloques del L293B en un ejemplo de aplicación con motores [92].....	92
Fig. 4-1. Esquema general de la relación entre los elementos del software.	94
Fig. 4-2. Acotaciones de los diagramas de flujo de datos.....	95
Fig. 4-3. Diagrama de flujo de datos de una autenticación fructuosa.....	96
Fig. 4-4. Diagrama de flujo de datos de un cierre de sesión.	97
Fig. 4-5. Diagrama de flujo de datos del proceso de captura de una nueva fotografía.	98
Fig. 4-6. Diagrama de flujo de datos del proceso de solicitud de la última fotografía almacenada en la base de datos.	99
Fig. 4-7. Diagrama de flujo de datos del proceso de recepción e inserción de datos de los sensores en la base de datos.....	99
Fig. 4-8. Diagrama de flujo de datos del proceso de graficación de datos de los sensores.	100
Fig. 4-9. Diagrama de flujo de datos del proceso de obtención del último estado almacenado en la base de datos para un actuador, por solicitud del microcontrolador 1.	101
Fig. 4-10. Diagrama de flujo de datos del proceso de obtención del último estado almacenado en la base de datos de los actuadores, por solicitud de la página web.	102
Fig. 4-11. Diagrama de flujo de datos del proceso de modificación del estado de un actuador.	102
Fig. 4-12. Logotipo de NodeJS [93].	103
Fig. 4-13. Logotipo de npm [95].	104
Fig. 4-14. Logotipo de MongoDB [96].....	104
Fig. 4-15. Diagrama de inserción de documentos a base de datos MongoDB [96].....	105
Fig. 4-16. Diagrama de organización interna de la base de datos implementada.	106
Fig. 4-17. Pantalla principal de ESPlorer.	109

Fig. 4-18. Logotipo de Lua [99].	110
Fig. 4-19. Diagrama de flujo del programa principal del microcontrolador 1.	112
Fig. 4-20. Diagrama de flujo de rutina asíncrona ejecutada ante mensaje MQTT de actualizar estado de actuador.	113
Fig. 4-21. Diagrama de flujo de rutina asíncrona ejecutada ante temporizador.	113
Fig. 4-22. Diagrama de flujo del programa principal del microcontrolador 2.	115
Fig. 4-23. Diagramas de flujo de ejecuciones asíncronas ante instrucciones MQTT. a) Ante instrucción de nueva fotografía. b) Ante instrucción de continuar con la transmisión de la fotografía.	116
Fig. 4-24. Diagrama de flujo de rutina de instrucciones enviadas a la cámara serial para obtener una imagen.	117
Fig. 4-25. Vista de la página de autenticación en un navegador.	118
Fig. 4-26. Barra lateral de navegación de la página web principal.	119
Fig. 4-27. Comportamiento expansivo de respuesta de los íconos de la barra de navegación lateral al cursor. a) ícono Imagen. b) Ícono Gráficos. c) Ícono Actuadores. d) Ícono Cerrar sesión.	120
Fig. 4-28. Contenedor central de la página principal.	121
Fig. 4-29. Barra inferior de opciones de la página principal.	121
Fig. 4-30. Comportamiento de respuesta al cursor, de los íconos de la barra inferior de opciones.	122
Fig. 4-31. Vista <i>Imagen</i> , de la página principal.	122
Fig. 4-32. Vista <i>Gráficos</i> , de la página principal.	123
Fig. 4-33. Vista <i>Actuadores</i> , de la página principal.	124
Fig. 5-1. Convertidor USB-Serial utilizado en el proyecto.	125
Fig. 5-2. Cámara serial conectada al convertidor USB-Serial.	125
Fig. 5-3. Pantalla principal del software LS-Y201.	126
Fig. 5-4. Puerto COM asignado por la computadora, al convertidor USB-Serial.	126
Fig. 5-5. Estableciendo conexión con cámara serial.	126
Fig. 5-6. Visualizando imagen tomada con la cámara serial, en el programa LS-Y201.	127
Fig. 5-7. Introduciendo credenciales a la página de autenticación.	129
Fig. 5-8. Solicitando nueva captura de fotografía desde la vista <i>Imagen</i> de la página principal.	130
Fig. 5-9. Contenedor central que se muestra durante el proceso de captura de la nueva fotografía, en la vista <i>Imagen</i> en la página principal.	131
Fig. 5-10. Contenedor central que se muestra durante el proceso de recepción de la nueva fotografía, en la vista <i>Imagen</i> en la página principal.	131
Fig. 5-11. Visualización de la nueva imagen capturada por la cámara tras la ejecución de la captura solicitada.	132
Fig. 5-12. Procedimiento para cambiar el gráfico en visualización por el de temperatura, a través de la barra inferior de opciones.	133
Fig. 5-13. Gráfico de temperatura, mostrado en la vista <i>Gráficos</i> de la página principal.	133
Fig. 5-14. Ventana emergente de selección de color en formato hexadecimal, para cambiar el tono y brillo de la iluminación LED.	134
Fig. 5-15. Cerrando sesión a través de la barra lateral de navegación.	135
Fig. 0-1. Diagrama de pines del NodeMCU V2 [76].	142

Fig. 0-2. Diagrama de pines del NodeMCU V3 [76]. 143

LISTA DE TABLAS

Tabla 2-1. Sintaxis de solicitud HTTP [63]. 46
Tabla 2-2. Ejemplo de una solicitud HTTP [63]. 46
Tabla 2-3. Sintaxis de respuesta HTTP [63]. 47
Tabla 2-4. Ejemplo de una respuesta HTTP [63]. 47
Tabla 3-1. Tabla de verdad del CI CD4051BCN [90]. 89

ABREVIATURAS Y ACRÓNIMOS INDEXADOS

A

a. C.: antes de Cristo	1, 3
ADC: Analog-to-Digital Converter	20, 66, 75, 77, 78, 80, 81, 82, 84, 90, 111, 123, 128
AES: Advanced Encryption Standard	56
ARP: Address Resolution Protocol	44

B

BASIC: Beginner's All-purpose Symbolic Instruction Code	19
BGP: Border Gateway Protocol	49
BLOB: Binary Large Objects	59
BNC: Bayonet Neill-Concelman	82, 83, 84

C

CAN: Controller Area Network	19
CD: Corriente Directa	70, 73, 74, 82, 85, 86, 92
CI: Circuito Integrado	86, 88, 89, 93
CO ₂ : Dióxido de carbono	4, 10
CPU: Central Processing Unit	18, 56
CSI: Camera Serial Interface	88
CSS: Cascading Stylesheets	III, IV, 117

D

DAC: Digital-to-Analog Converter	20
DNS: Domain Name System	49
DSI: Display Serial Interface	88

E

EEPROM: Electrically Erasable Programmable Read-Only Memory	18, 20
EEUU: Estados Unidos de América	4
EPROM: Erasable Programmable Read-Only Memory	18

F

FEM: Fuerza Electromotriz	23
FTP: File Transfer Protocol	41, 44, 54

G

GPIO: General Purpose Input/Output	66, 67, 88, 90
------------------------------------------	----------------

H

HDMI: High-Definition Multimedia Interface	88
HR: Humedad Relativa	31
HTML: HyperText Markup Language	III, IV, 45, 47, 117
HTTP: Hypertext Transfer Protocol	41, 44, 45, 46, 47, 102, 103, 106, 107, 138, 139

I

I ² C: Inter-Integrated Circuit	19
ICMP: Internet Control Message Protocol	44, 49
IDE: Integrated Development Environment	108, 128
IEEE: Institute of Electrical and Electronics Engineers	20
IGMP: Internet Group Management Protocol	49
IHL: Internet Header Length	48
IoT: Internet of Things	8, 9, 62
IP: Internet Protocol	41, 43, 44, 47, 48, 49, 50, 51, 52, 54
IPv4: Internet Protocol version 4	47, 48, 49, 50
IPv6: Internet Protocol version 6	48, 49, 50
IR: Infrarojo	26
IRC: Internet Relay Chat	41

L

LAN: Local Area Network	40, 88
LDR: Light Dependant Resistor	29, 75, 77

LED: Light Emitting Diode. 35, 37, 38, 62, 85, 86, 87, 88, 93, 111, 124, 134, 136

M

M2M: machine-to-machine56
MAC: Media Access Control 41, 43, 44
MIME: Multipurpose Internet Mail Extensions45
MQTT: Message Queue Telemetry Transport 56, 95, 97, 99, 102, 103, 106, 107, 108, 110, 111, 113, 114, 116, 136
MUX: Multiplexor..... 80, 88, 90

N

NFT: Nutrient Film Technique 11, 15, 136
NNTP: Network News Transport Protocol44
npm: Node Package Manager 104, 106, 108
NTC: Negative Temperature Coefficient 24, 75

O

ORP: Potencial de Oxidación/Reducción.....27
OSPF: Open Shortest Path First.....49

P

P2P: Peer-to-peer.....40
PC: Personal Computer43
pH: Potencial de hidrogenionesIII, IV, 10, 15, 22, 27, 28, 64, 81, 82, 83, 84, 111, 122, 123, 136
PLL: Phase Locked Loops20
PoR: Power-on Reset.....20
ppm: Partes por Millón28
PPP: Point-to-Point Protocol42
PTC: Positive Temperature Coefficient24
PVC: Policloruro de vinilo 4, 16, 62, 63, 64, 128
PWM: Pulse-Width Modulation 79, 80, 87, 88

R

RAM: Random Access Memory... 18, 20, 56, 66, 88, 110

RGB: Red, Green, Blue 38, 62, 85, 86, 124, 134, 136
ROM: Read-Only Memory18, 19, 20
RTD: Resistance Temperature Detector23, 24
RTOS: Real-Time Operating System 66

S

S: Siemens 28
SBC: Single Board Computer21, 22, 88
SD: Secure Digital70, 88
SHA: Standard Hashing Algorithm 56
SLIP: Serial Line Internet Protocol 42
SMTP: Simple Mail Transfer Protocol..... 44
SoC: System on a Chip 20, 21, 62, 63, 65, 66, 67, 74, 108, 110, 127, 128
SPI: Serial Peripheral Interface 20
SQL: Structured Query Language57, 58
SRAM: Static Random Access Memory 128
SSH: Secure Shell54, 55, 56

T

TCP: Transmission Control Protocol ...41, 44, 47, 49, 50, 51, 52, 53, 54
TDS: Total Dissolved Solids79, 80, 81
TTL: Time to Live..... 48; Transistor-Transistor Logic 72

U

UART: Universal Asynchronous Receiver-Transmitter19, 20
UDP: User Datagram Protocol44, 49
USB: Universal Serial Bus..... 20, 21, 68, 69, 88, 125, 126

V

VLAN: Virtual Local Area Network 41

W

106, 107, 110, 117, 118, 119, 122, 123, 124, 125,
128, 135, 136

web: World Wide Web... III, IV, 9, 10, 11, 12, 39, 41, 45,
62, 94, 95, 96, 97, 98, 99, 100, 101, 102, 104, 105,

CAPÍTULO 1. INTRODUCCIÓN

A lo largo de la historia de la humanidad, la agricultura ha jugado un papel fundamental para la supervivencia. Es una de las actividades económicas más antigua, y una de las que seguramente seguirá vigente mientras perdure la raza humana.

Pasando por el cultivo rudimentario, la utilización de herramientas como azadones, la siembra en chinampas, el desarrollo del arado tirado por animales, el empleo de maquinaria como tractores para el cultivo, y la innovación con técnicas de invernadero y técnicas de cultivo alternativas, la agricultura ha tenido una evolución lenta, pero constante, donde los avances tecnológicos y las mega tendencias han ido marcando el ritmo a lo largo de sus poco más de 12000 años de existencia [1].

Una de las mega tendencias que más ha roto paradigmas y ha dado pie a más innovaciones, ha sido la llegada del Internet y el uso de las computadoras. De ahí la importancia de plantear un cambio tecnológico moderno, respecto a la agricultura, aprovechando las enormes prestaciones que el Internet, las computadoras y la automatización de procesos, pueden brindarnos.

A continuación, se presentan los antecedentes históricos de la agricultura y la llegada de los invernaderos como variación al cultivo tradicional, mostrando posteriormente los últimos avances y conceptos similares al propuesto en esta tesis, y exponiendo finalmente los objetivos y la justificación de este trabajo.

1.1. ANTECEDENTES

El periodo neolítico en la historia de la humanidad, puede considerarse como uno de los más importantes ya que trajo consigo grandes cambios sociales. Fue un punto de inflexión en el que la especie humana se desarrolló como nunca antes lo había hecho y se produjeron importantes eventos que dieron pie a que el ser humano abandonara su estilo de vida, basado prácticamente en la supervivencia a través de la caza, la pesca y la recolección, por otro más sedentario y con perspectivas mucho más ambiciosas que el mero hecho de sobrevivir [2].

1.1.1. BREVE HISTORIA DE LA AGRICULTURA

En algún momento cerca del año 8500 a. C., los humanos en el Creciente Fértil (zona que se extiende por lo que hoy es Egipto, Israel y Turquía), comenzaron a sembrar granos en lugar de cosechar los silvestres. Antes de la llegada de la agricultura, los humanos eran nómadas. Con el aumento de esta como fuente de alimento centralizada y predecible, tenían un incentivo para quedarse. Fue así que comenzaron a formar ciudades [3].

Los siguientes 8500 años, la evolución fue lenta. Empíricamente, seleccionando las mejores semillas para sembrarlas, fueron mejorando genéticamente las plantas [3].

La agricultura mexicana en la época prehispánica era una de las más ricas en la historia de la humanidad. Surgida gracias a la domesticación del maíz, la agricultura alimentó a toda la región Mesoamericana con

sus cultivos de maíz, frijol, chile y calabaza, principalmente. Las zonas chinamperas del Valle de México generaban excedentes amplios que forjaron la civilización tolteca, la náhuatl y la tenoxca. Así como el cultivo hidráulico de camellones permitió el surgimiento del imperio maya clásico. Al llegar el conquistador español, encontró un sistema agrícola ajeno a su realidad ecológica que no comprendió. Por esta razón lo destruyó en gran medida, aunque también aportó la tradición ibérica que se aplicó a grandes zonas de México [4].

Los españoles introdujeron en América animales desconocidos: caballos, burros, vacas, ovejas y otros. También trajeron nuevas técnicas. Para sembrar se comenzó a utilizar el arado, aunque continuó el uso de la coa. El trabajo de los metales permitió la fabricación de herramientas metálicas hasta entonces inexistentes en América, como el serrucho, la gubia, el formón, el martillo, el hacha [5].

Durante la Edad de Bronce y la Edad de Hierro, las herramientas de piedra y madera rudimentarias, se reemplazaron con unas de metal más eficientes. Para ese entonces, la agricultura era una actividad económica que englobaba a cerca del 80% de la población mundial [3].

Fue entre los siglos XVIII y XIX que la innovación agrícola se disparó. El diseño del arado mejoró y el inglés Jethro Tull inventó la primera máquina de sembrar del mundo (véase Fig. 1-1): un aparato que permitía sembrar las semillas en hileras rectas y ordenadas. Poco después siguió el equipo mecánico para cosechar tirado por caballos, como la segadora de Cyrus McCormick (véase Fig. 1-2). La productividad de la agricultura se disparó [3].

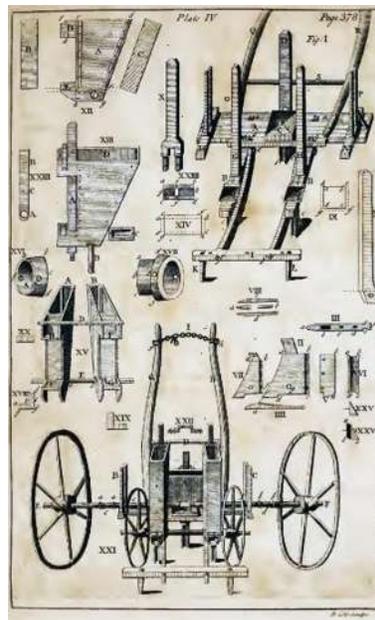


Fig. 1-1. Primera sembradora del mundo, construida por Jethro Tull [6].

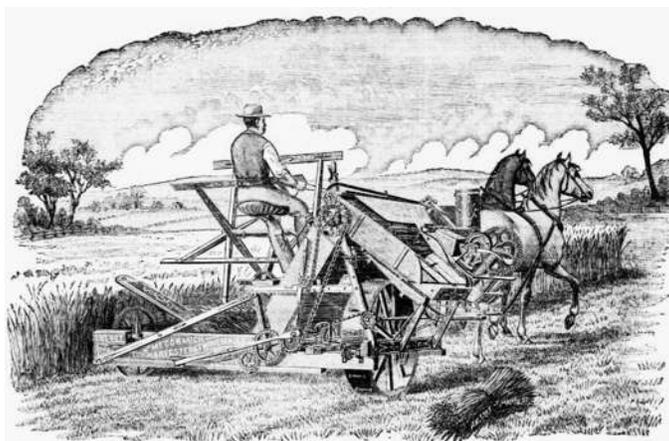


Fig. 1-2. Máquina cosechadora mecánica tirada por caballos, inventada por Cyrus McCormick [7].

Durante el siglo XX, las máquinas a gasolina comenzaron a reemplazar al equipo tradicional tirado por animales y al que usaba motores a vapor. Combinado con adelantos tecnológicos en fertilizantes y pesticidas, después de la Segunda Guerra Mundial, este cambio permitió otro salto en la productividad agrícola y trajo consigo una modificación importante en el balance de la fuerza laboral. En Estados Unidos, el porcentaje de la fuerza trabajadora dedicada a la agricultura pasó del 40% (en 1900), a sólo 2% (en 2000) [3].

Entre 1900 y 2012, la población mundial creció de 1600 millones a más de 7000 millones. En 1700, sólo el 7% de la superficie terrestre se usaba para agricultura. Hoy en día, es más del 40%, y solamente una porción del terreno que queda es apto para el cultivo [3].

1.1.2. INVERNADEROS

La Real Academia de la Lengua Española [8], define invernadero como “Recinto en el que se mantienen condiciones ambientales adecuadas para favorecer el cultivo de plantas”.

Aunque en la antigüedad no existían invernaderos como los conocidos hoy en día, sí se conocían diferentes prácticas para cultivar hortalizas y flores fuera de temporada. En Atenas y Roma, entre los años 327 y 287 a. C., las plantas se movían hacia patios cubiertos durante la noche y se calentaba el suelo mediante estiércol o composta; Agricultores en China, Egipto, Israel, Grecia y Roma cultivaron plantas en macetas que trasladaban durante la noche o periodos fríos a zonas protegidas [9].

Los primeros pasos hacia los invernaderos fueron la construcción y el uso de camas móviles con plantas. En el siglo XV, en el norte de Italia y el sur de Alemania, llegó a ser muy común el uso de cabañas para la protección en invierno, existiendo unas pocas hectáreas [9].

En el siglo XVII, los “orangeries”, como el mostrado en Fig. 1-3, llegaron a estar de moda en Europa, (siendo el primero del siglo XVI en Inglaterra) llegando a reemplazar gradualmente a las cabañas de invierno. La planta favorita para cultivar en ellos, era el naranjo. Tenían unas amplias ventanas de cristal en lado sur para permitir la entrada de luz hacia las plantas. Para calentarlos en invierno se usaban unas pocas estufas. Los orangeries fueron los precursores de los invernaderos. A partir de 1737 se empezó a

tener un gran interés por ahorrar energía en los invernaderos mediante cortinas, entre otros elementos. Se comenzó a cultivar con aporte de CO₂, calor y nutrientes procedente de estiércol [9].



Fig. 1-3. Orangerie ubicado en el Real Jardín Botánico de Kew, Londres, Inglaterra [10].

La industrialización en el siglo XIX trajo un rápido desarrollo de la tecnología de invernaderos. A principios de este siglo la mayoría de invernaderos eran de construcción sencilla con una pendiente en cubierta de 45° siendo calentado el invernadero por la descomposición de materia orgánica o mediante estufas. En 1829 fue mencionado el doble acristalamiento como aislante térmico [9].

En el siglo XX se desarrolló una amplia información sobre calefacción, riego y fertilización en invernadero. En Holanda se desarrollaron gradualmente invernaderos para un mejor uso de los cultivos, hasta que en 1937 se construyó el invernadero Venlo a base de acero y cristal, el cual podía ser utilizado para diferentes cultivos [9].

Es a partir de la Primera Guerra Mundial cuando aparecen los plásticos y con ellos su uso en la agricultura (materiales a base de polietileno, poliestireno o PVC). A partir de la Segunda Guerra Mundial aparecieron los plásticos rígidos como el poliéster en EEUU y Europa [9].

Las estructuras de invernadero pueden ser sencillas, tipo artesanal (hechos con material y conocimiento locales) o más complejas, tipo industrial (altamente mecanizados y equipados) [9].

Las estructuras de invernaderos se agrupan acorde a [11]:

- Características constructivas: geometría, pendiente de cubierta y orientación.
- Material de cubierta: vidrio, plástico rígido, plástico flexible o combinaciones de ellos.
- Material de construcción: acero, aluminio, madera o combinaciones de ellos.

El tipo de material que conforma la estructura condiciona la geometría y la altura del invernadero ya que debe de soportar diferentes cargas (viento, nieve, peso de la estructura, peso de los equipos, etcétera). Los materiales de estructura más habituales que están presentes en los invernaderos son la madera, el acero o el aluminio [9].

Además de los materiales utilizados para su construcción, la colocación de una pantalla o protección junto a la planta altera las condiciones ambientales que afectan a toda la planta o a una parte de ella. La posición de la pantalla o protección, respecto a la planta, determina el tipo de protección. Cuando la pantalla se coloca sobre el suelo y debajo de los órganos aéreos de la planta, tenemos el acolchado (véase Fig. 1-4). Los cortavientos constituyen pantallas o protecciones laterales (véase Fig. 1-5) [12].



Fig. 1-4. Acolchado negro en cultivo de piña [13]



Fig. 1-5. Cortavientos semipermeable [12]

Cuando las pantallas se colocan sobre las plantas, a modo de cubierta, tenemos un tercer tipo de protecciones: invernaderos, túneles y cubiertas flotantes. En estas últimas, la protección está directamente apoyada sobre las plantas, sin estructura que las soporte (véase Fig. 1-6) [12].



Fig. 1-6. Colocación mecanizada de cubierta flotante de material agrotéxtil [12].

Los túneles bajos son pequeñas estructuras que soportan la pantalla que provee protección temporal al cultivo, con una altura que no excede un metro de altura. En ellos, las prácticas de cultivo se efectúan desde el exterior [12]. Ver Fig. 1-7.



Fig. 1-7. Túneles bajos [12].

Los túneles altos o macro túneles (como el mostrado en Fig. 1-8), son aquellos que permiten el paso de una persona por su interior, y el cultivo de especies de altura diversa, incluso árboles frutales (véase Fig. 1-9). La distinción entre macro túneles e invernaderos, sin embargo, está poco definida [12].



Fig. 1-8. Túneles altos [12].



Fig. 1-9. Invernadero multitúnel de lámina plástica [12].

En México, las primeras estructuras tipo invernaderos llegaron en la década de 1950, cuando un productor del entonces Distrito Federal, las instaló para la producción de plantas ornamentales y flores para decorar los camellones, glorietas y áreas verdes de la metrópoli. Para 1970 la floricultura se convirtió en una industria de peso para el centro del país. A mediados de los 80's, se instalaron invernaderos para la producción de plántula en estados del norte del país. Fue hasta finales de dicha década y principios de 1990 cuando tecnologías, principalmente europeas, fueron instaladas en el país para la producción de ciclos completos de hortalizas. Fue en el occidente y noroeste de México donde inició la producción de tomate en condiciones de ambiente controlado [14].

1.2. ESTADO DEL ARTE

Los avances tecnológicos modernos como el Internet de las Cosas (IoT por sus siglas en inglés) y los sistemas electrónicos programables, han dado pie a desarrollos urbanos agrícolas para interiores, que buscan que cada familia pueda cultivar sus propios alimentos desde sus hogares, empleando el menor esfuerzo posible.

1.2.1. MINIFARM

El producto *MiniFarm* de la empresa *Replantable* fue fundada en 2015 por Ruwan Subasinghe y Alex Weiss después de que se dieron cuenta de los defectos del sistema industrializado de alimentos. Ellos querían, que hacer crecer su propia comida fuera tan fácil como prepararse una taza de café, y con ese propósito, desarrollaron su producto (mostrado en Fig. 1-10) [15].



Fig. 1-10. Diseño modular MiniFarm de Replantable [15].

MiniFarm está compuesto por un armario de cultivo, una bandeja de agua, una plataforma de planta y luz de cosecha, como se aprecia en Fig. 1-11. El gabinete incorpora tecnología para controlarlo de manera sencilla: tiene un botón de inicio y uno de cosecha, así como una opción para seleccionar las semanas requeridas para el crecimiento, dependiendo del tipo de cultivo [15].



Fig. 1-11. Estructura del sistema Replantable [15].

Se trata de un sistema hidropónico de raíz flotante donde el usuario introduce a la máquina una película hecha de pulpa reciclada (véase Fig. 1-12) que contiene las semillas y los nutrientes necesarios para el desarrollo de la planta, y añade el nivel de agua necesario para el cultivo. Posteriormente basta con encender el dispositivo, programar el tiempo de cosecha, esperar y cosechar [15].



Fig. 1-12. Película de pulpa reciclada con semillas y nutrientes [15].

1.2.2. SMARTCULTIVA

Haciendo uso del IoT, la empresa *Smartcultiva* comercializa un conjunto de sensores para monitorear y analizar los datos más relevantes de los cultivos hidropónicos empleando una aplicación móvil y una aplicación web ¹(ver Fig. 1-13); sus sistemas permiten controlar actuadores, cuentan con una plataforma

¹ Del inglés de *red*. Se dice de algo que pertenece a la Red Informática Mundial, el sistema de distribución de documentos de hipertexto o hipermedia interconectados y accesibles a través de Internet.

en la nube y permiten la interacción con redes sociales como Twitter, desde la cual pueden controlar el sistema o twittear información [16].



Fig. 1-13. Aplicación y plataforma web de Smartcultiva para el monitoreo y control de variables de cultivo [16].

Sus dispositivos permiten el monitoreo y control de la temperatura, humedad del ambiente, intensidad luminosa de interiores y exteriores, CO₂ en el aire, humedad del suelo, pH, temperatura y niveles del agua, entre otras [16]. Véase Fig. 1-14.

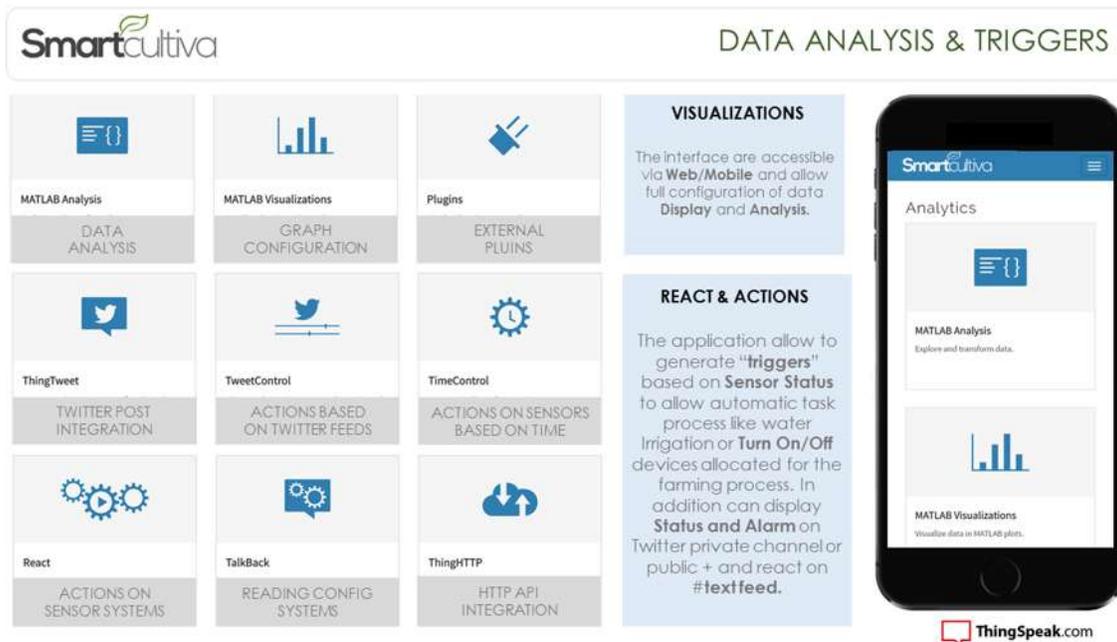


Fig. 1-14. Prestaciones de los sistemas y productos Smartcultiva para el análisis de datos [16].

1.3. JUSTIFICACIÓN

Si bien casi 800 millones de personas padecen hambre en la actualidad, en el 2050 la producción mundial de alimentos deberá aumentar en un 50% para alimentar a los más de nueve mil millones de personas que se prevé vivirán en nuestro planeta. Para lograrlo, es indispensable un cambio transformador en la forma en que producimos los alimentos [17].

Se estima que hay 3600 millones de personas (casi la mitad de la población mundial) que viven en áreas con riesgo de sufrir escasez de agua al menos un mes al año, y esta población podría llegar a alcanzar entre 4800 y 5700 millones en 2050 [17].

Atendiendo a la creciente necesidad de alimentos saludables y confiables, a través de una mejor eficiencia en el uso de los recursos, este trabajo plantea una alternativa para que los usuarios puedan contar con un sistema de producción de frutas y verduras desde sus hogares u oficinas, brindándoles también la posibilidad de observar de cerca el proceso de crecimiento de las mismas.

Teniendo en cuenta la problemática mundial y las proyecciones futuras sobre escasez de agua, se utiliza en este proyecto la técnica de cultivo hidropónico NFT², que permite ahorrar hasta 7/8 partes del agua utilizada, en comparación con la técnica de cultivo tradicional.

Empleando un microcontrolador y un servidor web, pueden automatizarse algunas de las tareas de cultivo; puede llevarse un registro detallado del progreso y se pueden hacer cambios a los actuadores del sistema de manera remota. Todo esto, con el propósito de que el usuario no requiera estar monitoreando de manera presencial sus cultivos.

El sistema en conjunto, considera variables físicas indispensables para el cultivo y desarrollo de las plantas a lo largo de sus ciclos de producción y muestra imágenes de los cultivos al usuario por si requiere tomar decisión o acción alguna, con el fin de que cuente con el control e información necesaria en todo momento para obtener cosechas óptimas.

² Técnica de Película Nutritiva, por sus siglas en inglés.

1.4. OBJETIVOS

1.4.1. OBJETIVO GENERAL

Desarrollar un sistema que permita cultivar frutas y verduras en interiores, de manera semiautomática y sustentable, con monitoreo y control de manera remota, que permita disminuir el consumo de agua al cultivar y que sea una alternativa en la producción de alimentos, apoyando la economía del usuario y brindando la experiencia de producir sus propios alimentos con el uso de nuevas tecnologías.

1.4.2. OBJETIVOS PARTICULARES

- Crear un sistema moderno de cultivo hidropónico³.
- Implementar un sistema para cultivos en interiores.
- Permitir el monitoreo remoto de los parámetros físicos más importantes del sistema, relacionados a los cultivos.
- Diseñar un sistema con cabida⁴ para al menos 2 cultivos diferentes.
- Permitir el control y programación de tareas en los actuadores del sistema.
- Hacer posible la gestión y monitoreo del sistema mediante un navegador web.
- Diseñar un sistema que requiera una mínima participación del usuario pero que, a la vez, permita la interacción de éste con el desarrollo de sus cultivos.

1.5. METODOLOGÍA

Se llevó a cabo el presente trabajo realizando investigación en recursos en línea: artículos web, hojas de datos, documentación sobre protocolos, recursos multimedia, manuales de agricultura y libros.

Se desarrolló posteriormente el hardware necesario y el software de control, considerando las características de diseño requeridas para cumplir los objetivos.

1.6. DESCRIPCIÓN DE CAPÍTULOS

El *Capítulo 2. Marco teórico*, presenta una introducción histórica al fundamento de la tesis: la agricultura. Continúa hablando de los invernaderos como el desarrollo que vino a cambiar el paradigma de la forma de cultivo tradicional y que dio pie a la modernización, y finaliza mostrando los desarrollos y conceptos más modernos, disponibles en el mercado, que combinan la agricultura con tecnologías contemporáneas como el Internet, la automatización, el control y el uso de aparatos electrónicos. A su vez, expone la teoría general de los elementos que conforman el proyecto. Habla también de las tecnologías involucradas en él, así como del principio de funcionamiento de los elementos que lo integran.

³ Cultivo sin tierra.

⁴ Espacio o capacidad que tiene una cosa para contener otra.

El *Capítulo 3. Diseño y descripción del hardware*, habla del proceso de diseño del prototipo creado para la tesis, y describe en términos específicos, las características de los elementos físicos empleados.

El *Capítulo 4. Diseño y descripción del software*, hace su parte describiendo el proceso de diseño del software creado para sumarse al hardware del proyecto. A su vez, describe la tecnología y protocolos utilizados.

Finalmente, *el Capítulo 5. Pruebas y resultados*, concluye conjuntando las pruebas realizadas al prototipo terminado, así como los resultados obtenidos de esas pruebas. Explica también a detalle cómo manipular el sistema, desde el punto de vista del usuario final.

CAPÍTULO 2. MARCO TEÓRICO

2.1. INTRODUCCIÓN

En este capítulo se aborda la teoría necesaria para comprender el funcionamiento del sistema desarrollado. Se exponen métodos de cultivo alternativos y se explica qué son los microcontroladores, sensores, actuadores, periféricos, redes de computadoras y protocolos de comunicación.

2.2. MÉTODOS DE CULTIVO ALTERNATIVOS

Los métodos de cultivo alternativos surgen como respuesta ante problemáticas sociales o ambientales, asociadas al cultivo tradicional. Algunos de ellos buscan producir alimentos fuera de temporada o en climas no propios para su producción tradicional, otros buscan incrementar la producción o reducir costos, unos más tienen como propósito lograr un mejor aprovechamiento de los recursos utilizados en la cosecha, y otros buscan reducir la huella ecológica y sustituir la agricultura tradicional por alternativas sustentables.

Puede considerarse que la problemática más urgente que podría combatirse utilizando métodos de cultivo alternativos es la escasez de agua. Misma de la que se habla ya en medios informativos, debido a su imperante urgencia:

Desde el pasado 1 de enero los más de cuatro millones de habitantes de Ciudad del Cabo, Sudáfrica, han tenido que aprender a reducir drásticamente su consumo de agua para evitar el “Día Cero”, la fecha en que las autoridades locales debían cortar el suministro de agua de la ciudad debido a la falta de reservas. Gracias a las mil y una soluciones ideadas por los ciudadanos para no quedarse sin agua y a la llegada de las esperadas lluvias, el temido ‘Día Cero’ ha quedado aplazado hasta el año 2019. De este modo, Ciudad del Cabo se ha salvado, al menos de momento, de convertirse en la primera gran urbe del mundo en quedarse sin este preciado bien. Más allá de restringir el uso de agua, también a la agricultura de regadío, las autoridades locales han buscado soluciones para diversificar sus fuentes de cara al futuro y no ser tan dependientes del actual sistema de presas. La ciudad está invirtiendo en proyectos basados en agua subterránea, reutilización del agua y desalinización, entre otros [18].

2.2.1. AGRICULTURA BIODINÁMICA

En 1924 un grupo de agricultores alemanes notaron que sus tierras perdían fertilidad y los suelos se encontraban desgastados. Ante el problema solicitaron alternativas a la academia de su país, por lo que se llevaron a cabo un conjunto de conferencias lideradas por Rudolf Steiner, padre de la agricultura biodinámica, la cual, concibe a las granjas como organismos complejos cuya dinámica natural incluye a los animales y al hombre [19].

La agricultura biodinámica se basa en nueve tipos de preparados inventados por Steiner, que a la fecha han comprobado tener efectos en la estructura del suelo y en los microorganismos, mejorando la fertilidad del suelo e incrementando su biodiversidad. Este revolucionario pensador creía que estos preparados transferían poderes sobrenaturales terrestres y fuerzas cósmicas al suelo. La rareza de estos es innegable, se dividían en dos tipos: los preparados para el acondicionamiento del suelo y los utilizados para la preparación y aplicación de composta [19].

Este tipo de siembra toma en cuenta un calendario de cultivo basado en el movimiento de los astros. Para la preparación del campo por ejemplo Steiner propuso el llamado cuerno de abonar, basado en una mezcla de tierra negra preparada al llenar el cuerno de una vaca que se entierra en la tierra en otoño para dejarlo descomponer durante el invierno y recuperarse para su uso la siguiente primavera. En este cuerno se añade además cuarzo molido en polvo preparado [19].

2.2.2. ACUAPONIA

La acuaponía es un sistema de producción cerrado que integra la técnica de la acuicultura con la hidroponía, es decir, es una combinación de la producción de peces y la producción de hortalizas sin suelo por el medio común “agua”. Las plantas y los peces crean una sinergia, ya que los desechos metabólicos de los peces son aprovechados como nutrientes por los vegetales para crecer, mientras que las plantas limpian el agua y eliminan los compuestos tóxicos para los peces (principalmente amonio y nitritos), reduciendo la frecuencia de renovación del agua. Sin embargo, en este sistema también intervienen microorganismos que inciden en los procesos de mineralización y nitrificación; principalmente bacterias nitrificantes. Este sistema de producción intensiva sustentable requiere de condiciones ideales para que exista interacción entre peces, microorganismos y plantas [20].

Deben en esta, considerarse parámetros como la oxigenación, el pH y la dureza del agua, tener los tanques apropiados y suficientes para contener a los peces, la temperatura del agua y el ambiente, elegir un sistema hidropónico (camas de sustrato, NFT, balsas flotantes), y la especie de cultivos y peces; pudiendo estos ser comúnmente albahacas, lechugas, pepinos, morrones, tomates, brócoli, para el caso de los primeros, y carpa común, tilapia del Nilo, pez gato y trucha arcoíris, para los segundos [20].

Dado que el mayor ingreso es proporcionado por el aprovechamiento de las plantas (66-90 %), ya que su ciclo de cosecha es más corto, se debe mantener el equilibrio entre los individuos dentro del sistema. La relación puede variar por muchos factores. Los más importantes son [20]:

- Capacidad del sistema
- Método de producción (NFT, recirculante, cama de sustrato)
- Pez (nivel de actividad, tamaño)
- Alimentación del pez (requerimientos)
- Tipo de planta (fruto u hoja)
- Ambiente y condiciones de agua
- Métodos de filtración

Los desbalances más comunes son, como se muestra en Fig. 2-1:

- a. Cuando existe exceso de plantas y pocos peces: el alimento transformado no es suficiente para cubrir la demanda nutricional de las plantas.
- b. Cuando hay más peces que plantas: el alimento transformado en nutrientes es más de lo que las plantas pueden consumir y se eleva la cantidad de nitrato.
- c. Cuando los peces exceden la capacidad del biofiltro: además de nitrato, se concentra el amonio que es perjudicial para los peces.

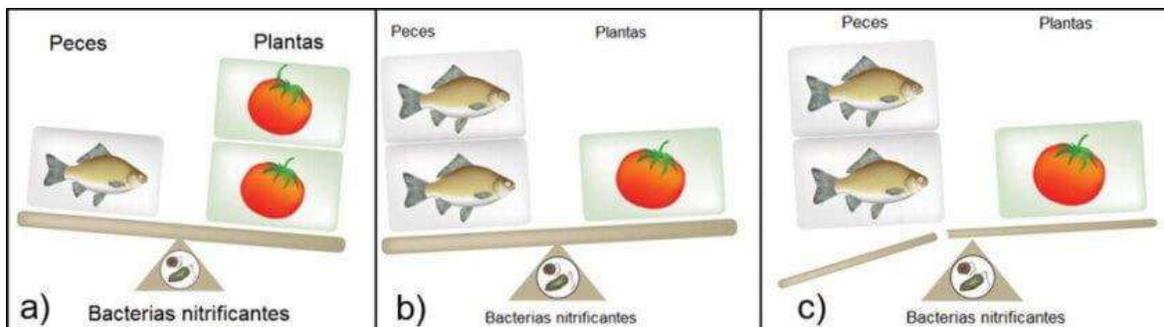


Fig. 2-1. Desbalances más comunes en la acuaponía [20].

2.2.3. AEROPONIA

La aeroponía o aeroponía es un sistema de cultivo en el que las raíces están suspendidas en el aire dentro de un contenedor y son constantemente rociadas con una nube de solución nutritiva por aspersores, el principal medio de soporte para éste cultivo son los tubos de PVC donde las plantas pueden desarrollarse [21].

El término “Aeroponía” proviene del griego *aero* (aire) y *ponos* (trabajo) y fue desarrollada en el año de 1920, el primero en usar ésta técnica fue el doctor Franco Massantini quién implementó por primera vez las columnas de cultivo. El método aeropónico está considerado como un sistema seguro y ecológico, ya que produce cosechas de forma natural sin el uso de químicos, asimismo ayuda a la conservación de los recursos como el agua, la luz y el suelo. Ésta técnica mantiene lejos a los cultivos de enfermedades y plagas, aparte provee a las plantas de todo lo necesario para su crecimiento [21].

Este sistema reduce los riesgos sanitarios por salinidad de agua, mantiene el sistema totalmente aireado, permite realizar rotación de cultivo más rápido y con más higiene, se adapta a cualquier cultivo, puede producirse en pequeña o gran escala, no genera algas, consume muy pocos nutrientes, es sencillo y fácil de manipular [21]. Véase Fig. 2-2.

Hay distintos tipos de cultivos aeropónicos, entre los principales se encuentra el cultivo por esquejes, el sistema re-circulante de baja y alta presión y por goteros de alta presión [21].



Fig. 2-2. Plantas cultivadas con aeroponía, y sus largas raíces [21].

2.2.4. HIDROPONIA

La hidroponía, hidroponia o agricultura hidropónica es un método utilizado para cultivar plantas usando soluciones minerales en vez de suelo agrícola. La palabra proviene del griego *hidro*, que significa agua, y *ponos*, que significa labor o trabajo; significando por tanto, trabajo en agua [22].

El cultivo hidropónico surge primero que el cultivo en tierra. Muchos creen que empezó en la antigua Babilonia en los famosos Jardines Colgantes (véase Fig. 2-3), que se listan como una de las Siete Maravillas del Mundo Antiguo, en lo que probablemente fuera uno de los primeros intentos exitosos de cultivar plantas hidropónicamente [23].



Fig. 2-3. Pintura idealizada del siglo XIX de los Jardines Colgantes de Babilonia [24].

Los principios son encontrados en China, Egipto e India. En el año 1600 el belga Helmont realizó experimentos que demuestran la obtención de nutrientes por parte de las plantas. Pero fue hasta 1699

cuando Woodward demostró finalmente cómo las plantas obtenían alimentos. Posteriormente en 1860, los alemanes Sachs y Knop fueron los primeros en hacer crecer las plantas en una solución nutritiva, llamando al proceso “nutriculture” [23].

La hidroponía es tan antigua como la civilización humana, no obstante, el término fue acuñado hasta 1929 por el profesor de la Universidad de California, William F. Gricke. Durante la Segunda Guerra Mundial, las fuerzas aliadas instalan en sus bases sistemas hidropónicos para proveer de vegetales y frutas frescas a las tropas en conflicto. A partir de esto, la hidroponía comercial se extiende en el mundo [23].

2.3. MICROCONTROLADORES

Un microcontrolador es un circuito integrado que en su interior contiene una unidad central de procesamiento (CPU), unidades de memoria (RAM y ROM), puertos de entrada y salida y periféricos. Estas partes están interconectadas dentro del microcontrolador, como se aprecia en Fig. 2-4, y en conjunto forman lo que se le conoce como microcomputadora. Se puede decir que un microcontrolador es una microcomputadora completa encapsulada en un circuito integrado [25].

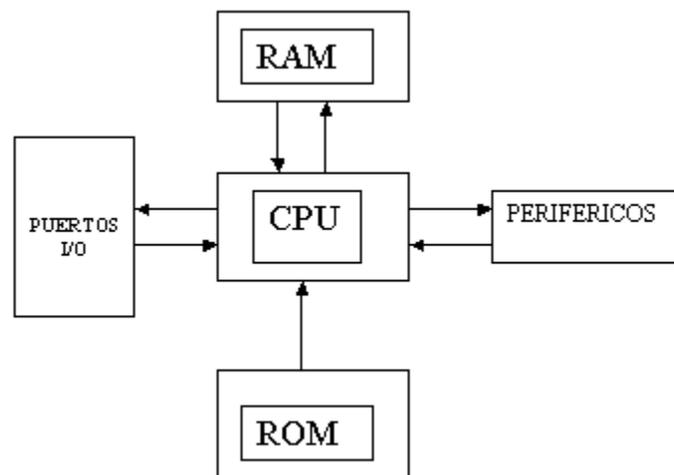


Fig. 2-4. Diagrama a bloques de un microcontrolador [25].

Un microcontrolador típico posee en su interior un generador de reloj integrado, una pequeña cantidad de memoria RAM⁵ y ROM⁶/EPROM⁷/EEPROM⁸. Para hacerlos funcionar todo lo que se necesita son unos pocos programas de control, alimentación (típicamente 3.3V o 5V) y un cristal de sincronización [26].

⁵ Memoria de Acceso Aleatorio, por sus siglas en inglés.

⁶ Memoria de Sólo Lectura, por sus siglas en inglés.

⁷ ROM Programable Borrable, por sus siglas en inglés.

⁸ ROM Programable y Borrable, por sus siglas en inglés.

Los microcontroladores disponen generalmente también de una gran variedad de dispositivos de entrada/salida, como convertidores analógico-digitales, temporizadores, UART⁹ y buses de interfaz serie especializados, como I2C y CAN [26].

Los microcontroladores no tienen un sistema operativo; En su lugar, se les carga un programa para controlar los periféricos, almacenar y procesar los datos, utilizar temporizadores, entre muchas acciones más, que dependerán del modelo del controlador en cuestión. Este se almacena normalmente en la memoria ROM. Sin un programa, los microcontroladores carecen de utilidad.

El carácter programable de los microcontroladores simplifica el diseño de circuitos electrónicos. Permiten modularidad y flexibilidad, ya que un mismo circuito se puede utilizar para realizar diferentes funciones con sólo cambiar el programa del microcontrolador [25].

Es común encontrar microcontroladores en campos como la robótica y el automatismo, en la industria del entretenimiento, en las telecomunicaciones, en la instrumentación, en el hogar, en la industria automotriz, etc [25].

Debido a que los microcontroladores interpretan y procesan datos con instrucciones en lenguaje máquina, la programación comúnmente se lleva a cabo en un lenguaje de alto nivel, es decir, un lenguaje que utilice frases o palabras semejantes o propias del lenguaje humano. Las sentencias de los lenguajes de alto nivel facilitan enormemente la programación ya que son familiares a nuestra manera de comunicarnos. Lenguajes como el C o BASIC son comúnmente utilizados en la programación de microcontroladores [25].

Otro tipo de lenguaje más especializado es el lenguaje ensamblador. El lenguaje ensamblador es una lista con un limitado número de instrucciones a los cuales puede responder un microcontrolador. Estas instrucciones son palabras o abreviaciones que representan las instrucciones en lenguaje máquina del microcontrolador y permiten operar directamente con los registros de memoria, así como con las instrucciones intrínsecas del microcontrolador. Es por esto que el lenguaje ensamblador es sin lugar a dudas el lenguaje por excelencia en la programación de microcontroladores, ya que permite hacer un uso eficiente de la memoria y minimizar el tiempo de ejecución de un programa [25].

Los microcontroladores más utilizados son los AVR, ARM, MSP430 y PIC [26].

2.3.1. COMPILACIÓN

Todo programa escrito en un lenguaje de alto nivel debe ser transformado en código máquina. Un software de computadora, llamado compilador, traduce y transforma el programa en código máquina, que es lo que realmente puede leer e interpretar el microcontrolador. Un diagrama a bloques de este proceso se observa en Fig. 2-5. [25].

⁹ Transmisor-Receptor Asíncrono Universal, por sus siglas en inglés.

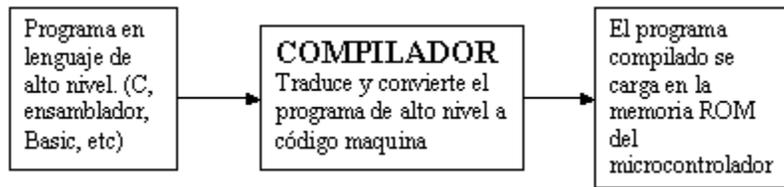


Fig. 2-5. Diagrama a bloques del funcionamiento del compilador [25].

2.3.2. SYSTEM ON A CHIP (SOC)

Un SoC (Sistema en un Chip, por sus siglas en inglés) busca la integración de todos los módulos de un sistema en un solo chip o circuito lo cual reduce en grandes proporciones el tamaño del dispositivo que lo incluye [27].

Un SoC estándar está constituido por [27]:

- Un microcontrolador, microprocesador o DSP.
- Memoria informática de tipo ROM, RAM, EEPROM o Flash. Incluyendo algunos o todos estos.
- Generadores de frecuencia fija; como osciladores o Lazos de Seguimiento de Fase (PLL por sus siglas en inglés).
- Componentes periféricos como contadores,-temporizadores, temporizadores/relojes a tiempo real o generadores Power-on Reset (PoR).
- Controladores de comunicación con interfaces externas con estándares como USB¹⁰, IEEE¹¹, Ethernet, UART, SPI¹², etc.
- Controladores de interfaces analógicas como conversores ADC¹³ y DAC¹⁴.
- Reguladores de voltaje y circuitos de gestión eficaz de la energía.

2.3.3. PLACA DE DESARROLLO

Las placas, tarjetas o kits de desarrollo son sistemas que llevan incorporados un SoC que, a su vez, dentro tiene un microcontrolador o unidad mínima de procesamiento. El esquema general de este tipo de placas se ve en Fig. 2-6 [28].

El objetivo principal de las placas de desarrollo, suele ser permitir programar el microcontrolador a través de sí, contando con material y herramientas extra para facilitar la implementación de proyectos con microcontroladores o similares [28].

¹⁰ Bus Universal en Serie, por sus siglas en inglés.

¹¹ Instituto de Ingeniería Eléctrica y Electrónica, por sus siglas en inglés.

¹² Interfaz Periférica Serial, por sus siglas en inglés.

¹³ Convertidor Analógico-Digital, por sus siglas en inglés.

¹⁴ Convertidor Digital-Analógico, por sus siglas en inglés.

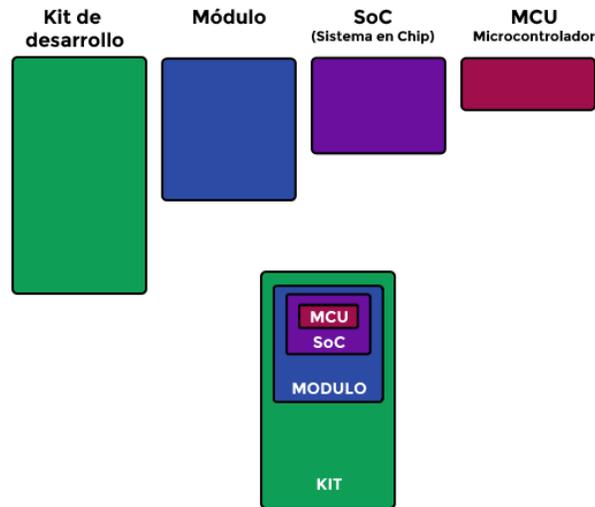


Fig. 2-6. Estructura general de una placa de desarrollo [28].

Los SoC, pueden a menudo estar integrados dentro de un Módulo. Este incluirá componentes que podrían no encontrarse presentes dentro del SoC, o que lo complementan de alguna manera [28].

2.4. ORDENADOR DE PLACA ÚNICA (SBC)

Un Ordenador de Placa Única (SBC por sus siglas en inglés), es una placa que contiene todos o la mayor parte de los componentes de un ordenador [29].

La principal característica de los SBC u ordenadores con placas SBC son sus reducidas dimensiones. Aproximadamente, estos minipc's están montados sobre placas con medidas compactas, desde tamaños relativos a un USB hasta medidas similares a una tarjeta de crédito como la Raspberry Pi (véase Fig. 2-7) que mide 8,5 x 5,3 cm [29].



Fig. 2-7. Ordenador de Placa Única (SBC) marca Raspberry Pi modelo 3B+ [30].

Otra de las características de las placas SBC es su precio. Por lo general las placas SBC son muy económicas, tanto que algunos proyectos que usan estas placas son más económicos que su equivalente estándar [29].

Una tercera característica de las placas SBC es que ofrecen poca potencia, relativamente, ya que actualmente todas las placas SBC ofrecen una potencia más que suficiente para el mundo de la ofimática, del desarrollo e incluso del mundo multimedia [29].

2.5. SENSORES

Los sensores son una parte muy importante para la instrumentación y el control de los procesos. Se utilizan para determinar el estado del proceso donde están instalados. Transforman las variaciones de la magnitud a medir en una señal eléctrica acondicionada de tal manera que pueda ser recibida en su destino [31].

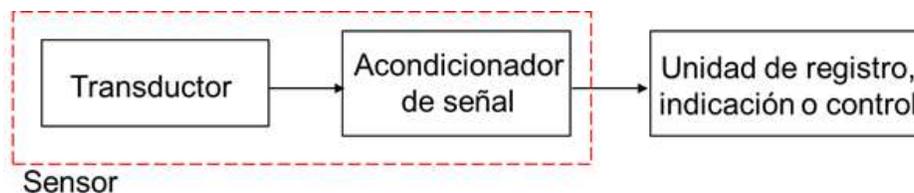


Fig. 2-8. Diagrama de bloques de un sensor [31].

La señal de salida de un sensor por lo general va a un indicador, a un registrador o a un controlador (ver Fig. 2-8). Existen una gran variedad de sensores según la variable que se quiera medir, por ejemplo: presión, temperatura, nivel, flujo, posición (proximidad), velocidad, peso, voltaje, corriente, frecuencia, viscosidad, resistividad, radiación, pH, conductividad eléctrica, humedad entre otras [31].

Muchas de las características de los sensores dependen de la variable a medir, pero otras son comunes a todos los sensores. Algunos de los aspectos a tener en cuenta en el momento de seleccionar un sensor son los siguientes [31]:

- Exactitud: Especifica la diferencia entre el valor medido y el valor real de la variable que se está midiendo.
- Conformidad o repetitividad: El grado con que mediciones sucesivas difieren unas de las otras.
- Resolución: Es el cambio más pequeño que se puede medir.
- Precisión: Se compone de las características de conformidad y resolución.
- Sensibilidad: Viene dado por el mínimo valor de la variable medida que produce un cambio en la salida.
- Error: Es la desviación entre valor verdadero y valor medido.
- Linealidad: Indica que tan cerca está la correlación entre la entrada y la salida a una línea recta.
- Rango: Es la diferencia entre el mayor valor y el menor valor que se puede medir.
- Rapidez de respuesta: Capacidad del instrumento de seguir las variaciones de la entrada.

2.5.1. SENSORES DE TEMPERATURA

Existen diferentes tipos de sensores de temperatura. Todos ellos infieren la temperatura al detectar algún cambio en una característica física. Hay seis tipos de sensor de temperatura [32]:

2.5.1.1. SENSOR DE TEMPERATURA TERMOPAR

Los termopares (como los mostrados en Fig. 2-9) consisten esencialmente en dos tiras o alambres hechos de metales diferentes y unidos en un extremo. Los cambios en la temperatura en esa junta inducen un cambio en la fuerza electromotriz (FEM) entre los otros extremos. A medida que la temperatura sube, esta FEM de salida del termopar aumenta, aunque no necesariamente en forma lineal [32].



Fig. 2-9. Cable termopar desnudo y con hebra cerámica [32].

2.5.1.2. SENSOR DE TEMPERATURA POR RESISTENCIA (RTD)

Los dispositivos termométricos de resistencia, como el mostrado en Fig. 2-10, aprovechan el hecho de que la resistencia eléctrica de un material cambia al cambiar su temperatura. Existen dos tipos de sensores de temperatura clave: los dispositivos metálicos (normalmente conocidos como RTD) y los termistores. Los RTD funcionan gracias al cambio de resistencia en un metal, la cual aumenta en forma más o menos lineal con la temperatura [32].



Fig. 2-10. Sensor de temperatura por resistencia (RTD) [32].

2.5.1.3. TERMISTOR PTC/NTC

Un termistor es un sensor resistivo de temperatura. Su funcionamiento se basa en la variación de la resistividad que presenta un semiconductor con la temperatura. El término *termistor* proviene de *Thermally Sensitive Resistor*. Existen dos tipos de termistor [33]:

- NTC: Coeficiente de Temperatura Negativo.
- PTC: Coeficiente de Temperatura Positivo.

En los NTC, la resistencia disminuye cuando aumenta la temperatura, mientras que, en los PTC, ésta aumenta cuando aumenta la temperatura [33].

Su funcionamiento se basa en la variación de la resistencia de un semiconductor con la temperatura, debido a la variación de la concentración de portadores. Para los termistores NTC, al aumentar la temperatura, aumentará también la concentración de portadores, por lo que la resistencia será menor, de ahí que el coeficiente sea negativo. Para los termistores PTC, en el caso de un semiconductor con un dopado muy intenso, éste adquirirá propiedades metálicas, tomando un coeficiente positivo en un margen de temperatura limitado. Usualmente, los termistores se fabrican a partir de óxidos semiconductores, tales como el óxido férrico, el óxido de níquel, o el óxido de cobalto [33].

Sin embargo, a diferencia de los sensores RTD, la variación de la resistencia con la temperatura es no lineal. Para un termistor NTC, la característica es hiperbólica. Para pequeños incrementos de temperatura, se darán grandes incrementos de resistencia [33].

2.5.1.4. SENSOR DE TEMPERATURA BIMETÁLICO

Los dispositivos bimetálicos, como el de Fig. 2-11, aprovechan la diferencia en la tasa de dilatación térmica entre diferentes metales. Se unen entre sí tiras de dos metales. Cuando se calientan, un lado se dilatará más que el otro, y la curvatura resultante se traduce a una lectura de temperatura mediante una articulación mecánica a un apuntador. Estos dispositivos son portátiles y no requieren una fuente de alimentación, pero normalmente no son tan sensibles como los termopares o RTD y no se prestan fácilmente al registro de temperatura [32].



Fig. 2-11. Sensor de temperatura bimetalico [32].

2.5.1.5. SENSOR DE TEMPERATURA POR DILATACIÓN DE FLUIDO

Los dispositivos de dilatación de fluido, cuyo ejemplo típico es el termómetro doméstico, en general vienen en dos clasificaciones principales: el tipo de mercurio (mostrado en Fig. 2-12) y el tipo de líquido orgánico. También hay disponibles versiones que usan gas en lugar de líquido. El mercurio se considera un riesgo ambiental, así que hay regulaciones que rigen el embarque de dispositivos que lo contienen. Los sensores de dilatación de fluido no requieren energía eléctrica, no plantean riesgos de explosión y son estables incluso después de ciclos repetidos. Por otra parte, no generan datos que se registren o transmitan fácilmente, y no pueden hacer mediciones puntuales [32].



Fig. 2-12. Termómetros de cristal, de dilatación de mercurio [32].

2.5.1.6. SENSOR DE TEMPERATURA POR CAMBIO DE ESTADO

Los sensores de cambio de estado consisten en etiquetas (véase Fig. 2-13), pellets o gránulos, crayones, lacas o cristales líquidos cuya apariencia cambia una vez que se alcanza cierta temperatura. Se usan por ejemplo con trampas de vapor: cuando una trampa supera una cierta temperatura, un punto blanco en una etiqueta de sensor adherida a la trampa se volverá negra. El tiempo de respuesta típicamente es de varios minutos, así que estos dispositivos con frecuencia no responden a los cambios de temperatura transitorios, y la precisión es más baja que con otros tipos de sensores. Además, el cambio en estado es irreversible, excepto en el caso de las pantallas de cristal líquido. Aun así, los sensores de cambio de estado pueden ser útiles cuando se necesita confirmación de que la temperatura en un equipo o material

no ha superado un cierto nivel, por ejemplo por razones técnicas o legales durante el embarque del producto [32].

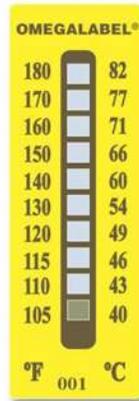


Fig. 2-13. Etiquetas no reversibles de temperatura [32].

2.5.1.7. SENSOR RADIADOR INFRARROJO

El diseño más básico consiste en un lente para enfocar la energía infrarroja (IR) en un detector, que convierte la energía en una señal eléctrica que se puede exhibir en unidades de temperatura después de que se compensa la variación en la temperatura ambiente. Esta configuración facilita la medición de temperatura a distancia sin contacto con el objeto a medir. Como tal, el termómetro infrarrojo es útil para medir la temperatura en circunstancias en las que los termopares u otros sensores de tipo de sonda no se pueden usar o no producen datos precisos por diversas razones. Algunas circunstancias típicas son cuando el objeto a medir está en movimiento; cuando el objeto está rodeado de un campo electromagnético, como en calentamiento por inducción; cuando el objeto está contenido en un vacío u otra atmósfera controlada; en aplicaciones en las que se requiere una respuesta rápida [34].

Los termómetros fabricados con este tipo de sensores (como el que se muestra en Fig. 2-14) deben tener en cuenta factores como el campo visual, tamaño del objetivo, distancia, tipo de superficie a medir, rasgo espectral, entre otros [34].



Fig. 2-14. Termómetro infrarrojo de mano [34].

2.5.2. PH-METROS

El pH-metro es un instrumento utilizado para medir el pH de una solución. El pH es la unidad de medida que describe el grado de acidez o alcalinidad y es medido en una escala que va de 0 a 14 [35].

Más específicamente, el pH mide la cantidad de iones de hidrógeno que contiene una solución determinada, el significado de sus siglas es “potencial de Hidrogeniones”. El pH se ha convertido en una forma práctica de manejar cifras de alcalinidad, en lugar de otros métodos un poco más complicados [36].

2.5.2.1. CINTAS DE PH

Una indicación aproximada del pH puede ser obtenida usando indicadores o cintas de pH, que cambian de color en función de la variación del nivel de pH. Esos indicadores presentan limitaciones en términos de exactitud y pueden ser difíciles de interpretar correctamente en muestras oscuras o coloridas [35].

2.5.2.2. MEDIDOR DE PH

Mediciones más exactas son obtenidas usando un medidor de pH, como el mostrado en Fig. 2-15. El sistema de medición es formado por tres partes: un electrodo de medición de pH, un electrodo de referencia y un medidor de alta impedancia de entrada. El electrodo de pH puede ser considerado como si fuera una batería, con una tensión que varía conforme el pH de la solución medida. El electrodo que mide el pH es un bulbo de vidrio sensible a iones de hidrógeno, con una salida en milivoltios que varía conforme las alteraciones en la concentración relativa de los iones de hidrógeno dentro y fuera del bulbo. La salida del electrodo de referencia no cambia con la actividad de los iones de hidrógeno. El electrodo de pH posee una resistencia interna muy alta, lo que dificulta la medición de la variación de la tensión con el pH. Por lo tanto, la impedancia de la entrada del medidor de pH y las resistencias de dispersión son factores importantes. Básicamente, el medidor de pH es un amplificador de alta impedancia que mide con exactitud las tensiones mínimas del electrodo y exhibe los resultados directamente en unidades de pH en una pantalla analógica o digital. En algunos casos, las tensiones también pueden ser interpretadas para aplicaciones especiales o uso con electrodos de iones selectivos o de potencial de oxidación/reducción (ORP) [35].

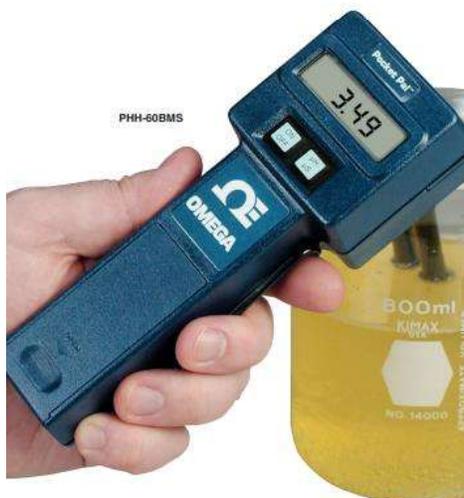


Fig. 2-15. Medidor electrónico de pH portátil [35].

2.5.3. MEDIDOR DE CONDUCTIVIDAD ELÉCTRICA

La conductividad se define como la capacidad de una sustancia de conducir la corriente eléctrica y es lo contrario de la resistencia. La unidad de medición utilizada comúnmente es el Siemens/cm (S/cm). En el caso de medidas en soluciones acuosas, el valor de la conductividad es directamente proporcional a la concentración de sólidos disueltos, por lo tanto, cuanto mayor sea dicha concentración, mayor será la conductividad [37].

Además de los medidores de conductividad, existen instrumentos que convierten automáticamente el valor de conductividad en ppm¹⁵, ofreciendo directamente las medidas de la concentración de sólidos disueltos [37].



Fig. 2-16. Medidor de conductividad potenciométrica de sobremesa [38].

¹⁵ Partes por millón. Es una unidad de medida con la que se mide la concentración.

Es posible diferenciar los distintos medidores de conductividad según el método de medición que utilicen: amperimétrico o potenciométrico [37].

2.5.3.1. SISTEMA AMPERIMÉTRICO

El sistema amperimétrico aplica una diferencia potencial conocida V a dos electrodos y mide la corriente alterna A que pasa a través de ellos. Según la ley de Ohm, las dos dimensiones está sujetas a la relación: $I = V/R$, donde R es la resistencia. Por tanto, cuanto más elevada sea la corriente obtenida, mayor será la conductividad [37].

2.5.3.2. SISTEMA POTENCIOMÉTRICO

El sistema potenciométrico de 4 anillos (como el de Fig. 2-16) está basado en el principio de inducción y elimina los problemas comunes asociados al sistema amperimétrico, como los efectos de la polarización. A los dos anillos externos va aplicada una corriente alterna, mientras que los dos anillos internos miden la diferencia de potencial inducida por el flujo de corriente, que depende de la conductividad de la solución donde se ha sumergido la sonda [37].

2.5.4. SENSOR DE INTENSIDAD LUMINOSA

Un sensor de intensidad luminosa o sensor fotoeléctrico, es un dispositivo electrónico que responde al cambio en la intensidad de la luz [39].

Los sensores de luz se usan para detectar el nivel de luz y producir una señal de salida representativa respecto a la cantidad de luz detectada. Un sensor de luz incluye un transductor fotoeléctrico para convertir la luz a una señal eléctrica y puede incluir electrónica para el acondicionamiento de la señal, compensación y formateo de la señal de salida [39].

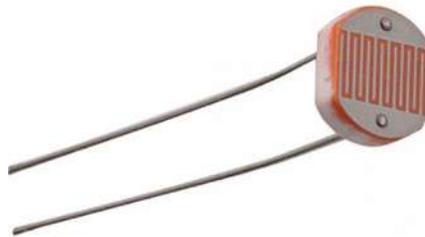


Fig. 2-17. Resistor Dependiente de la Luz (LDR) [40].

El sensor de luz más común es el Resistor Dependiente de la Luz (LDR por sus siglas en inglés), como el que se muestra en Fig. 2-17. Un LDR es básicamente un resistor que cambia su resistencia cuando cambia la intensidad de la luz [39].

2.5.5. SENSOR DE HUMEDAD

La medición de la humedad relativa consiste en la relación entre la presión parcial del vapor de agua en el gas de que se trate y la presión de saturación del vapor, a una temperatura dada. Por lo tanto, la humedad relativa es función de la temperatura. La medición es expresada como un porcentaje [41].

Los sensores de humedad se aplican para detectar el nivel de líquido en un depósito, o en sistemas de riego de jardines para detectar cuándo las plantas necesitan riego y cuándo no. Permiten medir la temperatura de punto de rocío, humedad absoluta y relación de mezcla. Existen los sensores que pueden mostrar su medición en valores analógicos o digitales [41].

Un sensor **analógico** de humedad (como el mostrado en Fig. 2-18) mide la humedad del aire relativo usando un sistema basado en un condensador. El sensor está hecho de una película generalmente de vidrio o de cerámica. El material aislante que absorbe el agua está hecho de un polímero que toma y libera el agua basándose en la humedad relativa de la zona dada. Esto cambia el nivel de carga en el condensador del circuito en el cuadro eléctrico [41].



Fig. 2-18. Sensor analógico de temperatura y humedad [42].

Un sensor **digital** de humedad (como el mostrado en Fig. 2-19) funciona a través de dos micro-sensores que se calibran a la humedad relativa de la zona dada. Estos se convierten luego en formato digital a través de un proceso de conversión de analógico a digital que se realiza mediante un chip situado en el mismo circuito. Un sistema basado en una máquina hecha de electrodos con polímeros es lo que constituye la capacitancia del sensor. Esto protege el sensor del panel frontal del usuario (interfaz) [41].

La precisión de los sensores se ve afectada por la temperatura y humedad a la que fue calibrado el sensor, la no-linealidad de los sensores, el envejecimiento del sensor, contaminantes externos, precisión, etc. [41].

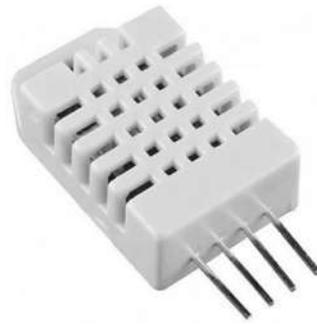


Fig. 2-19. Sensor digital de temperatura y humedad [43].

La medición de la humedad relativa (HR) consiste en la relación entre la presión parcial del vapor de agua en el gas de que se trate y la presión de saturación del vapor, a una temperatura dada. Por lo tanto, la humedad relativa es función de la temperatura. La medición es expresada como un porcentaje. La humedad relativa es un parámetro utilizado principalmente en aplicaciones ambientales (ej. acondicionamiento de aire) o mediciones meteorológicas ya que impacta directamente en el confort humano. Cuando los niveles de humedad relativa son bajos puede producirse electricidad estática que podría dañar al equipamiento electrónico [41].

Para medir la humedad relativa, se pueden utilizar sensores basados en psicrometría, desplazamiento, resistivos, capacitivos y por absorción de líquido [41].

2.5.5.1. PSICROMETRÍA POR BULBO HÚMEDO/BULBO SECO

La psicrometría, proveniente del griego “frío” y “medición” [44], desde hace tiempo es uno de los métodos más populares para el monitoreo de la humedad debido a su simplicidad e inherente bajo costo. Un psicrómetro industrial típico consiste de un par de termómetros eléctricos acoplados, uno de los cuales opera en estado húmedo. Cuando el dispositivo funciona la evaporación del agua enfría el termómetro humedecido, resultando una diferencia medible con la temperatura ambiente o la temperatura del bulbo seco. Cuando el bulbo húmedo alcanza su máxima caída de temperatura la humedad puede determinarse comparando la temperatura de los dos termómetros en una tabla psicrométrica [41].

2.5.5.2. SENSORES POR DESPLAZAMIENTO

Es quizás el tipo de sensor más antiguo y de uso común, utiliza un mecanismo para medir la expansión o contracción de un cierto material que es proporcional a los cambios en el nivel de humedad relativa. Los materiales más comunes el nylon y la celulosa. Las ventajas de este tipo de sensores son el bajo costo de fabricación y es altamente inmune a la contaminación. Las desventajas son la tendencia a la descalibración en el tiempo y los efectos de histéresis significativos [41].

2.5.5.3. SENSOR DE BLOQUE DE POLÍMERO RESISTIVO

Están compuestos de un sustrato cerámico aislante sobre el cual se deposita una grilla de electrodos. Estos electrodos se cubren con una sal sensible a la humedad embebida en una resina (polímero). La resina se recubre entonces con una capa protectora permeable al vapor de agua. A medida que la

humedad permea la capa de protección, el polímero resulta ionizado y estos iones se movilizan dentro de la resina. Cuando los electrodos son excitados por una corriente alterna, la impedancia del sensor se mide y es usada para calcular el porcentaje de humedad relativa. Por su misma estructura este tipo de sensores son relativamente inmunes a la contaminación superficial ya que no afecta su precisión, aunque si el tiempo de respuesta. Debido a los valores extremadamente altos de resistencia del sensor a niveles de humedad menores que 20% es apropiado para los rangos altos de humedad [41].

2.5.5.4. *SENSORES CAPACITIVOS*

Los sensores capacitivos (polímero orgánico capacitivo) son diseñados normalmente con platos paralelos con electrodos porosos o con filamentos entrelazados en el sustrato. El material dieléctrico absorbe o elimina vapor de agua del ambiente con los cambios del nivel de humedad. Los cambios resultantes en la constante dieléctrica causan una variación en el valor de la capacidad del dispositivo por lo que resulta una impedancia que varía con la humedad. Un cambio en la constante dieléctrica de aproximadamente el 30% corresponde a una variación de 0-100% en la humedad relativa. El material sensor es muy delgado para alcanzar grandes cambios en la señal con la humedad. Esto permite al vapor de agua entrar y salir fácilmente y el secado rápido para la sencilla calibración del sensor. Este tipo de sensor es especialmente apropiado para ambiente de alta temperatura porque el coeficiente de temperatura es bajo y el polímero dieléctrico puede soportar altas temperaturas. Los sensores capacitivos son también apropiados para aplicaciones que requieran un alto grado de sensibilidad a niveles bajos de humedad, donde proveen una respuesta relativamente rápida. A valores de humedad superiores al 85%, sin embargo, el sensor tiene una tendencia a saturar y se transforma en no lineal [41].

2.6. ACTUADORES

Los actuadores son dispositivos que brindan la posibilidad de transformar diferentes tipos de energía para generar algún funcionamiento dentro de un sistema automatizado determinado. Usualmente, los actuadores generan una fuerza mecánica a partir de distintos tipos de energía, como puede ser eléctrica, neumática, o hidráulica [45].

El actuador más antiguo que se conoce es el manual. El funcionamiento de este dispositivo se genera a partir de la fuerza de trabajo de un humano. El trabajador mueve el dispositivo para generar un resultado o accionar el funcionamiento de otro mecanismo [45].

Los actuadores **hidráulicos** utilizan líquidos a presión para generar fuerza mecánica. Uno de los líquidos más utilizados es el aceite. Este tipo de actuadores suelen ser usados en máquinas de gran peso y tamaño. Frecuentemente son clasificados con base en su resistencia mecánica y su velocidad. Cuando se usan con estos actuadores es muy importante impedir que el líquido de trabajo se filtre a alguna parte del sistema ya que podría ser altamente nocivo. Para disminuir el riesgo de filtraciones, se recomienda realizar periódicamente rutinas de mantenimiento preventivo a las máquinas y sistemas [45].

Los actuadores **neumáticos** son aquellos que convierten la energía del aire comprimido en trabajo mecánico. Son, en esencia, idénticos a los actuadores hidráulicos, pero el rango de compresión de estos es menor, además de que hay una pequeña diferencia en cuanto al uso y en lo que se refiere a la estructura [46].

Los actuadores **electrónicos** son accionados por medio de corrientes eléctricas. Existen actuadores electrónicos que consumen una considerable cantidad de energía, para este tipo de casos se utilizan controladores. Los actuadores eléctricos son utilizados en diferentes aparatos mecatrónicos, como robots. Algunos tipos de actuadores electrónicos son [45]:

- Motores de corriente directa.
- Motores de pulsos (paso a paso).
- Electro válvulas.
- Aleaciones con memoria de forma (como el Nitinol).
- Relevadores.

2.6.1. BOMBAS

Las bombas (ver un ejemplo de una en Fig. 2-20) son dispositivos que se encargan de transferir energía a la corriente del fluido impulsándolo, desde un estado de baja presión estática a otro de mayor presión. Están compuestas por un elemento rotatorio denominado impulsor, el cual se encuentra dentro de una carcasa llamada voluta [47].

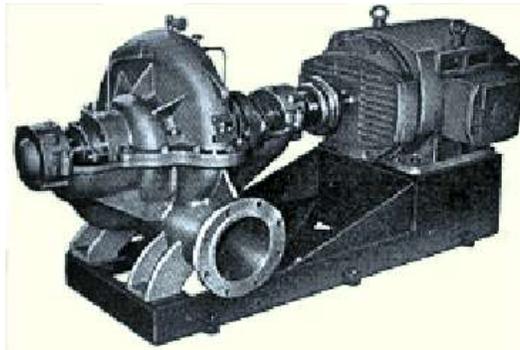


Fig. 2-20. Bomba centrífuga [47].

Una bomba centrífuga, como la mostrada en Fig. 2-20, es una máquina que consiste de un conjunto de paletas rotatorias encerradas dentro de una caja o cárter, o una cubierta o coraza. Se denominan así porque la elevación de presión que crean es ampliamente atribuible a la acción centrífuga. Las paletas imparten energía al fluido por la fuerza de esta misma acción. Así, despojada de todos los refinamientos, una bomba centrífuga tiene dos partes principales: Un elemento giratorio, incluyendo un impulsor y una flecha, y un elemento estacionario, compuesto por una cubierta, estoperas¹⁶ y chumaceras [47].

¹⁶ Junto con los empaques y sellos, evita el flujo hacia fuera del líquido bombeado a través del orificio por donde pasa la flecha de la bomba y el flujo de aire hacia el interior de la bomba.

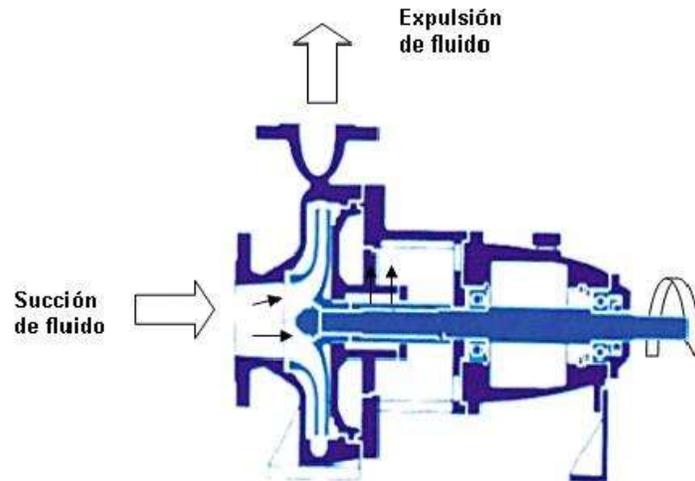


Fig. 2-21. Principio de funcionamiento de una bomba centrífuga [47].

El flujo entra a la bomba a través del centro u ojo del rodete y el fluido gana energía a medida que las paletas del rodete lo transportan hacia fuera en dirección radial. Esta aceleración produce un apreciable aumento de energía de presión y cinética, lo cual es debido a la forma de caracol de la voluta para generar un incremento gradual en el área de flujo de tal manera que la energía cinética a la salida del rodete se convierte en cabeza de presión a la salida [47]. Este proceso se ilustra en Fig. 2-21.

2.6.2. RELEVADORES

El relé o relevador es un dispositivo electromecánico. Funciona como un interruptor controlado por un circuito eléctrico en el que, por medio de una bobina y un electroimán, se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes y con diferentes valores de potencia. Fue inventado por Joseph Henry en 1835 [48].

El relevador es capaz de controlar un circuito de salida de mayor potencia que el de entrada, por lo que podría considerarse como amplificador eléctrico. Además, es un elemento que separa la etapa de control y la etapa de potencia, para así poder proteger ambos circuitos de cortos circuitos o sobrecargas en alguno de los dos [48].

Un relé es un interruptor accionado por un electroimán (véase Fig. 2-22). Un electroimán está formado por una barra de hierro dulce, llamada núcleo, rodeada por una bobina de hilo de cobre. Al pasar una corriente eléctrica por la bobina el núcleo de hierro se magnetiza por efecto del campo magnético producido por la bobina, convirtiéndose en un imán tanto más potente cuanto mayor sea la intensidad de la corriente y el número de vueltas de la bobina. Al abrir de nuevo el interruptor y dejar de pasar corriente por la bobina, desaparece el campo magnético y el núcleo deja de ser un imán [48].

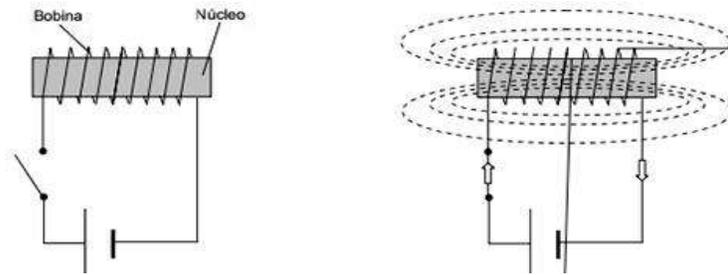


Fig. 2-22. Electroimán [48].

El relé más sencillo, mostrado en Fig. 2-23, está formado por un electroimán y un interruptor de contactos. Al pasar una pequeña corriente por la bobina, el núcleo se imanta y atrae al inducido por uno de sus extremos, empujando por el otro a uno de los contactos hasta que se juntan, permitiendo el paso de la corriente a través de ellos. Esta corriente es, normalmente, mucho mayor que la que pasa por la bobina [48].

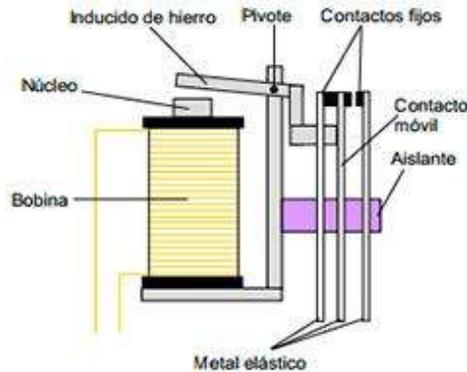


Fig. 2-23. Modelo de relevador sencillo [48].

2.6.3. ILUMINACIÓN

Las lámparas son dispositivos que transforman una energía eléctrica o química en energía lumínica. Desde un punto de vista más técnico, se distingue entre dos objetos: la lámpara es el dispositivo que produce la luz, mientras que la luminaria es el aparato que le sirve de soporte [49].

Existen diversos tipos de lámparas, entre las que destacan las de incandescencia, las de descarga y las de tecnología LED [49].

2.6.3.1. DE INCANDESCENCIA

Hay varios tipos de incandescencia: la primera es por combustión de alguna sustancia, ya sea sólida como una antorcha de madera, líquida como en una lámpara de aceite o gaseosa como en las lámparas de gas. La segunda es pasando una corriente eléctrica a través de un hilo conductor muy delgado como ocurre en las bombillas corrientes. Tanto de una forma como de otra, obtenemos luz y calor (ya sea calentando las moléculas de aire o por radiaciones infrarrojas). En general los rendimientos de este tipo de lámparas son bajos debido a que la mayor parte de la energía consumida se convierte en calor [50].

Se denomina lámpara incandescente al dispositivo que produce luz mediante el calentamiento por efecto Joule de un filamento metálico, hasta ponerlo al rojo blanco, mediante el paso de corriente eléctrica. Se conoce como efecto Joule al fenómeno por el cual, si en un conductor circula corriente eléctrica, parte de la energía cinética de los electrones se transforma en calor debido a los choques que sufren con los átomos del material conductor por el que circulan, elevando la temperatura del mismo. El nombre es en honor a su descubridor el físico británico James Prescott Joule [50].

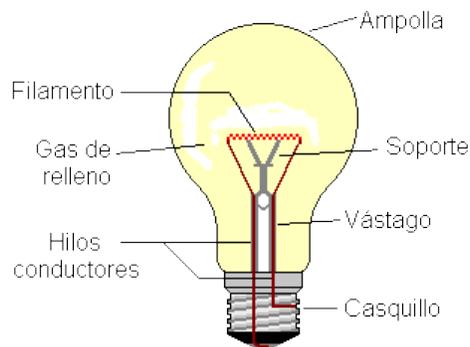


Fig. 2-24. Partes de una bombilla [50].

Es la más popular debido a su bajo precio y el color cálido de su luz. También es la de menor vida útil. No emite luz de la zona de colores fríos, convierte alrededor del 15% de la energía consumida en luz visible, 25% en calor y el 60% restante en luz ultravioleta e infrarroja (que también acaban convirtiéndose en calor). En Fig. 2-24, se muestra una bombilla incandescente y sus partes [50].

Existen dos tipos de lámparas incandescentes: las que contienen un gas halógeno en su interior (véase Fig. 2-25) y las que no lo contienen [50].



Fig. 2-25. Lámparas incandescentes de halógeno [50].

2.6.3.2. DE DESCARGA

Las lámparas de descarga son fuentes luminosas que producen luz mediante una descarga eléctrica en gases o vapores metálicos presentes en el interior de la ampolla [50].

Para encender las lámparas de descarga se requiere de un dispositivo llamado balasto, que produce el encendido con un alto voltaje inicial y luego disminuye la energía eléctrica al nivel operativo normal. Los

balastos electromagnéticos son los tradicionales de filamentos de cobre, que ya están siendo reemplazados por balastos electrónicos [50].



Fig. 2-26. Lámparas fluorescentes [50].

Las lámparas de descarga se pueden clasificar según el gas utilizado (vapor de mercurio o sodio) o la presión a la que este se encuentre (alta o baja presión). Las propiedades varían mucho de unas a otras y esto las hace adecuadas para unos usos u otros. Las principales son lámparas de vapor de mercurio, lámparas de baja presión y lámparas fluorescentes (ver Fig. 2-26) [50].

Las lámparas fluorescentes presentan varias ventajas frente a las incandescentes: Consumen hasta 3 veces menos corriente, sus colores son más fieles al color real, su emisión de luz es de 4 a 6 veces mayor que la de una incandescente de la misma potencia, provee una luz más uniforme y menos deslumbrante, tienen mucho menos calentamiento, y duran al menos 2.5 veces más [50].

2.6.3.3. LED

El LED (Light Emitting Diode) es un diodo emisor de luz, es decir, un dispositivo semiconductor que emite luz cuando circula por él, corriente eléctrica (ver Fig. 2-27); es un proyector electroluminiscente que emite luz mediante la recombinación de los pares de portadores de carga de un semiconductor [50].

La luz no se genera a través de un filamento incandescente sino por electroluminiscencia. Esto significa que se liberan fotones (luz) debido a electrones que cambian de nivel de energía durante su desplazamiento por el material semiconductor (diodo) [50].

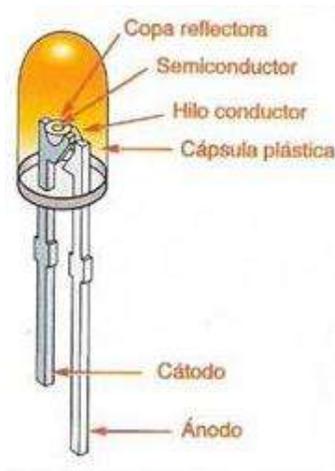


Fig. 2-27. Componentes de un LED [50].

El LED presenta muchas ventajas frente a los otros dos tipos de lámparas [50]:

- Elevada resistencia física: elementos 100% sólidos, resisten golpes y vibraciones mucho mejor que lámparas convencionales.
- Mayor duración, por no depender de que el filamento se quemé.
- Elevada eficiencia de conversión de la electricidad entrante hacia la energía luminosa: Aprovechan hasta el 90%.
- Si algún LED se rompe es posible reemplazarlo.
- Baratos y fáciles de fabricar.
- Larga vida útil: hasta 100.000 horas de vida útil comparado con 8000 horas de una lámpara convencional.
- Pueden emitir hasta 16 millones de colores distintos (esto en combinación de al menos 3 colores de LED distintos, como RGB¹⁷).
- No emiten radiaciones infrarrojas y/ o ultravioletas. Muy importante para la iluminación de obras de arte, donde habitualmente la radiación deteriora el objeto a iluminar.
- No explotan.
- No contaminan ni poseen elementos contaminantes.
- No emiten calor, por lo que son muy adecuados iluminar objetos inflamables y ahorrar energía necesaria para regular la temperatura ambiental.
- Resisten bien las variaciones en temperatura por lo cual son adecuados para iluminación de exteriores.
- Tamaño reducido: pocos milímetros cúbicos.
- Su velocidad de transmisión permite utilizarlos en los displays alfanuméricos o en aplicaciones de telecomunicación por aire o por fibra óptica.

¹⁷ Sigla en inglés de red, green, blue; en español «rojo, verde y azul».

- Funcionan con corriente continua, por lo que se reducen los riesgos de manipulación y electrocución por descuido.
- Muy adecuado para aplicaciones en zonas con elevada afluencia de público: centros comerciales, discotecas, teatros, discotecas, etc.

2.7. PERIFÉRICOS

Los dispositivos periféricos son instrumentos tecnológicos que interpretan la información y permiten la comunicación entre las personas y las computadoras (o microcontroladores) [51].

También existen los dispositivos periféricos de comunicación, que son los que permiten la interacción con otras máquinas (la tarjeta de red, el concentrador o el enrutador, etc.) [51].

2.7.1. CÁMARAS WEB

Una cámara web (véase Fig. 2-28) o cámara de red es una pequeña cámara digital conectada a una computadora o microcontrolador, la cual puede capturar imágenes y transmitir las a través de Internet, ya sea a una página web u otras computadoras de forma privada [52].

Una cámara web necesita una computadora o microcontrolador para transmitir las imágenes. Sin embargo, existen otras cámaras autónomas que únicamente necesitan un punto de acceso a la red informática, bien sea ethernet o inalámbrico. Para diferenciarlas de las cámaras web se las denomina cámaras de red [52].



Fig. 2-28. Cámara web para computadora [53].

Por lo general puede transmitir imágenes en vivo, pero también puede capturar imágenes o pequeños videos (dependiendo del programa de la cámara web) que pueden ser grabados y transmitidos por Internet. La cámara web es un dispositivo de entrada, ya que por medio de él se puede transmitir imágenes hacia la computadora [52].

2.8. REDES DE COMPUTADORAS

El término genérico *red* hace referencia a un conjunto de entidades (objetos, personas, etc.) conectadas entre sí. Por lo tanto, una red permite que circulen elementos materiales o inmateriales entre estas entidades, según reglas bien definidas. En el caso particular de las redes de computadoras, una red se define como un conjunto de equipos conectados entre sí mediante líneas físicas que intercambian información bajo la forma de datos digitales. La red más pequeña posible está conformada por dos equipos o dispositivos periféricos conectados entre sí [54].

El objetivo fundamental de una red informática es compartir recursos (archivos, aplicaciones o hardware, una conexión a Internet, etc.). Otros objetivos son facilitar la comunicación entre personas (correo electrónico, debates en vivo, etc.) y la comunicación entre procesos (por ejemplo, entre equipos industriales), garantizar el acceso único y universal a la información (bases de datos en red) e incluso poder jugar videojuegos de tipo multijugador [54].

Existen dos tipos de redes: redes de igual a igual y redes organizadas alrededor de servidores (Cliente/Servidor). Estos dos tipos de redes tienen diferentes capacidades [54].

2.8.1. RED DE ÁREA LOCAL (LAN)

Una LAN (Local Area Network) es un grupo de equipos pertenecientes a una misma organización y conectados dentro de un área geográfica pequeña a través de una red, generalmente con la misma tecnología [55]. Se utiliza para conectar equipos de una compañía u organización. Permite intercambiar información, comunicarse, acceder a diversos servicios [56].

Según los servicios que proporciona, se pueden distinguir dos modos de funcionamiento de una LAN: en una red *de igual a igual* (P2P), en la que la comunicación se establece de un equipo a otro sin la necesidad de un equipo central y donde cada equipo tiene la misma función; y en un entorno *cliente/servidor*, en el que un equipo central se encarga de brindar los servicios de red a los usuarios [55].

Una red de área local está compuesta por equipos conectados mediante un conjunto de elementos de software y hardware. Los elementos de hardware utilizados para la conexión de los equipos son [56]:

La tarjeta de red (o acoplador): Se trata de una tarjeta que se conecta a la placa madre del equipo y que se comunica con el medio físico, es decir, con las líneas físicas a través de las cuales viaja la información [56].

El transceptor (o adaptador): Se utiliza para transformar las señales que viajan por el soporte físico en señales lógicas que la tarjeta de red puede manejar, tanto para enviar como para recibir datos [56].

El conector para soporte físico (comúnmente RJ-45): Es el elemento utilizado para conectar mecánicamente la tarjeta de red con el soporte físico [56].

El soporte físico de interconexión: Es el soporte (generalmente cableado) utilizado para conectar los equipos entre sí. Los principales medios de soporte físicos utilizados son el cable coaxial, el par trenzado y la fibra óptica [56].

No obstante, los dispositivos de hardware solos no son suficientes para crear una red de área local que pueda utilizarse. También es necesario fijar un método de acceso estándar entre los equipos, para que sepan cómo se intercambiarán los datos, en especial cuando más de dos equipos comparten el mismo soporte físico. Este método de acceso se denomina *topología lógica*. La topología lógica se lleva a cabo mediante un *protocolo de acceso*. Los protocolos de acceso más comunes son el Ethernet y la red en anillo [56].

2.8.2. RED DE ÁREA LOCAL VIRTUAL (VLAN)

Una VLAN (Virtual Local Area Network) es una red de área local que agrupa un conjunto de equipos de manera lógica y no física. Gracias a las redes virtuales es posible liberarse de las limitaciones de la arquitectura física (limitaciones geográficas, limitaciones de dirección, etc.), ya que se define una segmentación lógica basada en el agrupamiento de equipos según determinados criterios (direcciones MAC, números de puertos, protocolos, etc.) [57].

La VLAN permite definir una nueva red por encima de la red física y, por lo tanto, ofrece diversas ventajas: una mayor flexibilidad en la administración y en los cambios de la red, ya que la arquitectura puede cambiarse usando los parámetros de los conmutadores; un aumento de la seguridad, puesto que la información se encapsula en un nivel adicional y puede ser analizada; una disminución en la transmisión de tráfico en la red [57].

2.8.3. INTERNET

Existen protocolos para permitir que las redes se puedan comunicar entre sí, asimilándose a una telaraña gigante (de ahí el nombre Web, en inglés), en la que la red más grande contiene a las demás redes. Esto se denomina Internet. Existen diferentes protocolos en Internet (lenguajes entre equipos) que permiten llevar a cabo diferentes acciones como: IRC (Chat en directo), HTTP (Navegar por páginas web), FTP (Transferir archivos) y muchas otras más [58].

En Internet se utilizan diversos protocolos, que forman parte de la serie de protocolos TCP/IP. Esta serie se basa en la identificación de cada equipo con una dirección denominada dirección IP, que posibilita la transmisión de datos a la dirección correcta. Estas direcciones se relacionan con nombres de dominios (explicado en el subcapítulo **¡Error! No se encuentra el origen de la referencia.**) para poder recordarse con más facilidad [58].

A cada acción se le asigna un número (el puerto) que se envía durante la comunicación (la transmisión se lleva a cabo mediante pequeños paquetes de información). Por lo tanto, es posible conocer con qué programa se relaciona cada pequeño paquete. Por ejemplo [58]:

- Los paquetes HTTP llegan al puerto 80 y se transmiten al navegador de Internet que solicitó la página.
- Los paquetes IRC llegan al puerto 6667 (u otro, generalmente ubicado cerca de 7000) y se transmiten a un cliente de chat IRC (Internet Relay Chat) como mIRC u otro.

2.8.4. MÓDEM

Un módem es un dispositivo que se utiliza para transmitir información entre varios equipos (básicamente 2) a través de las líneas telefónicas. Las computadoras operan de forma digital y usan el lenguaje binario, sin embargo, los módems son analógicos; por esta razón, el módem convierte la información digital en ondas analógicas y viceversa. De ahí su nombre: **MO**dulador/**DE**Modulador (véase Fig. 2-29) [59].

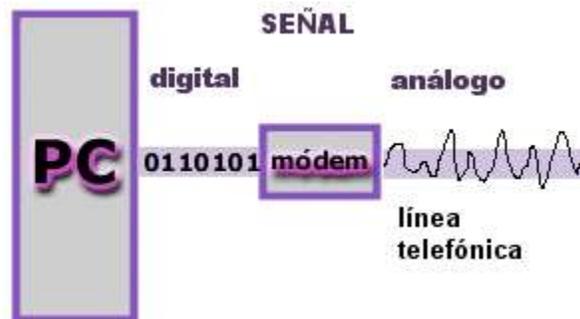


Fig. 2-29. El módem convierte señales digitales a analógicas y viceversa [59].

La línea telefónica clásica estaba diseñada para funcionar con teléfonos, por eso se necesitaba un módem para establecer la comunicación con un equipo remoto por medio de un número telefónico antes de poder intercambiar la información. El lenguaje que utilizan los ordenadores para comunicarse se denomina protocolo. Los protocolos que más se utilizan son el protocolo PPP (Protocolo de Punto a Punto) y el protocolo SLIP (Protocolo de Línea Serial de Internet) [59].

2.8.5. RUTEADOR

Ruteador quiere decir enrutador, es decir, “buscador” del camino o ruta. A diferencia de una red local del tipo Ethernet (la más habitual) en la que se utilizan switches que evitan las colisiones entre un número delimitado de usuarios, en Internet el volumen de estos es colosalmente alto. Así que los routers en vez de mover un mensaje entre todas las redes que componen Internet, solamente mueve el mensaje entre las dos redes que están involucradas: La del emisor y la del destinatario. Un ruteador tiene dos misiones distintas, aunque relacionadas [60]:

- Asegurarse de que la información no va a donde no es necesario.
- Encargarse de que la información si llegue al destinatario.

El ruteador une las redes del emisor y el destinatario de una información determinada (email, página Web, etc.) y además sólo transmite entre las mismas, la información necesaria [60]. En Fig. 2-30 se muestra la imagen de un ruteador casero común.

En el movimiento de información en Internet, la información se divide en pequeñas unidades o paquetes (de unos 1500 bytes por paquete). Cada paquete lleva información del origen, el destinatario y lugar de ese paquete en el total de la información transmitida (para que luego el mensaje pueda ser reconstruido correctamente) e información de cómo confirmar su llegada al destino [60].



Fig. 2-30. Fotografía de un ruteador común [61].

El ruteador se encarga de analizar paquete por paquete el origen y el destino y busca el camino más corto de uno a otro. Esta forma de transmitir información tiene grandes ventajas [60]:

- El ruteador es capaz de ver si una ruta no funciona y buscar una alternativa.
- El ruteador es capaz incluso de buscar la ruta más rápida (por ejemplo, la que tenga menos tráfico) en caso de poder escoger entre varias posibilidades.

Esto hace que Internet sea un sistema tan robusto para el envío de información [60].

En Internet, hay miles de routers que trabajan para buscar el camino más rápido de un punto a otro. Si se tiene un ruteador en una conexión a Internet, este buscará el ruteador óptimo para llegar a un destinatario, y ese ruteador óptimo, buscará a su vez el siguiente óptimo para llegar al destinatario. Tanto los routers medianos como los más sofisticados permiten configurar qué información se desea que pueda entrar o salir del PC¹⁸ o red. En caso de que se desee ampliar las posibilidades de control se deberá añadir un dispositivo llamado Firewall (en inglés) o Cortafuegos (en español). [60].

La primera función más básica de un ruteador es saber si el destinatario de un paquete de información está en la propia red o en una remota. Para determinarlo, el ruteador utiliza un mecanismo llamado *máscara de subred*. La máscara de subred es parecida a una dirección IP y determina a que grupo de ordenadores pertenece uno en concreto. Si la máscara de subred de un paquete de información enviado no se corresponde a la red de ordenadores, el ruteador determinará que el destino de ese paquete está en alguna otra red [60].

Cada PC conectado a una red (bien sea una local o a la red de redes -Internet-) tiene una tarjeta de red. Esta tarjeta gestiona la entrada y salida de información y tiene una identificación propia llamada identificación MAC (sigla de Media Access Control), conocida también como identificación física. Es única, real y exacta. A esta identificación física le podemos asociar una identificación lógica, la llamada IP. La identificación lógica puede cambiar con el tiempo, pero no así, la identificación física [60].

¹⁸ Computadora personal, por sus siglas en inglés.

El ruteador asocia las direcciones físicas (MAC) a direcciones lógicas (IP). En comunicaciones informáticas, una dirección física (MAC) puede tener varias direcciones lógicas (IP). Una vez identificado en Internet por una dirección lógica, los routers crean unas tablas de las rutas para llegar a otras direcciones IP. Si hay un problema, el ruteador prueba otra ruta y verifica si el paquete llega al destino, si no es así, prueba otra, y si esta tiene éxito, la almacena como posible ruta secundaria para cuando la primera (la más rápida no funcione). Toda esta información de rutas se va actualizando miles de veces por segundo durante las 24 horas del día [60].

2.9. PROTOCOLOS DE COMUNICACIÓN

Un protocolo es un método estándar que permite la comunicación entre procesos (que potencialmente se ejecutan en diferentes equipos), es decir, es un conjunto de reglas y procedimientos que deben respetarse para el envío y la recepción de datos a través de una red. Es un lenguaje común utilizado por todos los actores en la comunicación para intercambiar datos. Existen diversos protocolos de acuerdo a cómo se espera que sea la comunicación [62].

En Internet, los protocolos utilizados pertenecen a una sucesión de protocolos o a un conjunto de protocolos relacionados entre sí. Este conjunto de protocolos se denomina TCP/IP. Entre otros, contiene los siguientes protocolos: HTTP, FTP, ARP, ICMP, IP, TCP, UDP¹⁹, SMTP²⁰, Telnet y NNTP²¹ [62].

Generalmente, los protocolos se clasifican en dos categorías según el nivel de control de datos requerido [62]:

Los protocolos orientados a conexión; que son protocolos que controlan la transmisión de datos durante una comunicación establecida entre dos máquinas. En tal esquema, el equipo receptor envía acuses de recepción durante la comunicación, por lo cual el equipo remitente es responsable de la validez de los datos que está enviando. Los datos se envían entonces como flujo de datos. TCP es un protocolo orientado a conexión [62].

Los protocolos no orientados a conexión; son un método de comunicación en el cual el equipo remitente envía datos sin avisarle al equipo receptor, y este recibe los datos sin enviar una notificación de recepción al remitente. Los datos se envían entonces como bloques (datagramas). UDP es un protocolo no orientado a conexión [62].

Un protocolo define únicamente cómo deben comunicar los equipos, es decir, el formato y la secuencia de datos que van a intercambiar. Por el contrario, un protocolo no define cómo se programa el software para que sea compatible con el protocolo. Esto se denomina implementación o la conversión de un protocolo a un lenguaje de programación [62].

¹⁹ Protocolo de Datagrama de Usuario, por sus siglas en inglés.

²⁰ Protocolo para Transferencia Simple de Correo, por sus siglas en inglés.

²¹ Protocolo de Transferencia de Noticias en Red, por sus siglas en inglés.

Las especificaciones de los protocolos nunca son exhaustivas. Asimismo, es común que las implementaciones estén sujetas a una determinada interpretación de las especificaciones, lo cual genera especificidades de ciertas implementaciones o, aún peor, incompatibilidad o fallas de seguridad [62].

2.9.1. PROTOCOLO HTTP

El protocolo HTTP (Protocolo de transferencia de hipertexto, por sus siglas en inglés) es el protocolo más utilizado en Internet. La versión 0.9 sólo tenía la finalidad de transferir los datos a través de Internet (en particular páginas web escritas en HTML). La versión 1.0 del protocolo (la más utilizada) permite la transferencia de mensajes con encabezados que describen el contenido de los mensajes mediante la codificación MIME²² [63].

El propósito del protocolo HTTP es permitir la transferencia de archivos (principalmente, en formato HTML). Entre un navegador (el cliente) y un servidor web, localizado mediante una cadena de caracteres denominada dirección URL [63].

La comunicación entre el navegador y el servidor, se lleva a cabo en dos etapas: La solicitud HTTP y la respuesta HTTP [63].

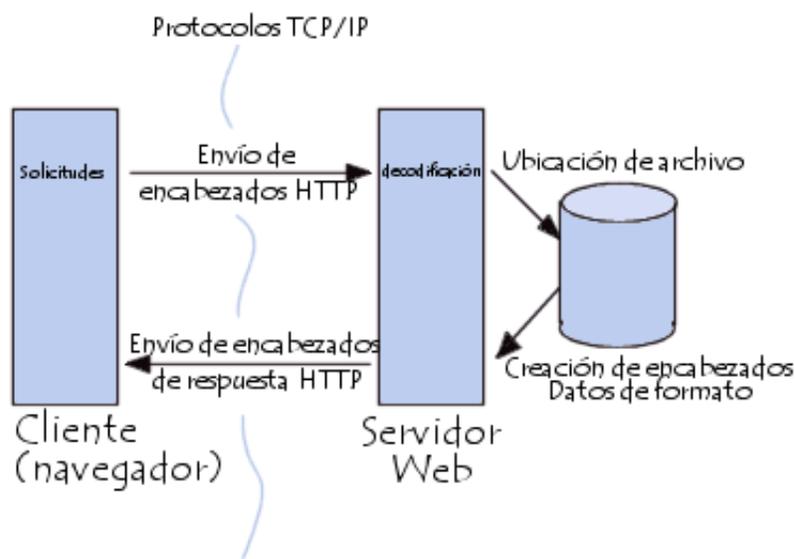


Fig. 2-31. Etapas de comunicación entre navegador y servidor [63].

El navegador realiza una solicitud HTTP y el servidor procesa la solicitud y después envía una respuesta HTTP, como se observa en Fig. 2-31 [63].

2.9.1.1. SOLICITUD HTTP

Una solicitud HTTP es un conjunto de líneas que el navegador envía al servidor. Comprende [63]:

²² Estándar propuesto en 1991 por *Bell Communications* para permitir la inserción de documentos en correos electrónicos. Describe el tipo de contenido del mensaje y el tipo de código usado con encabezados [104]. Extensiones Multipropósito de Correo de Internet, en inglés.

Una línea de solicitud: Una línea que especifica el tipo de documento solicitado, el método que se aplicará y la versión del protocolo utilizada. La línea está formada por tres elementos que deben estar separados por un espacio: el método, la dirección URL y la versión del protocolo utilizada por el cliente (por lo general, HTTP/1.0) [63].

Los campos del encabezado de solicitud: Un conjunto de líneas opcionales que permiten aportar información adicional sobre la solicitud y/o el cliente (navegador, sistema operativo, etc.). Cada una de estas líneas está formada por un nombre que describe el tipo de encabezado, seguido de dos puntos “:” y el valor del encabezado [63].

El cuerpo de la solicitud: Un conjunto de líneas opcionales que deben estar separadas de las líneas precedentes por una línea en blanco y, por ejemplo, permiten que se envíen datos por un comando POST durante la transmisión de datos al servidor utilizando un formulario [63].

Por lo tanto, una solicitud HTTP posee la sintaxis mostrada en Tabla 2-1 (<crLf> significa retorno de carro y avance de línea).

Tabla 2-1. Sintaxis de solicitud HTTP [63].

```
MÉTODO VERSIÓN URL<crLf>
ENCABEZADO: Valor<crLf>
. . . ENCABEZADO: Valor<crLf>
Línea en blanco <crLf>
CUERPO DE LA SOLICITUD
```

En Tabla 2-2, puede verse un ejemplo de una solicitud HTTP.

Tabla 2-2. Ejemplo de una solicitud HTTP [63].

```
GET http://es.kioskea.net HTTP/1.0 Accept : Text/html If-Modified-Since : Saturday, 15-
January-2000 14:37:11 GMT User-Agent : Mozilla/4.0 (compatible; MSIE 5.0; Windows 95)
```

En Apéndice 1, se incluye la lista de comandos HTTP y un listado de las palabras utilizadas como encabezados HTTP.

2.9.1.2. RESPUESTA HTTP

Una respuesta HTTP es un conjunto de líneas que el servidor envía al navegador. Está constituida por [63]:

Una línea de estado: Una línea que especifica la versión del protocolo utilizada y el estado de la solicitud en proceso mediante un texto explicativo y un código. La línea está compuesta por tres elementos que deben estar separados por un espacio: la versión del protocolo utilizada, el código de estado y el significado del código [63].

Los campos del encabezado de respuesta: Un conjunto de líneas opcionales que permiten aportar información adicional sobre la respuesta y/o el servidor. Cada una de estas líneas está compuesta por un nombre que califica el tipo de encabezado, seguido por dos puntos (:) y por el valor del encabezado. Cada una de estas líneas está formada por un nombre que describe el tipo de encabezado, seguido de dos puntos (:) y el valor del encabezado [63].

El cuerpo de la respuesta: Contiene el documento solicitado [63].

Por lo tanto, una respuesta HTTP posee la sintaxis mostrada en Tabla 2-3 (<crLf> significa retorno de carro y avance de línea).

Tabla 2-3. Sintaxis de respuesta HTTP [63].

```
VERSIÓN-HTTP CÓDIGO EXPLICACIÓN<crLf>
ENCABEZADO: Valor<crLf>
. . . ENCABEZADO: Valor<crLf>
Línea en blanco <crLf>
CUERPO DE LA RESPUESTA
```

En Tabla 2-4, puede verse un ejemplo de una respuesta HTTP.

Tabla 2-4. Ejemplo de una respuesta HTTP [63].

```
HTTP/1.0 200 OK Date: Sat, 15 Jan 2000 14:37:12 GMT Server : Microsoft-IIS/2.0 Content-
Type : text/HTML Content-Length : 1245 Last-Modified : Fri, 14 Jan 2000 08:25:13 GMT
```

En Apéndice 1, se encuentra la tabla de encabezados de respuesta HTTP así como la tabla de los códigos de respuesta HTTP.

2.9.2. PROTOCOLO IP

El protocolo IP (Protocolo de Internet, por sus siglas en inglés) es parte de la capa de Internet del conjunto de protocolos TCP/IP. Es uno de los protocolos de Internet más importantes, ya que permite el desarrollo y transporte de datagramas de IP (paquetes de datos), aunque sin garantizar su entrega. En realidad, el protocolo IP procesa datagramas de IP de manera independiente al definir su representación, ruta y envío [64].

El protocolo IP determina el destinatario del mensaje mediante 3 campos: el campo de dirección IP (dirección del equipo); el campo de máscara de subred, que permite al protocolo IP establecer la parte de la dirección IP que se relaciona con la red; y el campo de pasarela predeterminada, que permite al protocolo de Internet saber a qué equipo enviar un datagrama si el equipo de destino no se encuentra en la red de área local [64].

2.9.2.1. DATAGRAMA DEL PROTOCOLO IPV4

Los datos circulan en Internet en forma de datagramas (también conocidos como paquetes). Los datagramas (como se observa esquematizado en Fig. 2-32) son datos encapsulados, es decir, datos a los

que se les agrega un encabezado que contiene información sobre su transporte (como la dirección IP de destino) [64].

Los routers analizan (y eventualmente modifican) los datos contenidos en un datagrama para que puedan transitar [64].

32 bits			
Versión (4 bits)	Longitud del encabezado (4 bits)	Tipo de servicio (8 bits)	Longitud total (16 bits)
Identificación (16 bits)		Indicador (3 bits)	Margen del fragmento (13 bits)
Tiempo de vida (8 bits)	Protocolo (8 bits)	Suma de comprobación del encabezado (16 bits)	
Dirección IP de origen (32 bits)			
Dirección IP de destino (32 bits)			
Datos			

Fig. 2-32. Ejemplo esquematizado de un datagrama [64].

Sus diferentes campos son:

Versión (4 bits): Es la versión del protocolo IP que se está utilizando (actualmente se utiliza la versión 4 IPv4, pero se está en transición hacia la IPv6) para verificar la validez del datagrama [64].

Longitud del encabezado o IHL (por sus siglas en inglés) [4 bits]: Es la cantidad de palabras de 32 bits que componen el encabezado (Importante: el valor mínimo es 5) [64].

Tipo de servicio [8bits]: Indica la forma en la que se debe procesar el datagrama [64].

Longitud total [16 bits]: Indica el tamaño total del datagrama en bytes. El tamaño de este campo es de 2 bytes, por lo tanto, el tamaño total del datagrama no puede exceder los 65536 bytes. Si se lo utiliza junto con el tamaño del encabezado, este campo permite determinar dónde se encuentran los datos [64].

Identificación, indicadores y margen del fragmento: Son campos que permiten la fragmentación de datagramas [64].

TTL (Tiempo de vida, por sus siglas en inglés) [8 bits]: Este campo especifica el número máximo de routers por los que puede pasar un datagrama. Por lo tanto, este campo disminuye con cada paso por un ruteador y cuando alcanza el valor crítico de 0, el ruteador destruye el datagrama. Esto evita que la red se sobrecargue de datagramas perdidos [64].

Protocolo [8 bits]: Este campo, en notación decimal, permite saber de qué protocolo proviene el datagrama. Por ejemplo, ICMP: 1, IGMP²³: 2, TCP: 6, UDP: 17.

Suma de comprobación del encabezado [16 bits]: Este campo contiene un valor codificado en 16 bits que permite controlar la integridad del encabezado para establecer si se ha modificado durante la transmisión. La suma de comprobación es la suma de todas las palabras de 16 bits del encabezado (se excluye el campo suma de comprobación). Esto se realiza de tal modo que cuando se sumen los campos de encabezado (suma de comprobación incluida), se obtenga un número con todos los bits en 1 [64].

Dirección IP de origen [32 bits]: Este campo representa la dirección IP del equipo remitente y permite que el destinatario responda [64].

Dirección IP de destino [32 bits]: Este campo representa la dirección IP del destinatario del mensaje [64].

2.9.2.2. ENRUTAMIENTO IP

El enrutamiento IP es una parte integral de la capa de Internet del conjunto TCP/IP. Este consiste en asegurar el envío de un datagrama IP a través de la red por la ruta más corta. Esta función, la llevan a cabo los routers, es decir, equipos que conectan al menos dos redes [64].

2.9.2.3. PROTOCOLO IPV6

El protocolo IPv6 surge con la intención de reemplazar al IPv4, con su mayor flexibilidad y eficacia para resolver los actuales y nuevos problemas que puedan generarse: aumentar la cantidad de equipos superando la limitación de las IP actuales, reducir el tamaño de las tablas de enrutamiento, simplificar el protocolo para permitir que los routers enruten datagramas de manera más rápida, prestar más atención al tipo de servicio y tráfico en tiempo real, permitir la movilidad de un equipo sin cambiar su dirección, permitir el futuro desarrollo del protocolo, posibilitar la coexistencia pacífica del antiguo protocolo y el nuevo, entre otras [65].

En general, IPv6 no es compatible con IPv4, pero es compatible con todos los demás protocolos de Internet como TCP, UDP, ICMP, IGMP, OSPF²⁴, BGP²⁵ y DNS, requiriendo a veces modificaciones mínimas. La principal innovación de IPv6 es el uso de direcciones más extensas que con IPv4. Están codificadas con 16 bytes y esto permite que se resuelva el problema que hizo que IPv6 sea tan prometedor: brindar un conjunto prácticamente ilimitado de direcciones de Internet. Mientras que IPv4 puede admitir 2^{32} direcciones, IPv6 puede admitir hasta 2^{128} [65].

La mejora más importante de IPv6 es la simplificación de los encabezados de los datagramas. El encabezado del datagrama IPv6 básico (**iError! No se encuentra el origen de la referencia.**) contiene sólo 7 campos (a diferencia de los 14 de IPv4). Este cambio permite que los routers procesen datagramas de manera más rápida y mejore la velocidad en general [65].

²³ Protocolo de Gestión de Grupo de Internet, por sus siglas en inglés.

²⁴ Primer Camino Más Corto, por sus siglas en inglés.

²⁵ Protocolo de Puerta de Enlace de Frontera, por sus siglas en inglés.

La tercera mejora consiste en ofrecer mayor flexibilidad respecto de las opciones. Este cambio es esencial en el nuevo encabezado, ya que los campos obligatorios de la versión anterior ahora son opcionales. De hecho, la manera en la que las opciones están representadas es distinta, dado que permite que los routers simplemente ignoren las opciones que no están destinadas a ellos. Esta función agiliza los tiempos de procesamiento de datagramas [65].

Además, IPv6 brinda más seguridad: La autenticación y confidencialidad constituyen las funciones de seguridad más importantes del protocolo IPv6 [65].

Finalmente, se ha prestado más atención que antes a los tipos de servicios. Si bien el campo Type of services (Tipo de servicios) en el datagrama IPv4 se utiliza pocas veces, el esperado aumento del tráfico multimedia en el futuro demanda que se le otorgue mayor importancia [65].

2.9.3. PROTOCOLO TCP

El TCP (Protocolo de Control de Transmisión) es uno de los principales protocolos de la capa de transporte del modelo TCP/IP. En el nivel de aplicación, posibilita la administración de datos que vienen del nivel más bajo del modelo, o van hacia él, (es decir, el protocolo IP). Cuando se proporcionan los datos al protocolo IP, los agrupa en datagramas IP fijando el campo del protocolo en 6 (para que sepa con anticipación que el protocolo es TCP). TCP es un protocolo orientado a conexión, es decir, que permite que dos máquinas que están comunicadas controlen el estado de la transmisión [66].

Entre las principales características del protocolo TCP se pueden mencionar las siguientes: permite poner nuevamente los datagramas en orden cuando vienen del protocolo IP; permite el monitoreo del flujo de los datos, evitando así la saturación de la red; permite que los datos se formen en segmentos de longitud variada para entregarlos al protocolo IP; y permite multiplexar los datos para que la información que viene de diferentes fuentes (por ejemplo, aplicaciones) en la misma línea, pueda circular simultáneamente (Como se ve en Fig. 2-33). Por último, TCP permite comenzar y finalizar la comunicación amablemente. Las operaciones de multiplexado se realizan empleando el concepto de puertos (o conexiones), es decir, un número vinculado a un tipo de aplicación que, cuando se combina con una dirección de IP, permite determinar en forma exclusiva una aplicación que se ejecuta en una máquina determinada [66].

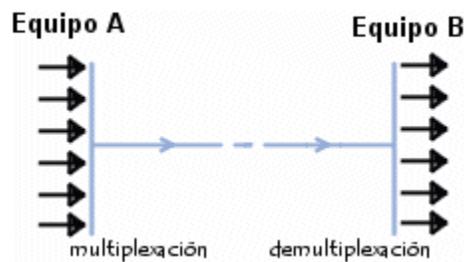


Fig. 2-33. Multiplexado realizado por el protocolo TCP [66].

Con el uso del protocolo TCP, las aplicaciones pueden comunicarse en forma segura (gracias al sistema de acuse de recibo del protocolo TCP) independientemente de las capas inferiores. Esto significa que los routers (que funcionan en la capa de Internet) sólo tienen que enviar los datos en forma de datagramas, sin preocuparse con el monitoreo de datos porque esta función la cumple la capa de transporte (o más específicamente el protocolo TCP) [66].

Durante una comunicación utilizando el protocolo TCP, las dos máquinas deben establecer una relación cliente-servidor. Dicha comunicación es en modo *en línea*, es decir, que la comunicación se realiza en ambas direcciones [66].

Otra función del TCP es la capacidad de controlar la velocidad de los datos usando su capacidad para emitir mensajes de tamaño variable. Estos mensajes se llaman *segmentos* [66].

El protocolo TCP permite garantizar la transferencia de datos confiable, a pesar de que usa el protocolo IP, que no incluye ningún monitoreo de la entrega de datagramas. De hecho, el protocolo TCP tiene un sistema de acuse de recibo que permite al cliente y al servidor garantizar la recepción mutua de datos [66].

2.9.3.1. MODELO CLIENTE-SERVIDOR

TCP es un protocolo orientado a conexión. No hay relaciones maestro/esclavo. Las aplicaciones, sin embargo, utilizan un modelo cliente/servidor en las comunicaciones [67].

Un servidor es una aplicación que ofrece un servicio a usuarios de Internet; un cliente es el que pide ese servicio. Una aplicación consta de una parte de servidor y una de cliente (véase Fig. 2-34), que se pueden ejecutar en el mismo o en diferentes sistemas [67].

Los usuarios invocan la parte cliente de la aplicación, que construye una solicitud para ese servicio y se la envía al servidor de la aplicación que usa TCP/IP como transporte [67].

El servidor es un programa que recibe una solicitud, realiza el servicio requerido y devuelve los resultados en forma de una respuesta. Generalmente un servidor puede tratar múltiples peticiones (múltiples clientes) al mismo tiempo [67].

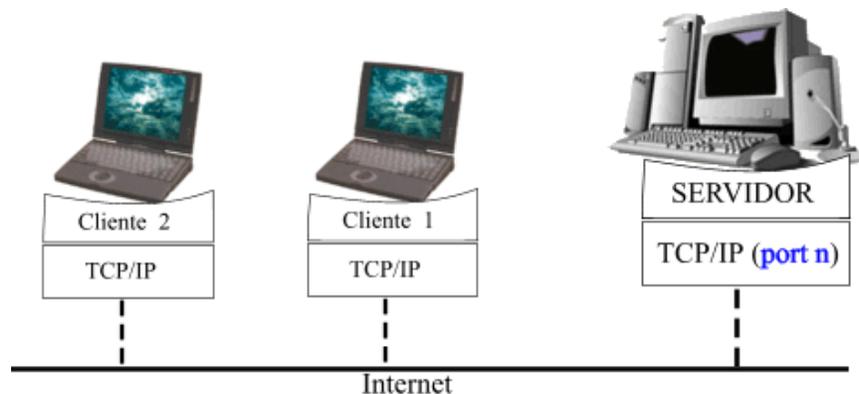


Fig. 2-34. Modelo de la aplicación cliente-servidor [67].

Algunos servidores esperan las solicitudes en puertos bien conocidos de modo que sus clientes saben a qué zócalo IP deben dirigir sus peticiones. El cliente emplea un puerto arbitrario para comunicarse. Los clientes que se quieren comunicar con un servidor que no usa un puerto bien conocido tienen otro mecanismo para saber a qué puerto dirigirse. Este mecanismo podría usar un servicio de registro como Portmap, que utiliza un puerto bien conocido [67].

2.9.3.2. FORMATO DE DATOS EN TCP

Un segmento TCP está formado como se muestra en Fig. 2-35. El significado de cada campo se explica a continuación [66]:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31														
Puerto de origen																Puerto de destino																													
Número de secuencia																																													
Número de acuse de recibo																																													
Margen de los datos				Reservado						U R G						A C K						P R H						S S T						F I N						Ventana					
Suma de control																Puntero urgente																													
Opciones																								Relleno																					
Datos																																													

Fig. 2-35. Formación de los segmentos en TCP [66].

Puerto de origen [16 bits]: Puerto relacionado con la aplicación en curso en la máquina origen [66].

Puerto de destino [16 bits]: Puerto relacionado con la aplicación en curso en la máquina destino [66].

Número de secuencia [32 bits]: Cuando el indicador SYN está fijado en 0, el número de secuencia es el de la primera palabra del segmento actual. Cuando SYN está fijado en 1, el número de secuencia es igual al número de secuencia inicial utilizado para sincronizar los números de secuencia (ISN) [66].

Número de acuse de recibo [32 bits]: El número de acuse de recibo, también llamado número de descarga se relaciona con el número (secuencia) del último segmento esperado y no el número del último segmento recibido [66].

Margen de datos [4 bits]: Esto permite ubicar el inicio de los datos en el paquete. Aquí, el margen es fundamental porque el campo opción es de tamaño variable [66].

Reservado [6 bits]: Un campo que actualmente no está en uso pero se proporciona para el uso futuro [66].

Indicadores [6x1 bit]: Los indicadores representan información adicional [66]:

URG: Si este indicador está fijado en 1, el paquete se debe procesar en forma urgente [66].

ACK: Si este indicador está fijado en 1, el paquete es un acuse de recibo [66].

PSH (PUSH): Si este indicador está fijado en 1, el paquete opera de acuerdo con el método PUSH [66].

RST: Si este indicador está fijado en 1, se restablece la conexión [66].

SYN: El indicador SYN de TCP indica un pedido para establecer una conexión [66].

FIN: Si este indicador está fijado en 1, se interrumpe la conexión [66].

Ventana [16 bits]: Campo que permite saber la cantidad de bytes que el receptor desea recibir sin acuse de recibo [66].

Suma de control (CRC): La suma de control se realiza tomando la suma del campo de datos del encabezado para poder verificar la integridad del encabezado [66].

Puntero urgente (16 bits): Indica el número de secuencia después del cual la información se torna urgente [66].

Opciones (tamaño variable): Diversas opciones [66].

Relleno: Espacio restante después de que las opciones se rellenan con ceros para tener una longitud que sea múltiplo de 32 bits [66].

2.9.3.3. CONEXIÓN TCP

Considerando que este proceso de comunicación, que se produce con la transmisión y el acuse de recibo de datos, se basa en un número de secuencia, las máquinas originadora y receptora (cliente y servidor) deben conocer el número de secuencia inicial de la otra máquina [66].

La conexión establecida entre las dos aplicaciones a menudo se realiza siguiendo el siguiente esquema: los puertos TCP deben estar abiertos; la aplicación en el servidor es pasiva, es decir, que la aplicación escucha y espera una conexión; la aplicación del cliente realiza un pedido de conexión al servidor en el lugar donde la aplicación es abierta pasiva. La aplicación del cliente se considera abierta activa [66].

Las dos máquinas deben sincronizar sus secuencias usando un mecanismo comúnmente llamado negociación en tres pasos que también se encuentra durante el cierre de la sesión [66].

Este diálogo posibilita el inicio de la comunicación porque se realiza en tres etapas, como se puede observar en Fig. 2-36. En la primera, la máquina originadora (el cliente) transmite un segmento donde el indicador SYN está fijado en 1 (para indicar que es un segmento de sincronización), con número de secuencia N llamado número de secuencia inicial del cliente. En la segunda etapa, la máquina receptora (el servidor) recibe el segmento inicial que viene del cliente y luego le envía un acuse de recibo, que es un segmento en el que el indicador ACK está fijado en 1 y el indicador SYN está fijado en 1 (porque es nuevamente una sincronización). Este segmento incluye el número de secuencia de esta máquina (el

servidor), que es el número de secuencia inicial para el cliente. El campo de mayor importancia en este segmento es el de acuse de recibo que contiene el número de secuencia inicial del cliente incrementado en 1. Por último, el cliente transmite un acuse de recibo, que es un segmento en el que el indicador ACK está fijado en 1 y el indicador SYN está fijado en 0 (ya no es un segmento de sincronización). Su número de secuencia ha sido incrementado y el acuse de recibo indica el número de secuencia inicial del servidor incrementado en 1 [66].

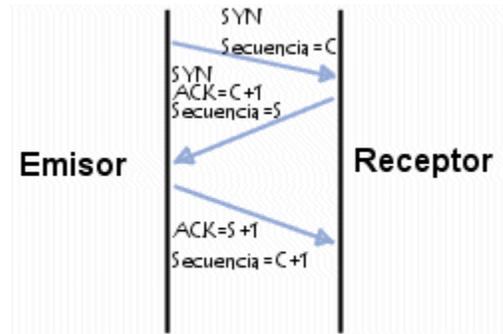


Fig. 2-36. Tres etapas en el inicio de una comunicación TCP [66].

Después de esta secuencia con tres intercambios, las dos máquinas están sincronizadas y la comunicación puede comenzar. Existe una técnica de piratería llamada falsificación de IP, que permite corromper este enlace de aprobación con fines maliciosos [66].

El cliente puede pedir que se termine una conexión del mismo modo que el servidor. Para terminar una conexión se procede de la siguiente manera: una de las máquinas envía un segmento con el indicador FIN fijado en 1, y la aplicación se pone en estado de espera, es decir que deja de recibir el segmento actual e ignora los siguientes. Después de recibir este segmento, la otra máquina envía un acuse de recibo con el indicador FIN fijado en 1 y sigue enviando los segmentos en curso. Después de esto, la máquina informa a la aplicación que se ha recibido un segmento FIN y luego envía un segmento FIN a la otra máquina, que cierra la conexión [66].

2.9.4. SSH

El protocolo SSH (Shell Segura, por sus siglas en inglés) es un método para el inicio de sesión remoto seguro desde una computadora a otra. Proporciona varias opciones alternativas para una autenticación sólida y protege la seguridad e integridad de las comunicaciones con un cifrado sólido. Es una alternativa segura a los protocolos de inicio de sesión no protegidos (como Telnet, Rlogin) y métodos de transferencia de archivos inseguros (como FTP) [68].

2.9.4.1. USOS TÍPICOS

El protocolo se utiliza en redes corporativas para [68]:

- Proporcionar acceso seguro para los usuarios y procesos automatizados.
- Realizar transferencias de archivos interactivas y automatizadas.
- Emitir comandos remotos.
- Gestionar la infraestructura de red y otros componentes del sistema de misión crítica.

2.9.4.2. FUNCIONAMIENTO

El protocolo funciona en el modelo cliente-servidor, lo que significa que la conexión la establece el cliente SSH que se conecta al servidor SSH (véase Fig. 2-37). El cliente SSH controla el proceso de configuración de la conexión y utiliza la criptografía de clave pública para verificar la identidad del servidor SSH. Después de la fase de configuración, el protocolo SSH utiliza un cifrado simétrico fuerte y algoritmos de *hashing*²⁶ para garantizar la privacidad e integridad de los datos que se intercambian entre el cliente y el servidor [68].

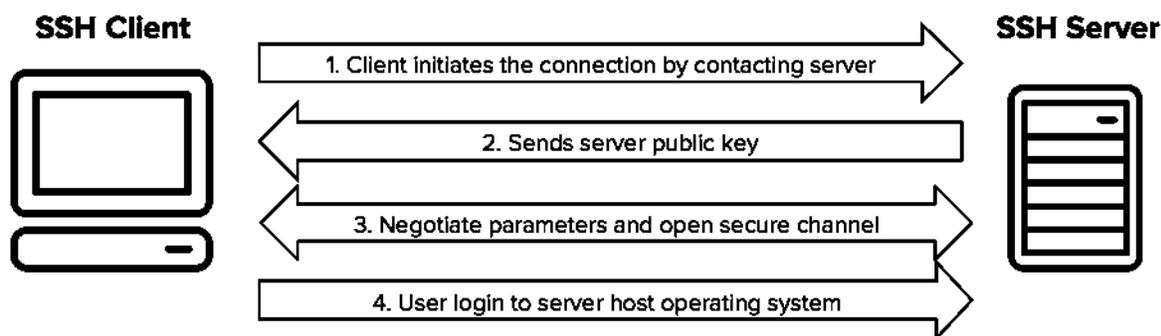


Fig. 2-37. Flujo de configuración simplificado de una conexión de SSH [68].

2.9.4.3. AUTENTICACIÓN FUERTE CON CLAVES SSH

Hay varias opciones que se pueden usar para la autenticación del usuario. Los más comunes son las contraseñas y la autenticación de clave pública [68].

El método de autenticación de clave pública se utiliza principalmente para la automatización y, a veces, por los administradores del sistema para el inicio de sesión único. La idea es tener un par de claves criptográficas (clave pública y clave privada) y configurar la clave pública en un servidor para autorizar el acceso y otorgar a cualquier persona que tenga una copia de la clave privada el acceso al servidor. Las claves utilizadas para la autenticación se denominan claves SSH. La autenticación de clave pública también se usa con tarjetas inteligentes, como las tarjetas CAC y PIV utilizadas por el gobierno de los Estados Unidos [68].

El uso principal de la autenticación basada en clave es habilitar la automatización segura. Las transferencias de archivos de shell seguras y automatizadas se utilizan para integrar a la perfección aplicaciones y también para sistemas automatizados y administración de configuración [68].

2.9.4.4. CIFRADO FUERTE Y PROTECCIÓN DE INTEGRIDAD

Una vez que se ha establecido una conexión entre el cliente y el servidor SSH, los datos que se transmiten se cifran de acuerdo con los parámetros negociados en la configuración. Durante la negociación, el cliente y el servidor acuerdan el algoritmo de cifrado simétrico que se utilizará y generan la clave de cifrado que se utilizará. El tráfico entre las partes que se comunican está protegido con algoritmos de

²⁶ Algoritmos que permiten aumentar la velocidad de búsqueda sin necesidad de tener los elementos ordenados [106].

encriptación sólidos de la industria (como AES²⁷), y el protocolo SSH también incluye un mecanismo que garantiza la integridad de los datos transmitidos mediante el uso de algoritmos hash estándar (como SHA²⁸) [68].

2.9.5. MQTT

MQTT es la sigla de Message Queue Telemetry Transport (Transporte de Telemetría de la Cola de Mensajes). Es un protocolo usado para la comunicación machine-to-machine (M2M²⁹) en el Internet de las Cosas. Este protocolo está orientado a la comunicación de sensores, debido a que consume muy poco ancho de banda y puede ser utilizado en la mayoría de los dispositivos empotrados con pocos recursos (CPU, RAM, etc.). Un ejemplo de uso de este protocolo es la aplicación de Facebook Messenger tanto para Android y IOS. La arquitectura de MQTT sigue una topología de estrella, con un nodo central que hace de servidor o *bróker*³⁰ con una capacidad de hasta 10000 clientes. El *bróker* es el encargado de gestionar la red y de transmitir los mensajes, para mantener activo el canal, los clientes mandan periódicamente un paquete (PINGREQ) y esperan la respuesta del *bróker* (PINGRESP). La comunicación puede ser cifrada entre otras muchas opciones [69].

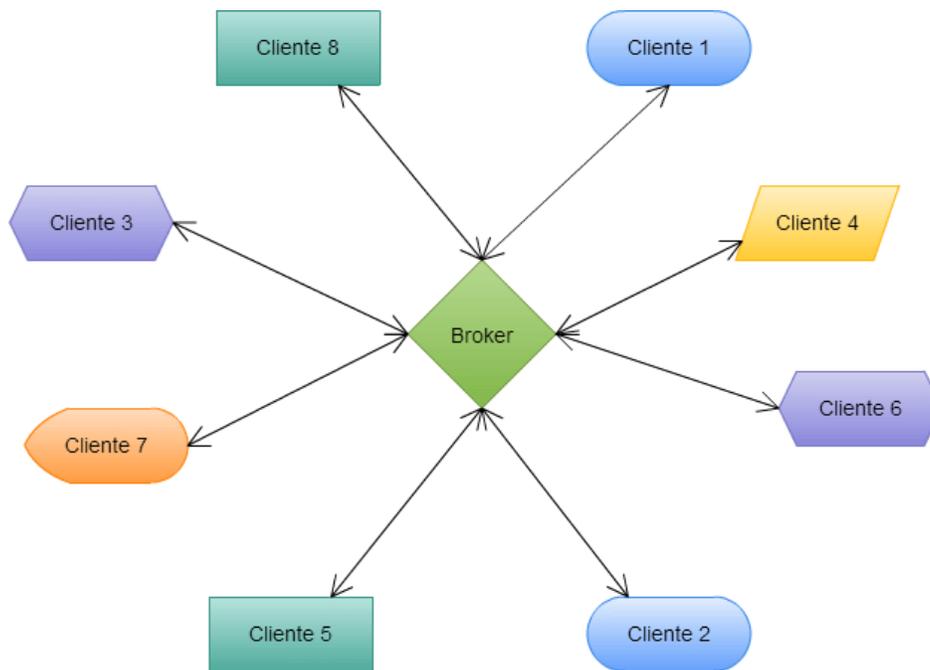


Fig. 2-38. Arquitectura de MQTT con topología de estrella [69].

La comunicación se basa en *topics* (temas), que el cliente que publica el mensaje crea y los nodos que deseen recibirlo deben subscribirse a él. La comunicación puede ser de uno a uno, o de uno a muchos (como se ejemplifica en Fig. 2-38). Un *topic* se representa mediante una cadena y tiene una estructura

²⁷ Estándar de Cifrado Avanzado, por sus siglas en inglés.

²⁸ Algoritmo de Hash Estándar, por sus siglas en inglés.

²⁹ De máquina a máquina, por sus siglas en inglés.

³⁰ Del inglés *bróker*. Agente intermediario en operaciones financieras, comerciales o de comunicación digital.

jerárquica. Cada jerarquía se separa con '/'. Por ejemplo, "edificio1/planta5/sala1/temperatura" o "/edificio3/planta0/sala3/ruido". De esta forma se pueden crear jerarquías de clientes que publican y reciben datos: uno puede suscribirse a un *topic* concreto o a varios [69].

2.10. BASE DE DATOS

Una base de datos es un conjunto de datos almacenados en memoria externa que están organizados mediante una estructura de datos. Cada base de datos ha sido diseñada para satisfacer los requisitos de información de una empresa u otro tipo de organización como, por ejemplo, una universidad o un hospital [70].

Una base de datos se puede percibir como un gran almacén de datos con una estructura previamente definida, y que se utiliza al mismo tiempo por distintos usuarios. Antes de existir las bases de datos, los programas debían manejar los datos que se encontraban almacenados en ficheros desconectados y con información redundante. En una base de datos todos los datos se integran con una mínima cantidad de duplicidad. De este modo, la base de datos no pertenece a un solo departamento, sino que se comparte por toda la organización. Además, la base de datos no sólo contiene los datos de la organización, también almacena una descripción de dichos datos. Esta descripción es lo que se denomina metadatos, se almacena en el diccionario de datos o catálogo y es lo que permite que exista lo que se denomina independencia de datos lógica-física [70].

Dependiendo de la estructura con la cual se almacenan los datos, pueden clasificarse en dos principales categorías: Bases de datos relacionales y Bases de datos no relacionales.

2.10.1. BASES DE DATOS RELACIONALES

Las bases de datos relacionales son conocidas popularmente como *bases de datos SQL*. SQL (Structured Query Language) o Lenguaje de Consulta Estructurada, es un lenguaje utilizado para crear, manipular y controlar datos de las bases de datos relacionales en un solo lenguaje. SQL integra Data manipulation Language, Data Definition Language y Data Control Language en uno mismo. Cuentan con su propio lenguaje para consultar, insertar, actualizar y eliminar registros de la base de datos. Se componen de una o varias tablas y cada tabla puede contener múltiples relaciones (Véase Fig. 2-39) [71].

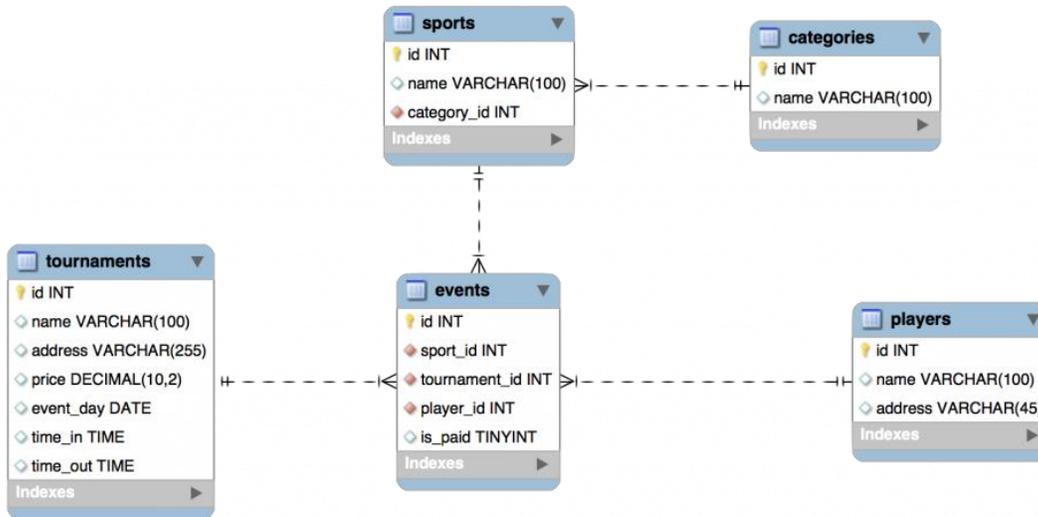


Fig. 2-39. Estructura de una base de datos SQL por medio de tablas [71].

Las tablas representan la entidad de cada modelo de datos. Las relaciones permiten garantizar la persistencia de datos, evita la duplicidad de datos y controla la eliminación en cascada. Para ingresar un nuevo evento, bastaría con ingresar un nuevo registro en la tabla *events* [71].

SQL cuenta con un lenguaje de alto nivel que permite manipular la base de datos casi de cualquier forma. Por su manera de operar ayuda a mantener la integridad de los datos. Su uso es más frecuente es en sistemas donde se necesite cuidar la integridad de datos como los bancos, sistemas de ventas, hospitales, CMS como WordPress. Ejemplos de sistemas de gestión de base de datos son: MySQL, MariaDB, PostgreSQL [71].

2.10.2. BASES DE DATOS NO RELACIONALES

También existen otros mecanismos de almacenamientos alternativos a las bases de datos relacionales que se definen por tablas. Este tipo de base de datos, se apoyan en otros formatos. Las bases de datos no relacionales (conocidas popularmente como NOSQL) son más adecuados para aquellos que manejan grandes volúmenes de datos. Es por eso que empresas como Facebook, Google, Amazon, Twitter, entre otros han popularizado estos tipos de base de datos [71].

Esas empresas tenían que enfrentarse a grandes desafíos motivados a la alta demanda de internet. Donde surge la necesidad de proporcionar información de grandes volúmenes de datos e indeterminado número de usuarios en el menor tiempo posible. Esas compañías se dieron cuenta que podían sacrificar la consistencia de los datos que garantiza una base de datos relacional para poder obtener un mejor rendimiento y ofrecer soluciones en tiempo real [71].

Se pueden usar en soluciones donde se requiere interactuar con datos en tiempo real como en un chat, para análisis estadísticos donde procesas mucha información o incluso en blogs [71].

Dependiendo de la forma en la que almacenen la información, se pueden encontrar varios tipos distintos de bases de datos NoSQL [72].

2.10.2.1. BASES DE DATOS CLAVE-VALOR

Son el modelo de base de datos NoSQL más popular, además de ser la más sencilla en cuanto a funcionalidad. En este tipo de sistema, cada elemento está identificado por una llave única, lo que permite la recuperación de la información de forma muy rápida, información que habitualmente está almacenada como un objeto binario (BLOB³¹). Se caracterizan por ser muy eficientes tanto para las lecturas como para las escrituras [72].

Un diagrama de una base de datos clave-valor, conocida también como *key-value*, se muestra en Fig. 2-40. En él, se realiza una consulta del valor (values) relacionado con cada uno de los nombres (keys).

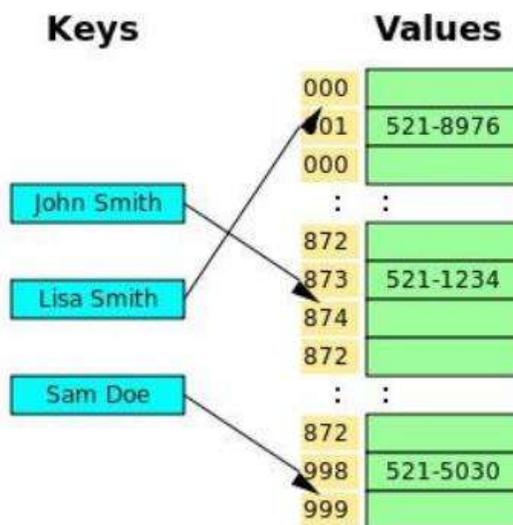


Fig. 2-40. Diagrama de ejemplo de una base de datos key-value [72].

Algunos ejemplos de este tipo de manejadores de bases de datos son Cassandra, BigTable o HBase [72].

2.10.2.2. BASES DE DATOS DOCUMENTALES

Este tipo de manejador de bases de datos almacena la información como un documento, generalmente utilizando para ello una estructura simple como JSON o XML y donde se utiliza una clave única para cada registro. Este tipo de implementación permite, además de realizar búsquedas por clave-valor, realizar consultas más avanzadas sobre el contenido del documento [72].

Son las bases de datos NoSQL más versátiles. Se pueden utilizar en gran cantidad de proyectos, incluyendo muchos que tradicionalmente funcionarían sobre bases de datos relacionales. Algunos ejemplos de este tipo de manejador de bases de datos, son MongoDB o CouchDB. Un ejemplo de una base de datos documental, se ilustra en el diagrama de Fig. 2-41 [72]. En ese diagrama, se observa cómo un documento que contiene información en formato JSON es almacenado en la base de datos no relacional, donde cada campo del JSON corresponde a una columna.

³¹ Objeto Binario Grande, por sus siglas en inglés.

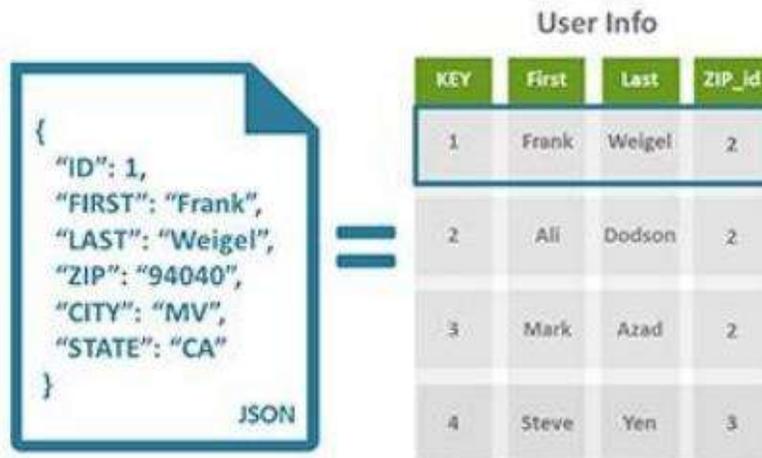


Fig. 2-41. Ejemplo en diagrama de una base de datos documental [72].

2.10.2.3. BASES DE DATOS EN GRAFO

En este tipo de bases de datos, la información se representa como nodos de un grafo y sus relaciones con las aristas del mismo, de manera que se puede hacer uso de la teoría de grafos para recorrerla. Para sacar el máximo rendimiento a este tipo de bases de datos, su estructura debe estar totalmente normalizada, de forma que cada tabla tenga una sola columna y cada relación dos [72].

Este tipo de bases de datos ofrece una navegación más eficiente entre relaciones que en un modelo relacional. Algunos ejemplos de este tipo son Neo4j, InfoGrid o Virtuoso. En el diagrama mostrado en Fig. 2-42, se observa la normalización que tiene que tener las bases de datos en grafo; es decir, que cada tabla tiene solamente una columna y cada relación, tan sólo dos [72].

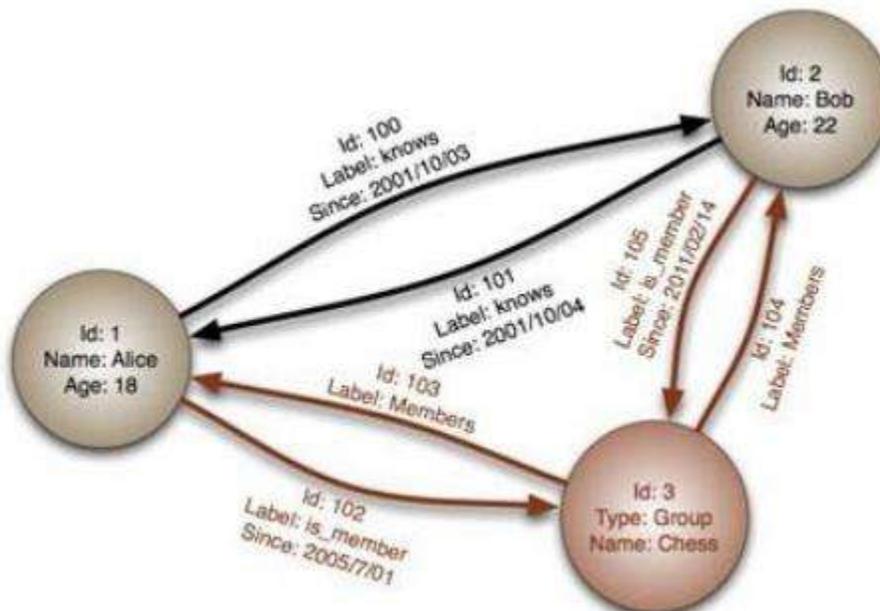


Fig. 2-42. Diagrama de base de datos en grafo [73].

2.10.2.4. BASES DE DATOS ORIENTADAS A OBJETOS

En este tipo, la información se representa mediante objetos, de la misma forma que son representados en los lenguajes de programación orientada a objetos (POO) como ocurre en JAVA, C# o Visual Basic .NET [72].

Algunos ejemplos de este tipo de bases de datos son Zope, Gemstone o Db4o [72].

CAPÍTULO 3. DISEÑO Y DESCRIPCIÓN DEL HARDWARE

3.1. INTRODUCCIÓN

En este capítulo se describen todos los elementos tangibles que conforman el proyecto: la instalación hidropónica, el System on Chip, la cámara serial, los diferentes sensores utilizados, los actuadores, el servidor en una SBC y otros circuitos integrados. De la programación y las interacciones entre el cliente desde una plataforma web, el servidor y los microcontroladores, se encuentra la información en Capítulo 4.

El sistema en su conjunto, está integrado por dos SoC NodeMCU v3, una cámara serial, una Raspberry Pi 3B, una tira de LED's RGB de 5 metros, una bomba de agua sumergible de 6 W, una instalación hidropónica de PVC, un sensor DTH11 de temperatura y humedad, un sensor de PH modelo PH_4502C, un sensor de CE y un sensor de luminosidad. El diagrama a bloques del sistema puede observarse en Fig. 3-1. En él, puede observarse que la Raspberry Pi 3B es el intermediario entre los microcontroladores y el navegador web. Uno de los microcontroladores controla exclusivamente la cámara web, mientras que el otro, controla y se comunica con todos los actuadores y sensores del sistema. Ambos microcontroladores intercambian información con los servicios alojados en la Raspberry Pi 3B. El usuario, desde un navegador web, solamente establece comunicación directa con el servidor web, y obtiene los datos y actualizaciones provenientes de los microcontroladores, a través de la página web y la base de datos.

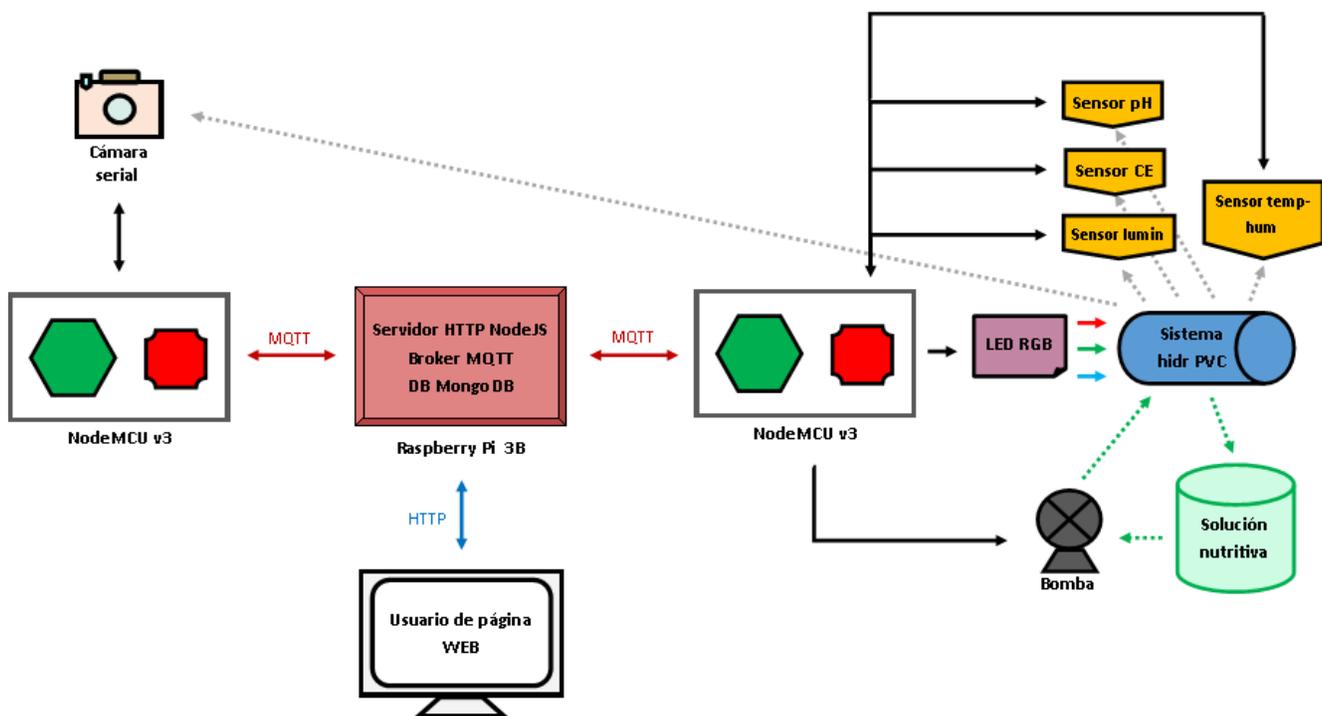


Fig. 3-1. Diagrama a bloques del hardware del sistema.

En Fig. 3-2 se encuentra información adicional como acotaciones y explicación de algunas figuras utilizadas en el diagrama de bloques de Fig. 3-1.

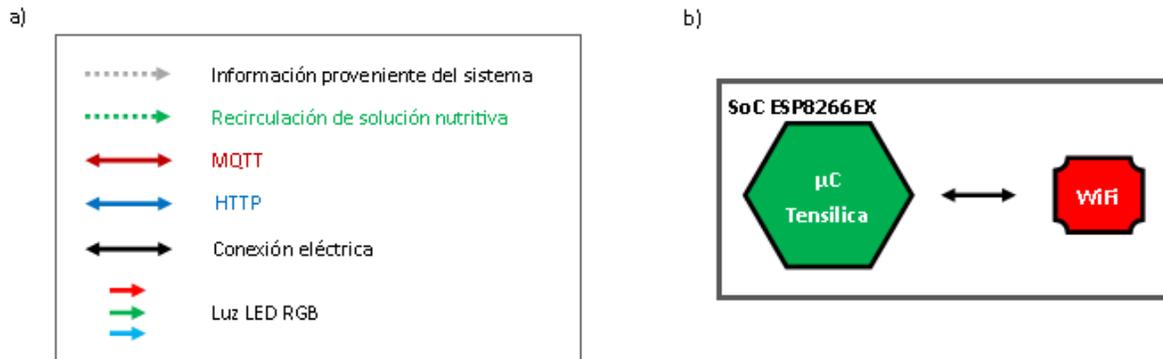


Fig. 3-2. a) Acotaciones de Fig. 3-1. b) Composición interna del SoC representado en Fig. 3-1.

3.2. INSTALACIÓN HIDROPÓNICA PVC

Para la construcción del sistema de hidroponía se utilizó un tubo de PVC sanitario de 110cm de diámetro de 6m de largo, el cual fue cortado a la mitad para obtener 2 piezas de 3m. Se utilizaron coples y tapas para cerrar ambas partes. A cada tubo se le realizaron 11 perforaciones de 5cm de diámetro, siendo un total de 22 orificios en todo el sistema hidropónico de PVC. Los tubos fueron colocados uno encima de otro, dejando el espacio suficiente entre ellos para el apropiado desarrollo de las plantas.

Un diagrama de la distribución del sistema se muestra en Fig. 3-3. Se observan también los canales de suministro y desagüe de líquido.

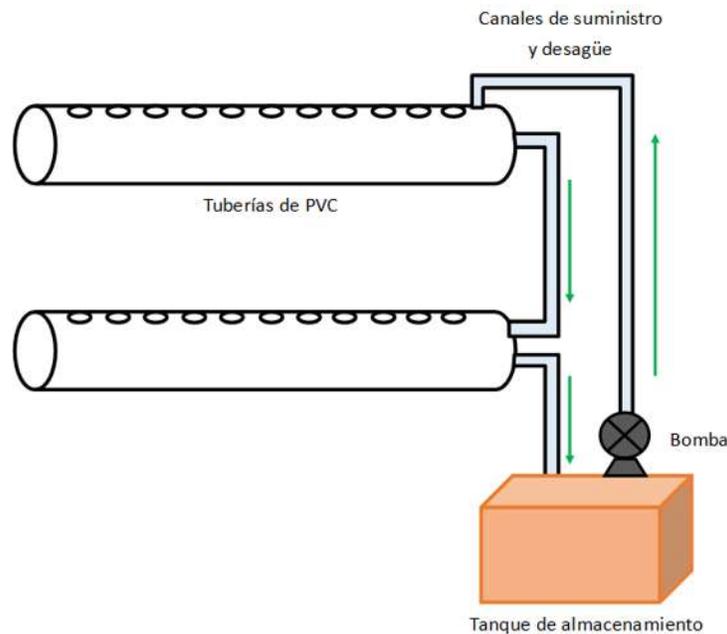


Fig. 3-3. Diseño de la estructura de PVC utilizada para el sistema hidropónico.

El sistema de desagüe se construyó de tal forma que los tubos siempre mantengan un nivel de líquido inferior a la mitad de su capacidad máxima, con el propósito de que las plantas colocadas en los orificios junto con su espuma agrícola, mantengan una parte de sus raíces en contacto con el líquido nutritivo y otra en contacto con el aire, como se ilustra en Fig. 3-4.

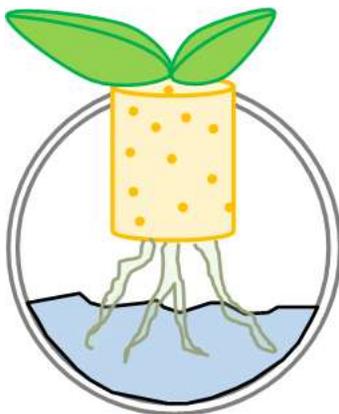


Fig. 3-4. Vista transversal del modelo de la tubería PVC que permite la aireación de las raíces.

El líquido nutritivo se recircula en el sistema hidropónico proviniendo de un tanque de almacenamiento, a través de la bomba, y terminando en el mismo tanque.

3.2.1. SOLUCIÓN NUTRITIVA

Para que las plantas puedan desarrollarse apropiadamente, se utiliza una solución nutritiva especialmente diseñada para hortalizas. La solución marca Steiner está diseñada para la producción de hortalizas de hoja como lechugas orejonas, lechugas italianas, lechugas hoja de roble, acelgas, escarolas, espinacas, microgreens o germinados, forraje verde, entre otras [73].

Esta marca de solución nutritiva soluble en agua, en su presentación que rinde hasta 500 litros, contiene dos paquetes de sales A y B, los cuales incluyen los macronutrientes y micronutrientes diseñados para estas hortalizas, en un contenido de 800g. Además, incluye un paquete más pequeño de quelatos micronutrientes. Entre estas sales se encuentran, en las proporciones adecuadas para este tipo de cultivos, fósforo, magnesio, potasio, nitrógeno, calcio, azufre, manganeso, cobre, molibdeno, zinc, hierro y boro [73].

Con cada uno de los paquetes de sales, debe prepararse una solución madre de 5 litros (los micronutrientes se preparan en la solución madre del paquete A). Estas soluciones tienen una caducidad de 2 años a partir de su preparación, en condiciones no expuestas al sol. Si la hortaliza se encuentra en las primeras etapas de germinación y hasta los 30 días, deben utilizarse 5ml de la solución madre A y 5ml de la solución madre B por cada litro de agua a preparar. Si la planta está ya en la etapa posterior de desarrollo, deben emplearse 10ml de A y 10ml de B por cada litro de agua a usar [73].

La solución nutritiva recién preparada tendrá un pH de entre 6.5 a 6.7, dependiendo del agua que se haya utilizado para prepararla. En caso de requerir ajustar a la baja el pH se puede utilizar vinagre blanco

(0.5ml por cada litro de agua) o ácido fosfórico (1ml por cada 5 litros de agua). Debe evitarse a toda costa el uso del ácido muriático con este fin [73].

Se recomienda preparar la solución en un ambiente sin viento, a fin de evitar inhalar los polvos; preparar las soluciones madres una vez abiertas las bolsas de las sales y guardar las soluciones en un lugar fresco, evitando el contacto directo con el sol [73].

3.2.2. SEMILLAS

Las semillas utilizadas para los cultivos son las *Semillas de Lechuga Italiana Black Seeded Simpson* de la marca *Hortaflor*, mostradas en Fig. 3-5.



Fig. 3-5. Semillas de lechuga italiana marca Hortaflor.

Para su germinación requieren luz solar, una profundidad de siembra de 2cm y germinan dentro de los primeros 4 y 6 días, según Hortaflor. Necesitan una distancia de 25 cm entre planta y planta (ya en su etapa de desarrollo), y se pueden cosechar dentro de 30 y 40 días luego de su germinación [74].

3.3. SYSTEM ON CHIP (SOC)

La parte encargada de realizar el control de los actuadores, obtener la información de los sensores y realizar las operaciones, es un microcontrolador. El utilizado en este proyecto es el Tensilica L106, del fabricante Espressif. Este microcontrolador está integrado dentro de un SoC modelo ESP8266EX; que a su vez, se encuentra integrado dentro del módulo ESP-12E; mismo que forma parte de la tarjeta de desarrollo NodeMCU v1.0/V3, como se muestra en Fig. 3-6.

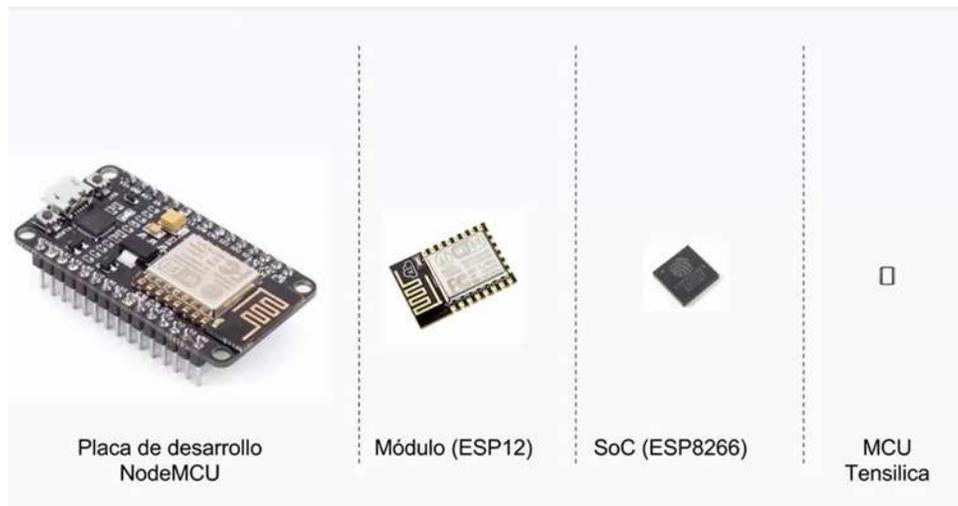


Fig. 3-6. Composición de la placa de desarrollo NodeMCU [28].

El Tensilica L106 es un microcontrolador de 32 bits, que trabaja a una velocidad de 80MHz, aunque puede llegar a los 160MHz. Tiene un consumo extra bajo de energía. Forma parte del SoC ESP8266EX (véase el recuadro rojo de Fig. 3-7) [28].

El Sistema Operativo en Tiempo Real (RTOS) y el stack de WiFi permite que el 80% de la capacidad de procesamiento esté disponible para las tareas de programación y desarrollo del usuario [75].

Entre las características principales del SoC empleado (ver diagrama funcional en Fig. 3-7) se encuentran [28]:

- Incorpora una MCU de 32 bits de bajo consumo (El Tensilica L106).
- Módulo WiFi de 2.4GHz.
- RAM de 32KB para instrucciones y 80KB para datos.
- 1 entrada analógica con resolución de 10 bits (ADC).
- 17 pines de entrada y salida de propósito general (GPIO).
- Núcleo a 80MHz de velocidad (160MHz si se le hace el *overclocking*³²)

³² Incrementar la frecuencia de reloj de los componentes electrónicos, aumentando su rendimiento y velocidad de funcionamiento, a costa de un mayor consumo energético, mayor calentamiento y posible reducción de la vida útil del componente [106].

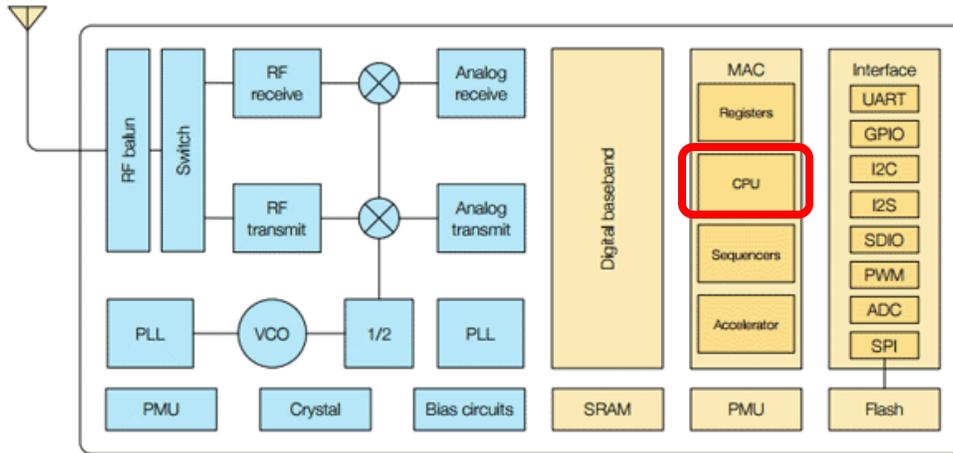


Fig. 3-7. Diagrama funcional del SoC ESP8266EX [75].

El área color azul de Fig. 3-7 está dedicada a las funciones inalámbricas de WiFi, mientras que la amarilla, proporciona el procesamiento de información, almacenamiento, procesos y microcontrolador.

Para subsanar la memoria tan limitada que tiene, parte de los GPIO pueden ser utilizados para conectarse a una memoria Flash externa.

El SoC ESP8266EX puede alimentarse con corriente directa de entre 2.5v y 3.6v. El diagrama de pines del ESP8266EX se muestra en Fig. 3-8 [75].

El listado completo de sus demás especificaciones técnicas puede consultarse en Apéndice 2.

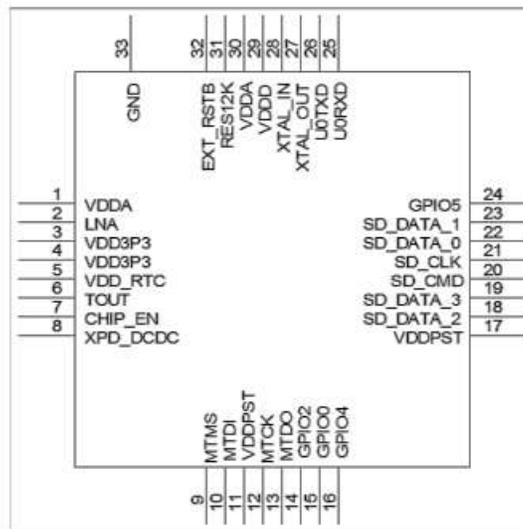


Fig. 3-8. Diagrama de pines del SoC ESP8266EX [75].

Existen diversos módulos que contienen al ESP8266EX. La principal diferencia entre ellos es el acceso a los pines. El utilizado por el NodeMCU v1.0/V3 es el *ESP-12E*, mostrado en Fig. 3-9. Básicamente este módulo incorpora la memoria flash para almacenar los programas o sketches y la antena WiFi. También está pensado para facilitar el acceso a los pines y demás conectores del SoC y del microcontrolador [28].



Fig. 3-9. Módulo ESP-12E [28].

Las placas de desarrollo tienen como objetivo facilitar el desarrollo de los proyectos. Las principales características del NodeMCU son [28]:

- Convertidor Serie-USB para programar y alimentar a través del puerto USB de la computadora.
- Fácil acceso a los pines.
- Pines de alimentación para sensores y componentes.
- LEDs para indicar estado.
- Botón de reset.

A este nivel de tarjeta de desarrollo, ya no hay que preocuparse por averiguar cómo cargar el programa o conectar los pines. Sólo se necesita una computadora, un cable USB y un entorno de desarrollo para programar el microcontrolador [28].

NodeMCU es una placa de hardware abierto, por lo que cualquier fabricante puede crear su propia distribución. En Fig. 3-10 se muestran las 3 versiones disponibles en el mercado. No obstante, todas las NodeMCU se basan en el módulo ESP-12 o ESP-12E, ambos basados en ESP8266. Las diferencias serán entonces en el número de pines al que permiten el acceso y el tamaño de cada placa [28].

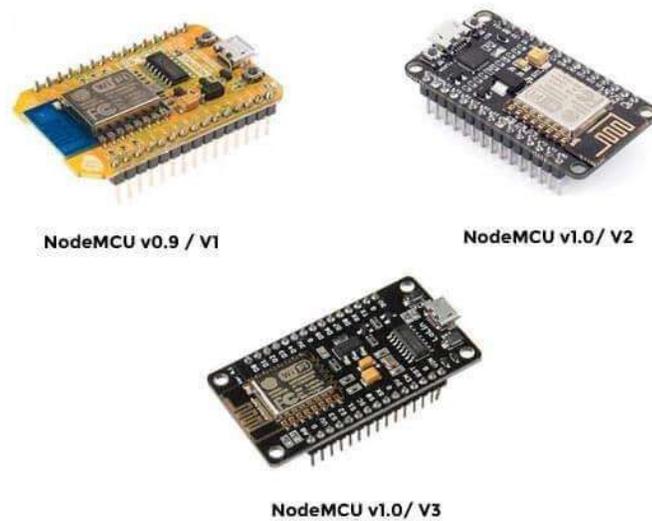


Fig. 3-10. Variantes en el mercado del NodeMCU [28].

La empleada en este proyecto es la V3, que técnicamente no es una especificación oficial por parte de NodeMCU. Se trata de una versión creada por *LoLin*, que aporta ciertas ventajas respecto a las otras versiones [28].

El cambio principal es que la V3 monta un adaptador USB-serial CH340G en lugar del CP2102 que tiene la V2. El fabricante asegura que el puerto USB sea más robusto. Por otro lado, se proporcionan los dos pines reservados que tenía la V2, para obtener de la tarjeta GND y VUSB a 5v, directo del puerto USB [76].

En Apéndice 3 se muestran los diagramas de pines del NodeMCU V2; y del V3, visualizando éste último por la parte superior, con el puerto USB hacia abajo y la antena del módulo WiFi hacia arriba.

A pesar de que el NodeMCU tiene 12 pines digitales, algunos de ellos son utilizados para conectarse a la memoria flash externa del ESP8266EX. Descartando utilizar estos dos como pines digitales, así como los dedicados para la transmisión y recepción serial, quedan disponibles los pines del D0 al D8 como pines digitales de entrada y salida, como se muestra en Fig. 3-11 [28].

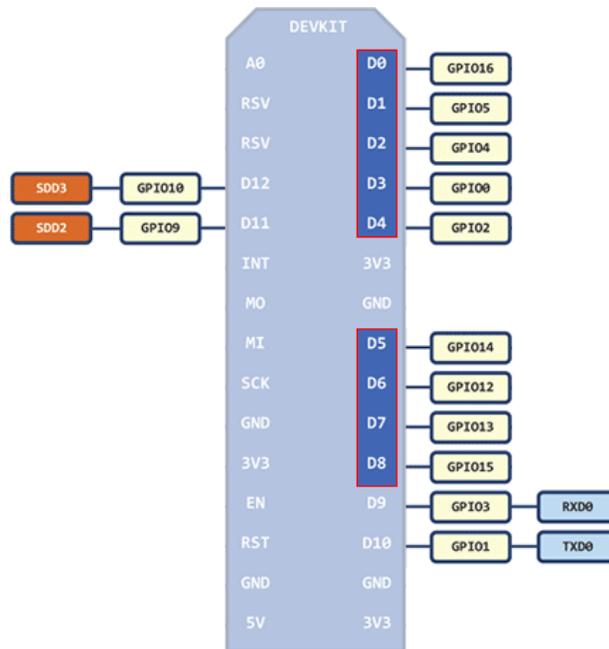


Fig. 3-11. Pines digitales recomendados para utilizarse en el NodeMCU V2 y V3 [28].

El NodeMCU tiene un solo pin analógico que admite valores de entre 0v y 3.3v de CD, con una resolución de 10 bits [28].

El modelo de programación del NodeMCU es similar al de Node.js, sólo que en lenguaje Lua. Es asíncrono y manejado por eventos. Muchas funciones reciben funciones de devolución de llamada como parámetros [77].

3.3.1. CLOUD BUILD SERVICE

El NodeMCU no tiene un firmware preinstalado. Debe instalársele uno para que funcione. Afortunadamente, es el mismo equipo desarrollador de NodeMCU quien proporciona una herramienta en línea para elegir un firmware personalizando con las bibliotecas necesarias. La herramienta llamada “NodeMCU custom builds” se encuentra disponible en <https://nodemcu-build.com/>. Esta página solicita una dirección de correo electrónico a la cuál enviarán el enlace de descarga del firmware, una vez terminado el proceso de personalización. En la página se permite elegir entre un firmware maestro, uno enfocado al desarrollo o uno diseñado para tarjetas de desarrollo con una memoria flash pequeña (de 512kB). Consecuentemente, se debe elegir entre una lista de módulos que se deseen incluir al firmware (listado con las elecciones de módulos a usar realizadas, mostrado en Fig. 3-12). Finalmente, se debe elegir si se desea tener soporte para una o todas de las opciones: TLS, debug y soporte para tarjetas SD³³; y comenzar la construcción del firmware.

³³ Tarjeta de memoria Digital Segura, por sus siglas en inglés.

- | | | | |
|-----------------------------------------------------------------------------|-------------------------------------------------------------|------------------------------------------------------------------|-------------------------------------------------------------|
| <input checked="" type="checkbox"/> ADC 📄 | <input type="checkbox"/> end user setup 📄 | <input type="checkbox"/> perf 📄 | <input type="checkbox"/> Switec 📄 |
| <input type="checkbox"/> ADS1115 📄 | <input checked="" type="checkbox"/> file 📄 | <input checked="" type="checkbox"/> PWM 📄 | <input type="checkbox"/> TCS34725 📄 |
| <input type="checkbox"/> ADXL345 📄 | <input type="checkbox"/> gdbstub 📄 | <input type="checkbox"/> RC (no docs) | <input type="checkbox"/> TM1829 📄 |
| <input type="checkbox"/> AM2320 📄 | <input checked="" type="checkbox"/> GPIO 📄 | <input type="checkbox"/> rswitch 📄 | <input checked="" type="checkbox"/> timer 📄 |
| <input type="checkbox"/> APA102 📄 | <input type="checkbox"/> HDC1080 📄 | <input type="checkbox"/> rotary 📄 | <input type="checkbox"/> TSL2561 📄 |
| <input checked="" type="checkbox"/> bit 📄 | <input type="checkbox"/> HMC5883L 📄 | <input type="checkbox"/> RTC fifo 📄 | <input type="checkbox"/> U8G2 📄 |
| <input type="checkbox"/> Bloom filter,
requires crypto 📄 | <input checked="" type="checkbox"/> HTTP 📄 | <input type="checkbox"/> RTC mem 📄 | <input checked="" type="checkbox"/> UART 📄 |
| <input type="checkbox"/> BME280 📄 | <input type="checkbox"/> HX711 📄 | <input checked="" type="checkbox"/> RTC time 📄 | <input type="checkbox"/> UCG 📄 |
| <input type="checkbox"/> BME680 📄 | <input type="checkbox"/> I ² C 📄 | <input type="checkbox"/> Si7021 📄 | <input type="checkbox"/> websocket 📄 |
| <input type="checkbox"/> BMP085 📄 | <input type="checkbox"/> L3G4200D 📄 | <input type="checkbox"/> Sigma-delta 📄 | <input checked="" type="checkbox"/> WiFi 📄 |
| <input type="checkbox"/> CoAP 📄 | <input type="checkbox"/> MCP4725 📄 | <input checked="" type="checkbox"/> SJSON 📄 | <input type="checkbox"/> WiFi monitor 📄 |
| <input type="checkbox"/> color utils 📄 | <input checked="" type="checkbox"/> mDNS 📄 | <input checked="" type="checkbox"/> SNTP 📄 | <input type="checkbox"/> WPS 📄 |
| <input type="checkbox"/> Cron 📄 | <input checked="" type="checkbox"/> MQTT 📄 | <input type="checkbox"/> Somfy 📄 | <input type="checkbox"/> WS2801 📄 |
| <input type="checkbox"/> crypto 📄 | <input checked="" type="checkbox"/> net 📄 | <input type="checkbox"/> SPI 📄 | <input type="checkbox"/> WS2812 📄 |
| <input checked="" type="checkbox"/> DHT 📄 | <input checked="" type="checkbox"/> node 📄 | <input type="checkbox"/> SQLite 3 (large !)
📄 | <input type="checkbox"/> WS2812 effects 📄 |
| <input type="checkbox"/> DS18B20 📄 | <input type="checkbox"/> 1-Wire 📄 | <input type="checkbox"/> struct 📄 | <input type="checkbox"/> XPT2046 📄 |
| <input type="checkbox"/> encoder 📄 | <input type="checkbox"/> PCM 📄 | | |

Fig. 3-12. Listado de módulos disponibles para crear el firmware del NodeMCU desde el Cloud Build Service.

La página envía al correo que se proporcionó previamente, una liga para descargar el archivo binario (.bin) que contiene el firmware que se programa en la memoria flash.

3.3.2. ESCRIBIENDO EL FIRMWARE

Para cargar el firmware en el NodeMCU puede utilizarse la herramienta *esptool.py*, disponible en: <https://github.com/espressif/esptool>. Esta herramienta basada en Python es una plataforma independiente que funciona en OS X, Linux, Windows y cualquier otro sistema capaz de ejecutar Python [78].

En Ubuntu con Python ya instalado, primero se instala python-pip para poder descargar aplicaciones de Python desde la consola. Esto se realiza introduciendo el siguiente comando en la terminal:

```
sudo apt install python-pip
```

Terminada la instalación de pip, se puede proceder a la instalación desde la misma consola de *esptool*, la herramienta con la cual se flashea el firmware al NodeMCU. Esto se realiza con el siguiente comando:

```
pip install esptool
```

Con el NodeMCU conectado a la computadora, se introduce el siguiente comando para ver los puertos seriales en uso y así identificar el que le fue asignado a la placa de desarrollo:

```
dmesg | grep tty
```

El sistema suele asignar número de dispositivo tty en el orden en que fueron detectados por el sistema. Para identificar cuál fue asignado al NodeMCU, puede ejecutarse este comando antes y después de

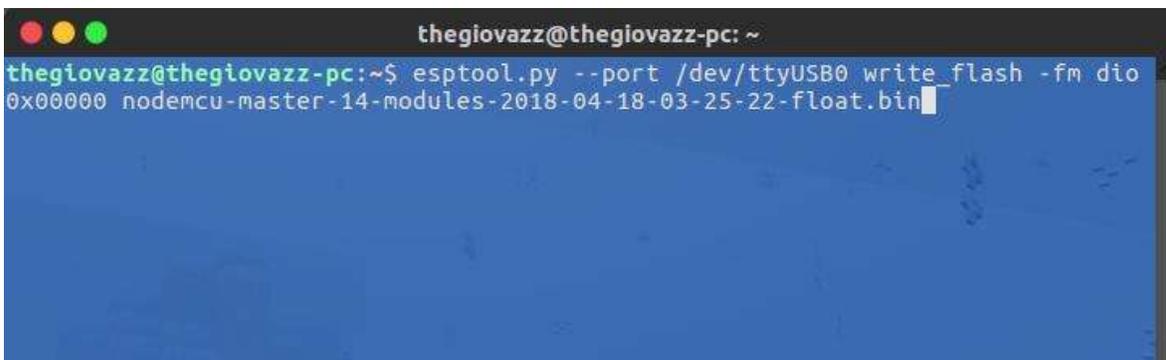
conectar el NodeMCU. De esta forma, podrá saberse cuál dispositivo tty fue le asignado, identificando cuál aparece en pantalla en la consola tras la segunda ejecución del comando, y que antes no aparecía. Si ya se tiene identificado el puerto, se puede introducir el siguiente comando para borrar todo el contenido de la memoria flash (este paso puede omitirse si se tiene la certeza de que la memoria está vacía). Hay que escribir el puerto que corresponde al NodeMCU, en lugar de *<Puerto-serial-del-ESP8266>*:

```
esptool.py--port <Puerto-serial-del-ESP8266> erase_flash
```

Luego de esto ya se puede programar el firmware con el programa `esptool.py`, utilizando el puerto serie correspondiente, el modo (que será *dio* para la mayoría de los ESP32 y ESP8266 ESP-12) y el nombre del archivo `.bin`:

```
esptool.py--port <serial-port-of-ESP8266> write_flash -fm <mode> 0x000000 <nodemcu-firmware>.bin
```

Por ejemplo, para el puerto `/dev/ttyUSB0`, modo *dio* para el ESP8266 y nombre de archivo `nodemcu-master-14-modules-2018-04-18-03-25-22-float.bin`, el comando a utilizar sería el mostrado en Fig. 3-13.

A screenshot of a terminal window on a Linux system. The window title is "thegiovazz@thegiovazz-pc: ~". The prompt is "thegiovazz@thegiovazz-pc:~\$". The command entered is "esptool.py --port /dev/ttyUSB0 write_flash -fm dio 0x000000 nodemcu-master-14-modules-2018-04-18-03-25-22-float.bin". The terminal background is blue, and the text is white. The cursor is at the end of the command line.

```
thegiovazz@thegiovazz-pc: ~
thegiovazz@thegiovazz-pc:~$ esptool.py --port /dev/ttyUSB0 write_flash -fm dio
0x000000 nodemcu-master-14-modules-2018-04-18-03-25-22-float.bin
```

Fig. 3-13. Comando en terminal de Ubuntu para cargar el firmware en el NodeMCU.

3.4. CÁMARA SERIAL

La cámara utilizada es una cámara serial de tecnología TTL: LinkSprite LS-Y201-TTL JPEG Color Camera (véase Fig. 3-14).

- **VCC:** Que es el pin de alimentación positivo, que puede recibir 3.3v o 5v de CD.
- **GND:** Donde se conecta la referencia (tierra) de la fuente de alimentación.
- **TV:** Es el pin analógico de salida.



Fig. 3-16. Pines disponibles en la cámara LinkSprite modelo LS-Y201 [82].

La cámara LinkSprite LS-Y201-TTL JPEG Color Camera se controla a través de comandos, que son enviados vía comunicación serial. En este proyecto, el microcontrolador que se encuentra dentro de un módulo ESP82636, envía los códigos a la cámara.

La cámara es capaz de reconocer 9 comandos:

- Reinicio.
- Tomar fotografía.
- Obtener tamaño de archivo JPEG con base en la configuración actual de la cámara.
- Recibir la imagen JPEG desde la cámara a través de la comunicación serial.
- Dejar de tomar fotografías.
- Modificar la relación de compresión.
- Modificar el tamaño de imagen.
 - De manera que el cambio permanezca después de reiniciar la cámara.
 - De manera que la configuración sólo dure hasta que se reinicie la cámara.
- Entrar al modo de ahorro de energía.
- Modificar la relación de compresión.

Este modelo de cámara envía una respuesta de confirmación después de la recepción de cada comando, que permite saber que el comando fue recibido con éxito y la configuración o acción solicitada ha sido efectuada.

Antes de conectar la cámara al microcontrolador, con ayuda de una computadora, se envió un comando a la cámara para fijar el tamaño de imagen. Este comando almacena la configuración aún después de reiniciar la cámara. El tamaño de imagen elegido fue el más grande permitido por la cámara: (640*480).

El proceso de pruebas con la cámara, previo a su integración en el proyecto, se encuentra en la sección 5.1.

3.5. SENSORES

Los sensores son una parte fundamental del proyecto, al ser las herramientas que permiten obtener información del sistema.

Este desarrollo cuenta con:

- Sensor de temperatura y humedad DHT11.
- Sensor de luminosidad construido con LDR.
- Sensor de Conductividad Eléctrica de líquidos.
- Sensor de pH.

Se utilizó un sensor de temperatura y humedad DHT11 debido a su bajo costo, al hecho de que mide dos variables y a que entrega los datos en formato digital, eliminando la necesidad de utilizar el convertidor analógico-digital del microcontrolador.

Se empleó un LDR para construir un pequeño circuito y realizar mediciones de la luminosidad del ambiente mediante el ADC del microcontrolador.

Se diseñó un sensor de conductividad eléctrica en líquidos, emulando el funcionamiento de uno comercial, que con ayuda de una señal de corriente alterna, hace posible obtener un voltaje que esté función de la conductividad eléctrica.

Se utilizó un sensor PH-4502C de la marca Diy More para medir el PH de la solución nutritiva y enviar los datos en forma de variaciones de voltaje al ADC del microcontrolador.

3.5.1. SENSOR DE TEMPERATURA Y HUMEDAD DHT11

En el proyecto se utiliza el sensor DHT11 (mostrado en Fig. 3-17). Este es un sensor compuesto que contiene una salida de señal digital calibrada en temperatura y humedad. Consiste en un sensor capacitivo de humedad y un termistor NTC, conectados con un microcontrolador de alto rendimiento de 8 bits. Es un sensor de bajo costo y fácil uso. Tanto para medición de humedad relativa como de temperatura, tiene una resolución de 16 bits, de los cuales, los 8 más significativos contienen la parte entera de la medición, y los menos significativos, la parte decimal [83].

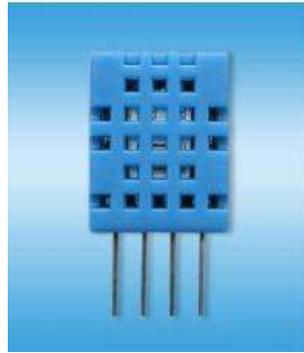


Fig. 3-17. Vista frontal del sensor de temperatura y humedad DHT11 [83].

Se alimenta con corriente directa de entre $3v$ y $5v$, teniendo un consumo de corriente de $0.3mA$. Puede medir temperatura entre $0^{\circ}C$ y $50^{\circ}C$ con una precisión de $\pm 2^{\circ}C$ y una resolución de $0.1^{\circ}C$. Puede medir humedad relativa entre $20\% RH$ y $90\% RH$, con una precisión de $4\% RH$ y una resolución de $1\% RH$. Su periodo de muestreo es de poco más de 2 segundos. Cuenta con 4 pines disponibles como se muestra en Fig. 3-18. [83].

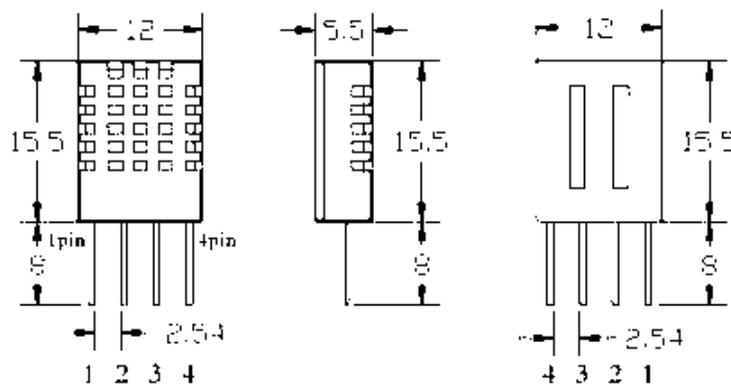


Fig. 3-18. Dimensiones en mm del DHT11 y numeración de pines [83].

Los datos son transmitidos de manera serial, por el pin 2, con el siguiente formato de datos [83]:

8 bits de información entera de humedad relativa – 8 bits de información decimal de humedad relativa – 8 bits de información entera de temperatura – 8 bits de información decimal de temperatura – 8 bits de paridad.

Donde la suma de los 4 grupos más significativos de 8 bits es igual a los 8 bits de paridad. Esto sirve para poder verificar que se hayan recibido correctamente todos los datos y no haya datos con errores. Si se recibe, por ejemplo [83]:

0011 0101 0000 0000 0001 1000 0000 0000 0100 1001

Se verifica que la transmisión de datos fue exitosa porque:

0011 0101 + 0000 0000 + 0001 1000 + 0000 0000 = 0100 1001

Y los datos de las mediciones realizadas serían:

Humedad: 0011 0101 = 35H = 53% RH

Temperatura: 0001 1000 = 18H = 24°C

3.5.2. SENSOR DE LUMINOSIDAD

Para medir la luminosidad ambiental en el entorno del sistema hidropónico, se implementó un divisor usando un sensor LDR de 10MΩ y una resistencia de 10KΩ. De esta forma, las variaciones de intensidad luminosa se pueden ver como variaciones de voltaje, que pueden ser enviadas al convertidor analógico-digital del microcontrolador. El diagrama del circuito construido se muestra en Fig. 3-19.

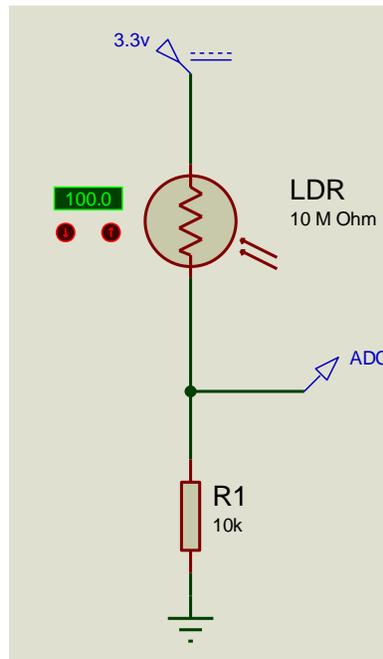


Fig. 3-19. Circuito resistivo para medir de luminosidad del sistema.

El valor resistivo del LDR se incrementa conforme se reduce la luminosidad que percibe, siendo su máximo, 10MΩ en condiciones completamente oscuras, y su mínimo, cercano a 0Ω cuando detecta la mayor cantidad de luz. El diseño del circuito hace que el ADC reciba el máximo voltaje cuando exista la mayor cantidad de luz y el mínimo ante la menor cantidad de luz.

El voltaje recibido en el ADC es función de la resistencia del LDR y determinado por (3-1). Siendo el máximo voltaje 3.3v el mínimo 3.2mV.

$$V_{ADC} = 3.3v \frac{10K\Omega}{(10K\Omega + R_{LDR})} \quad (3-1)$$

Para ajustar los valores de voltaje y relacionarlos con valores en luxes, se tomó como referencia un luxómetro analógico marca *Hioki*, modelo *LUX Tester 3421* (mostrado en Fig. 3-20). Se realizó un

muestreo para obtener una ecuación polinómica con la cual determinar la relación de voltaje del ADC y sus valores correspondientes de luminosidad en luxes. En el ADC un voltaje de $3.3v$ correspondería, por ejemplo, a una medición de 1024.



Fig. 3-20. Luxómetro Hioki LUX Tester 3421.

La gráfica de datos obtenidos mediante el muestreo y la ecuación polinómica propuesta como mejor solución, utilizando el método de mínimos cuadrados para dichos datos, se muestran en Fig. 3-21 y (3-2), respectivamente.

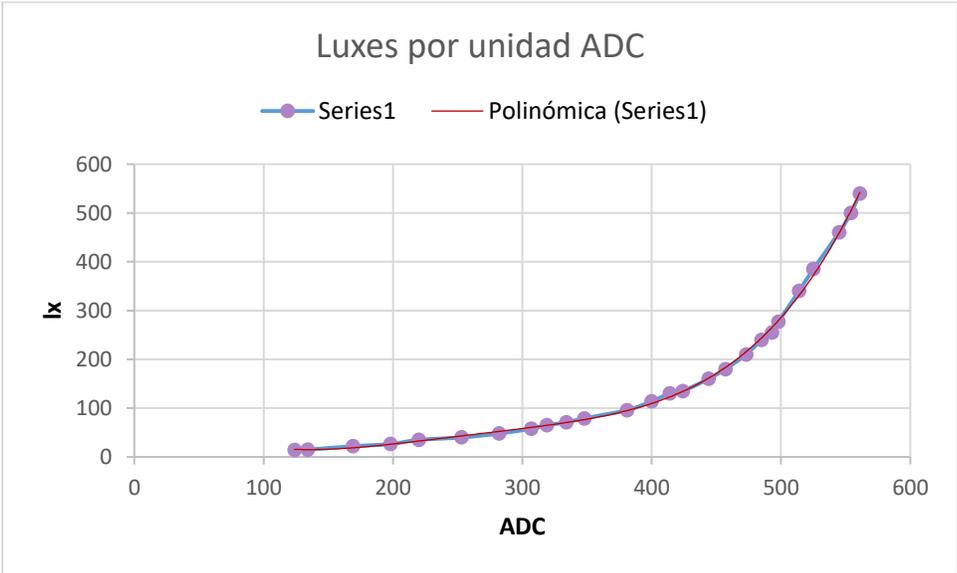


Fig. 3-21. Gráfico de luxes por unidad del ADC obtenidos con el muestreo y el luxómetro.

$$lx = 4.64901 * 10^{-8} V_{ADC}^4 - 475.934797 * 10^{-7} V_{ADC}^3 + 182.481543375 * 10^{-4} V_{ADC}^2 - 2.7864772028496 V_{ADC} + 160.332312585461 \quad (3-2)$$

3.5.3. SENSOR DE CE

Para medir la conductividad eléctrica de la solución nutritiva a emplearse en el sistema hidropónico, se diseñó e implementó un sensor. Se tomó como referencia el funcionamiento de un sensor de CE y TDS³⁴ como el de Fig. 3-22, y se utilizó el mismo para obtener la ecuación con la cual fueron calibrados los valores del sensor diseñado, obtenidos en variaciones de voltaje, a su correspondiente valor de CE.



Fig. 3-22. Medidor de TDS y CE portátil [84].

3.5.3.1. OPERACIÓN DEL SENSOR CONSTRUIDO

El microcontrolador envía una señal de 3.3v PWM de 500Hz con 50% de ciclo de trabajo a un transistor, como se describe en 3.8.2, con el fin de obtener dos señales PWM en fase complementaria. Estas señales son enviadas a dos drivers, de acuerdo a 3.8.3, para ser amplificadas a 12v.

Las señales PWM en fase complementaria de 12v, son aplicadas a un divisor resistivo elaborado con dos resistencias de 220Ω y un par de electrodos. Estos últimos son los que se sumergen en la sustancia de la cual se desea conocer su conductividad. Las resistencias de 220Ω sirven para limitar la corriente en caso de un corto circuito entre los electrodos.

La conductividad es por definición el inverso de la resistividad; a mayor conductividad de la sustancia, menor será su resistencia. El aumento o disminución de la resistencia presente en los electrodos se transforma en una variación de voltaje en los mismos, debido a la corriente que se hace circular por ellos. La corriente fluye en ambas direcciones de los electrodos a lo largo del tiempo, debido a que en *Pin1* y *Pin2* (Véase el circuito de Fig. 3-23) se aplican 0v y 12v alternadamente, provenientes de los driver. Ese

³⁴ Sólidos Totales Disueltos, por sus siglas en inglés.

cambio en la dirección de la corriente de los electrodos se hace para evitar el fenómeno de electrólisis en los mismos.

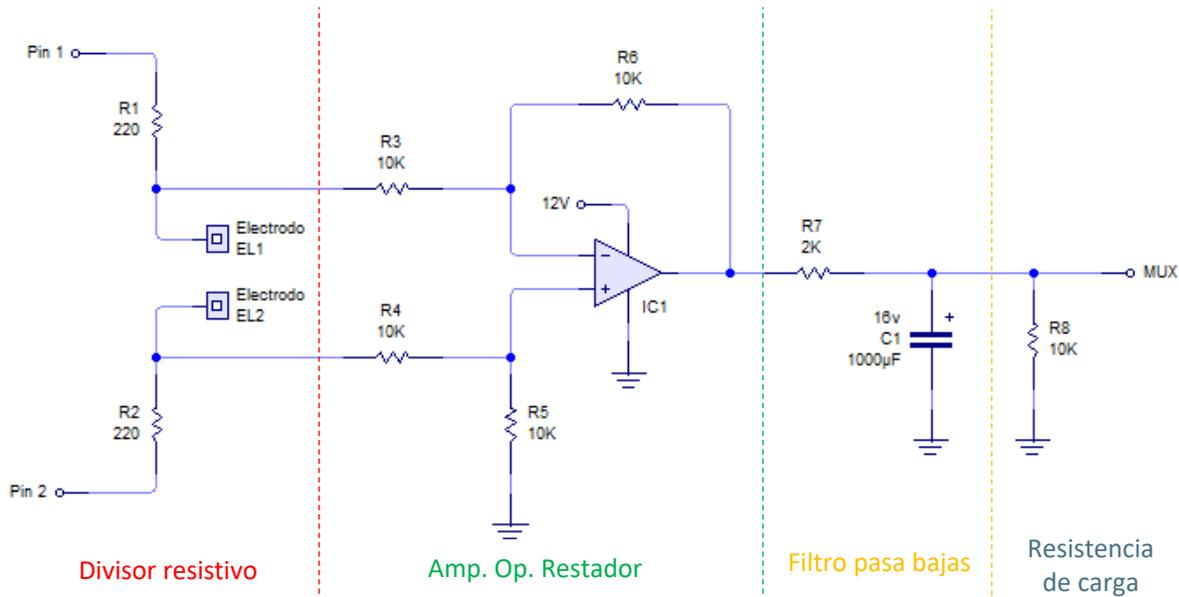


Fig. 3-23. Circuito diseñado para medir la conductividad eléctrica de la solución nutritiva.

El circuito restador de ganancia unitaria, construido con un amplificador operacional, permite obtener la diferencia del voltaje presente en los electrodos. Dicha diferencia varía en función de la conductividad de la sustancia. El circuito restador emplea, además, resistencias de un valor lo suficientemente grande en comparación con las del divisor resistivo, con el fin de evitar la interferencia con éste al tener una alta impedancia. El voltaje obtenido a la salida del restador, es una señal PWM siempre positiva con la misma frecuencia que la señal proveniente del microcontrolador, pero con una amplitud variable en función de la conductividad de la solución nutritiva.

Esta señal entra a un filtro pasa bajas, construido con una resistencia de $2k\Omega$ y un capacitor de $1000\mu F$ y $16v$. Este filtro, con una frecuencia de corte de $0.08Hz$, permite obtener a la salida una señal de corriente directa no pulsante. Finalmente se obtiene una señal de voltaje que varía en función de la conductividad de la solución nutritiva, la cual se envía a una resistencia de carga y puede medirse mediante un ADC. En el caso de este proyecto, se conecta antes a un MUX (del cual se hace mención a detalle en la sección 3.8.1) ya que el SoC utilizado sólo cuenta con un módulo ADC.

Por la naturaleza de la solución nutritiva que se emplea, se tiene la certeza de que los valores de conductividad no generan voltajes a la salida del sensor, superiores a los del rango de entrada del ADC.

Una vez construido el sensor de conductividad eléctrica, se procede a realizar un muestreo. Los resultados se comparan con el medidor de TDS y CE portátil (como el mostrado en Fig. 3-22) para relacionar las mediciones obtenidas en unidades del ADC del microcontrolador, con los valores reales en $\mu S/cm$.

La gráfica de datos obtenidos mediante el muestreo y la ecuación lineal propuesta en función de ellos, se muestran en Fig. 3-24 y (3-3), respectivamente.

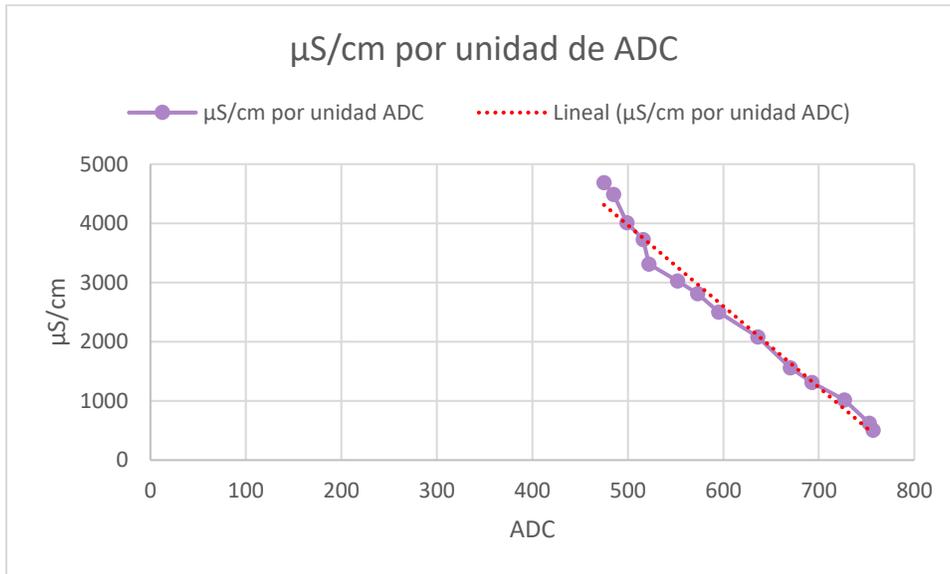


Fig. 3-24. Gráfico de μS/cm por unidad del ADC obtenidos con el muestreo y el medidor de TDS y CE empleado.

$$\frac{(\mu S)}{cm} = -13.669 ADC + 10819 \quad (3-3)$$

3.5.4. SENSOR DE PH

En el proyecto se utiliza un módulo de monitoreo y medición de pH líquido modelo PH-4502C, de la marca DIY MORE, como el mostrado en Fig. 3-25.



Fig. 3-25. Módulo de monitoreo y sensor de pH líquido modelo PH-4502C [85].

Las especificaciones técnicas del módulo son:

- Alimentación: 5v CD.
- Tamaño del módulo: 43mm x 32mm x 20mm.
- Rango de medición: 0-14 pH.
- Temperatura de medición: 0-60°C.
- Precisión: ± 0.1 pH (a 25°C).
- Tiempo de respuesta: ≤ 1 min.
- Tipo de conector BNC³⁵ para el sensor de pH.
- Cuenta con potenciómetros para el ajuste de ganancia.
- Cuenta con indicador de encendido.

La salida del sensor de pH es una señal analógica de hasta 5v que puede ser conectada a un ADC. El módulo funciona en conjunto con un electrodo sensor de pH con conector tipo BNC, como el que se muestra en Fig. 3-26.



Fig. 3-26. Electrodo sensor de PH con conector tipo BNC [86].

El electrodo debe calibrarse con una solución tampón estándar antes de cada uso continuo. Para obtener resultados más precisos, la temperatura ambiente debe ser preferiblemente de alrededor de 25°C.

Las especificaciones técnicas del electrodo son las siguientes:

- Rango de operación: 0-60°C.
- Resistencia interna: ≤ 250 M Ω .
- Tiempo de respuesta: ≤ 1 min.

³⁵ Bayoneta Neill-Concelman, por sus siglas en inglés. Es un conector de rápida conexión y desconexión utilizado para cable tipo coaxial.

El módulo permite calibrarse a través de unos potenciómetros. Si la muestra que se está midiendo es ácida, debe corregirse el electrodo con una solución tampón de pH 4.00. Si la muestra que se está midiendo es alcalina, debe usarse una solución tampón de pH 9.18 para calibrar el electrodo. La calibración se debe realizar en secciones para una mejor precisión.

Después de cada medición de pH de una solución con el electrodo de pH, éste debe limpiarse con agua (preferiblemente con agua desionizada).

Cuando el electrodo se usa por primera vez o cuando no se usa durante mucho tiempo, el bulbo del electrodo y el núcleo de arena deben sumergirse en una solución 3 mol/L de cloruro potásico (KCl 3M) durante 8 horas.

Una vez completada la medición, se debe colocar el manguito de protección del electrodo en el contenedor y se debe colocar una pequeña cantidad de solución de cloruro de potasio de 3.3 mol/L en el manguito protector para mantener húmeda la bombilla del electrodo.

El cable del electrodo debe mantenerse limpio y seco para evitar cortocircuitos en ambos extremos de la salida; de lo contrario el resultado de la medición será inexacto o no válido.

El rango dinámico de las mediciones del electrodo oscila entre valores negativos y positivos. Un 0 representaría un pH de 7.0. El circuito del módulo al que se conecta el electrodo le añade un valor de offset al valor medido por éste para así tener únicamente valores positivos de tensión a la salida del módulo. Por lo tanto, es necesaria la calibración del módulo. Se simula un pH de 7.0 desconectando la sonda del circuito y cortocircuitando la parte interna del conector BNC con la exterior. Entonces se mide con un multímetro el valor del pin Po y se ajusta el potenciómetro de calibración hasta que la salida del módulo sea 2.5V.

Dado que la relación entre la salida de voltaje y el pH medido es lineal, se utilizaron dos soluciones de calibración de *pH* 4.01 y *pH* 6.86 para realizar un muestreo y obtener la ecuación que relaciona el voltaje con los valores de pH, así como la que relaciona las mediciones en unidades del ADC con los valores de pH. La primera, sirve para determinar el rango de la señal de voltaje de salida del sensor e instrumentarla adecuadamente antes de conectarse al ADC, a través del MUX; y la segunda, sirve para relacionar los valores medidos en el ADC con los valores reales de pH.

La gráfica de datos obtenidos mediante el muestreo para el voltaje y la ecuación lineal propuesta en función de ellos, se muestran en Fig. 3-27 y (3-4), respectivamente.

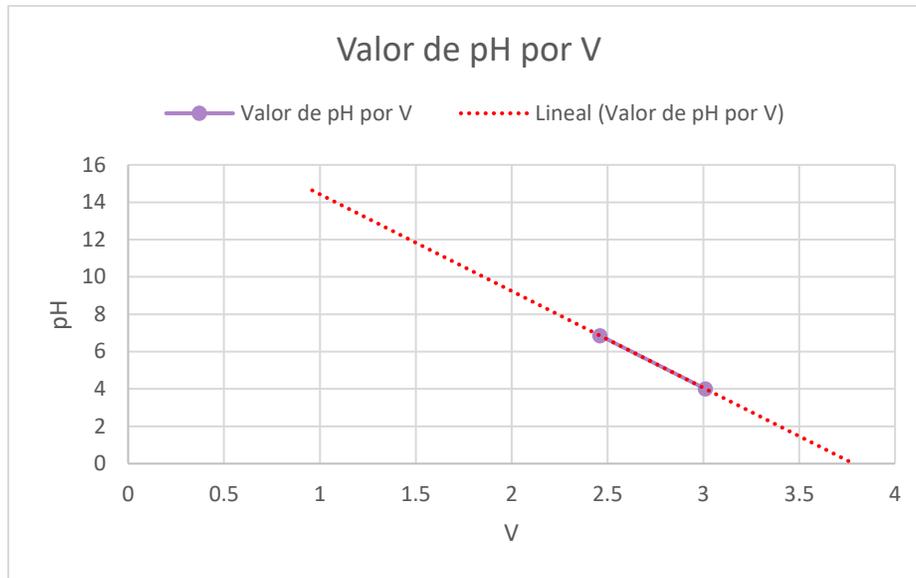


Fig. 3-27. Gráfico de valores de pH por unidad de voltaje, obtenidos durante la calibración del sensor de pH utilizado.

$$pH = -5.1818V + 19.607 \quad (3-4)$$

De acuerdo a la ecuación, la salida del sensor varía entre $1v$ y $3.7v$ para los valores de pH entre 0 y 14. No obstante, al estar desconectado el electrodo, la salida del sensor entrega $5v$; como el ADC del microcontrolador acepta voltajes máximos de $3.3v$, para evitar daños por sobrevoltajes en caso de desconexión del electrodo, se implementó un divisor resistivo que reduce el voltaje que se conecta al ADC del microcontrolador, a través del MUX, a $3.23v$ máximo en ese caso. La inclusión del divisor resistivo entre el MUX y el sensor de pH, reduce el voltaje en un 35.3%.

La ecuación que relaciona los valores de pH del sensor con las mediciones del microcontrolador en unidades del ADC y al gráfico de la recta obtenida, se pueden observar en (3-5) y Fig. 3-28, respectivamente.

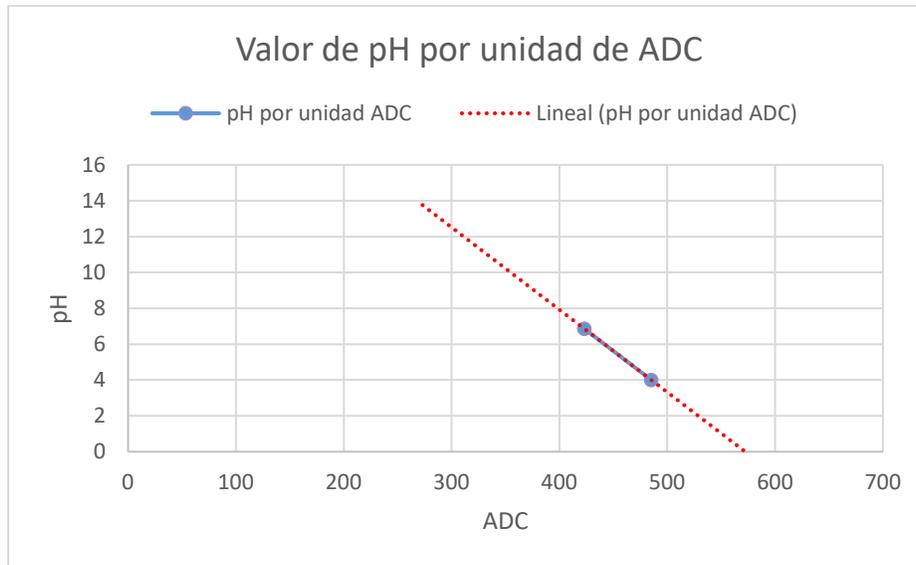


Fig. 3-28. Gráfico de relación entre las mediciones en unidades del ADC y su correspondencia con unidades pH reales.

$$pH = -0.04567 \text{ ADC} + 26.30435 \quad (3-5)$$

3.6. ACTUADORES

Los actuadores son dispositivos que permiten realizar cambios físicos a las variables del sistema; en este caso: hacer recircular el agua del circuito hidropónico y modificar la cantidad y tipo de iluminación que requieren las plantas para su correcto desarrollo.

Para este proyecto se utilizó una bomba de agua sumergible miniatura para hacer circular la solución nutritiva y una tira de LED RGB para regular la intensidad y longitud de onda deseada para los cultivos en cuestión.

3.6.1. BOMBA DE AGUA

La bomba de agua utilizada fue el modelo *zjchao-6935824c* de la marca *Yosoo*, mostrada en Fig. 3-29. Se trata de una bomba con una temperatura de funcionamiento de hasta 100°C , de bajo consumo y poco ruido (30 dB en 1 metro). Se alimenta con 12V de CD y tiene una corriente nominal de 500mA [87].



Fig. 3-29. Bomba Yosoo zjchao-6935824c de 12v de CD [87].

Tiene una tasa de flujo máximo de 6.5 *litros/minuto* y un sistema de presión máxima de 10*Bar*, que le permiten lanzar agua en vertical hasta 3 metros por encima de ella [87].

Para poder controlar el encendido y apagado con una señal digital proveniente del microcontrolador, se utilizaron 2 drivers L293B. De esta forma, se le proporciona el voltaje de alimentación y la corriente necesarios.

3.6.2. TIRA DE LEDs

La tira de LEDs utilizada en el proyecto es el modelo *TMU5SOLA* de la marca *laser&led* (mostrado en Fig. 3-30). La cual es una tira de LEDs RGB de intensidad regulable, que tiene una potencia de 27*W*, un consumo de 2.25*A*, se alimenta con 12*v* de CD, tiene un ángulo de haz de iluminación de 120° cuenta con un factor de protección contra salpicaduras *IP 65*, mide 5*m* de largo y 1*cm* de ancho [88].



Fig. 3-30. Tira de LED de 5m modelo *TMU5SOLA* de la marca *laser&led* [88].

Originalmente, la tira incluye un control remoto con el cual puede seleccionarse el color, la intensidad de la iluminación, apagar y encender la tira de LED. Este funciona en conjunto con la caja de recepción

infrarroja (que se muestra en Fig. 3-31). Esta misma caja cuenta con una conexión a 12v y una salida de 4 pines que se conecta directamente con la entrada de alimentación que tiene la tira de LED, no obstante, no se utilizó en el proyecto: sirvió como referencia para deducir que de los 4 pines que tiene la tira de LED, uno es de alimentación a 12v, y el resto, sirve para controlar la intensidad de cada color, permitiendo obtener cualquier color de la gama disponible. Cada uno de los 3 canales restantes controla el encendido de uno de los 3 colores, utilizando una señal de PWM³⁶.



Fig. 3-31. Caja receptora de IR y control remoto marca laser&led.

Enviando una señal en bajo (0v) por un canal, se logra que todos los LED del color correspondiente enciendan. Regulando la señal enviada con un ciclo de trabajo variable, se logra que la intensidad de ese color percibida por el ojo humano, varíe en función al ciclo de trabajo. Un ejemplo de la señal que hace que uno de los 3 colores de LED encendiera al 25% de su intensidad, se muestra en Fig. 3-32.

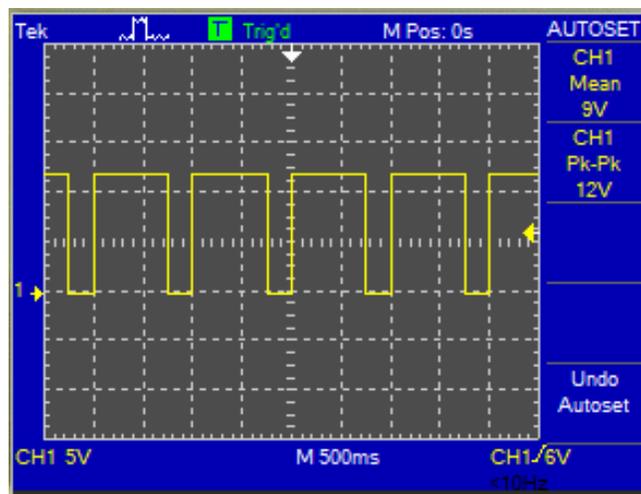


Fig. 3-32. Ejemplo de señal PWM con 75% de ciclo de trabajo.

³⁶ Modulación por Ancho de Pulso, por sus siglas en inglés.

Para que el microcontrolador sea capaz de regular el encendido y apagado de los LED con sus señales PWM de 3.3v, se utiliza un driver L293B (mencionado en la sección 3.8.3) por cada uno de los canales de la tira de LED, para proporcionarles una señal PWM amplificada a 12v y la corriente necesaria.

3.7. SERVIDOR WEB IMPLEMENTADO EN UN SBC

Se empleó un SBC modelo Raspberry Pi 3B (como la mostrada en Fig. 2-7) como hardware para montar el servidor web en él. Se trata de uno de los modelos de la tercera generación de Raspberry Pi. Cuenta con las siguientes especificaciones técnicas [89]:

- Procesador de cuatro núcleos a 1.2GHz de velocidad Broadcom BCM2837 de 64 bits.
- 1Gb de memoria RAM.
- Chip de conectividad inalámbrica LAN y Bluetooth de bajo consumo de energía (BLE) integrado.
- Estandar Ethernet Base 100.
- 40 pines GPIO.
- 4 puertos USB, separados en 2 unidades.
- Salida de audio estéreo de 4 polos y puerto compuesto de video.
- HDMI³⁷ *full size*.
- Puerto de cámara CSI³⁸ para conectar una cámara Raspberry Pi.
- Puerto DSI³⁹ para conectar una pantalla táctil Raspberry Pi.
- Puerto micro SD para cargar el sistema operativo y almacenar datos.
- Fuente de alimentación conmutada Micro USB actualizada, de hasta 2.5a.

La Raspberry Pi modelo 3B permanecerá en producción hasta al menos enero del 2022 [89].

3.8. OTROS CIRCUITOS INTEGRADOS (CI)

Además de los componentes electrónicos utilizados para los sensores o actuadores mencionados anteriormente, se utilizaron otros circuitos integrados en el proyecto. Cada uno de ellos cumple una función específica.

3.8.1. MULTIPLEXOR/DEMÚLTIPLEXOR

Se utilizó un multiplexor de matrícula CD4051BCN (de aquí en adelante, MUX), el cual es un multiplexor/demultiplexor de 8 canales, para realizar mediciones analógicas con el NodeMCU en más de una fuente, pero contando sólo con un módulo ADC. Cuenta con un pin de inhabilitación, tres pines de selección 8 canales de entrada/salida y uno de salida/entrada. Su diagrama de conexión se muestra en Fig. 3-33.

³⁷ Interfaz Multimedia de Alta Definición, por sus siglas en inglés.

³⁸ Interfaz Serie para Cámaras, por sus siglas en inglés.

³⁹ Interfaz Serie para Pantallas, por sus siglas en inglés.

Sus principales características son [90]:

- Puede manejar señales digitales de 3 a 15v, y analógicas de 15v_{pp}.
- Baja resistencia en estado encendido (80Ω).
- Alta resistencia en estado de apagado.
- Muy baja disipación de potencia de reposo en todas las condiciones de entrada y suministro de control digital (1μW).
- Decodificación de direcciones binarias en chip.

Su tabla de verdad es la mostrada en

Tabla 3-1.

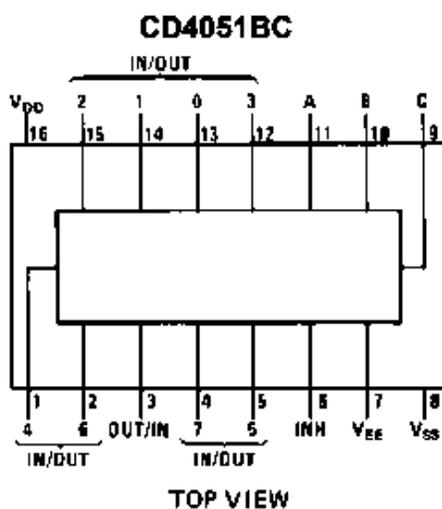


Fig. 3-33. Diagrama de conexión del CI CD4051BCN [90].

Tabla 3-1. Tabla de verdad del CI CD4051BCN [90].

Estados de las entradas				Canal habilitado
Inhabilitado	C	B	A	CD4051BCN
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	*	*	*	Ninguno

En este proyecto se utilizan sólo los primeros 4 canales del MUX, por lo que la entrada C se tiene conectada a un 0 lógico constantemente. Las entradas A y B, son controladas por el microcontrolador a través de salidas digitales, mientras que la salida del canal habilitado del MUX, se envía al ADC del microcontrolador. Esto permite utilizar un solo ADC para leer múltiples señales provenientes de sensores, seleccionando una a la vez, según se estén realizando las mediciones.

La salida del MUX se conecta en serie a una resistencia de 100Ω , la cual funge como resistencia de *pull-up*, es decir, asegura que el microcontrolador reciba un 0 lógico cuando corresponde y evita valores flotantes o inciertos; y en paralelo a un diodo zener de $3.3v$ y a una resistencia de $10K\Omega$. El diodo zener en paralelo, que se polariza inversamente, limita el voltaje máximo que puede llegar al microcontrolador a $3.3v$, evitando así que se dañe el ADC; mientras que la resistencia en paralelo con el diodo sirve como protección para este, en caso de sobrevoltajes. El circuito descrito en éste párrafo se puede encontrar en Fig. 3-34.

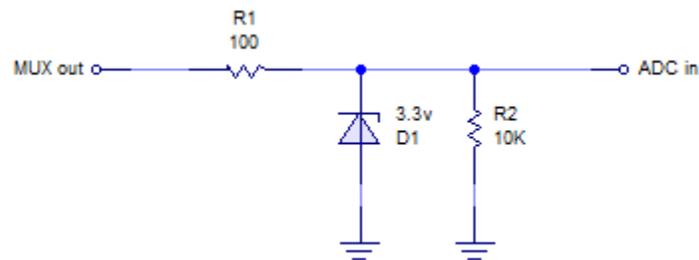


Fig. 3-34. Circuito de instrumentación entre la salida del MUX y la entrada del ADC.

3.8.2. TRANSISTOR BC548B

Para obtener una señal lógica invertida a la proveniente de uno de los GPIO del microcontrolador, se requirió un transistor BC548B. Este permite tener dos señales lógicas invertidas (véase Fig. 3-35) que son utilizadas como señales de control para el sensor de CE, mencionado en la sección 3.5.3, y utilizar un solo GPIO del microcontrolador para ello. Su funcionamiento es análogo al de una compuerta lógica inversora.

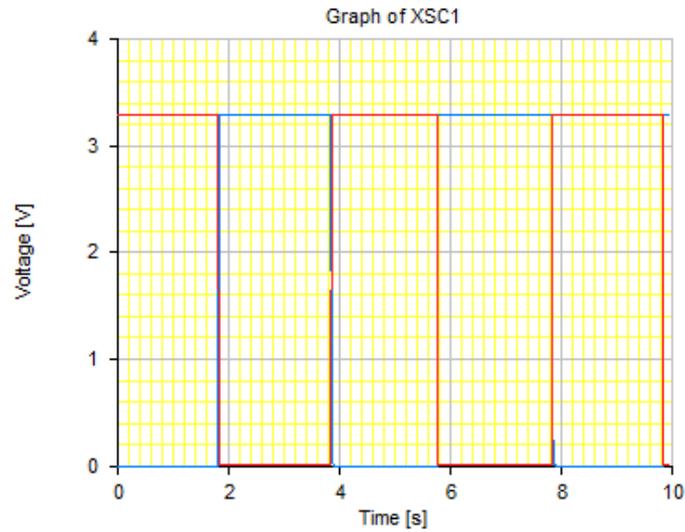


Fig. 3-35. Gráfica de señales lógicas invertidas de 3.3v, en colores rojo (señal original) y azul (señal invertida con transistor).

Se trata de un transistor NPN cuyas principales características son las siguientes [91]:

- Voltaje máximo de colector-emisor de $30V_{CD}$.
- Voltaje máximo de colector-base de $30V_{CD}$.
- Voltaje máximo de base-emisor de $6V_{CD}$.
- Corriente de colector de corriente continua $100mA$.
- Temperatura de funcionamiento entre -55 y $150^{\circ}C$.

El diagrama de conexión del transistor se muestra en Fig. 3-36.

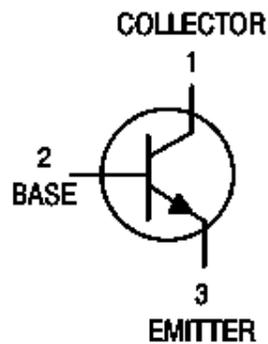


Fig. 3-36. Diagrama de terminales del transistor BC548B [91].

El circuito empleado para este propósito utilizando este transistor, se muestra en Fig. 3-37. Las resistencias se eligieron con los valores de $1K\Omega$ para limitar la corriente de colector y $100K\Omega$ para la corriente de base, esto con el propósito de garantizar la conmutación del transistor.

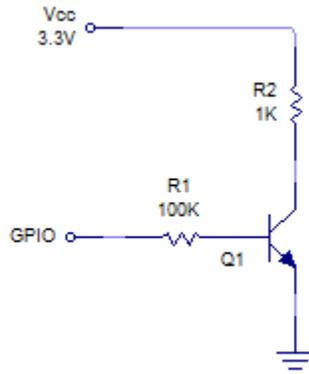


Fig. 3-37. Circuito de transistor utilizado para obtener una señal lógica invertida.

3.8.3. DRIVER L293B

Es un circuito integrado que contiene 4 drivers push-pull. Sus características principales son [92]:

- Cada canal soporta hasta 1A de corriente.
- Soporta una corriente de pico no repetitiva de hasta 2A por canal.
- Alta inmunidad al ruido.
- Cuenta con protección para temperaturas elevadas.
- Alimentación independiente para la lógica de control.
- Cuenta con dos pines de habilitación para 2 grupos de 2 drivers.
- Voltaje de alimentación de hasta 36v de CD.
- Voltaje de alimentación lógico de entre 4.5v y 36v de CD.

El diagrama de bloques de su estructura, en un ejemplo de aplicación con motores, se muestra en Fig. 3-38.

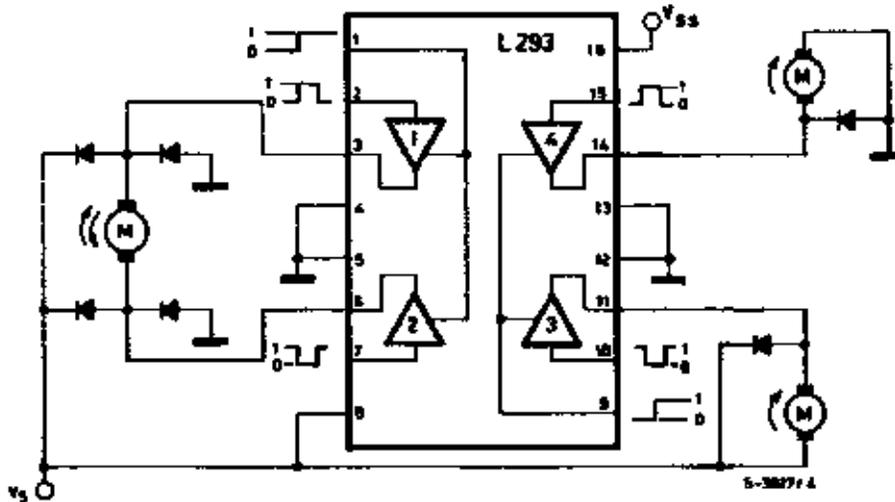


Fig. 3-38. Diagrama de bloques del L293B en un ejemplo de aplicación con motores [92].

Cada circuito integrado L293B contiene 4 drivers en su interior. Si se juntan 2 drivers, puede crearse un puente H para controlar el encendido bidireccional de motores, teniendo así hasta 2 puentes H por cada CI, con un pin de habilitación para cada puente. En el prototipo se utilizan dos de estos circuitos integrados: se emplean 2 drivers para el sensor de CE (del que se habla en la sección 3.5.3), 2 más para conectarlos en paralelo y alimentar de la bomba de agua (que se describe en la sección 3.6.1) y otros 3 para alimentar los 3 canales de la tira de LED.

El uso individual de cada uno de los drivers permite amplificar una señal lógica a un voltaje mayor. En este caso, la señal lógica es la proveniente del microcontrolador de 3.3v y es transformada a una de 12v.

CAPÍTULO 4. DISEÑO Y DESCRIPCIÓN DEL SOFTWARE

4.1. INTRODUCCIÓN

En este capítulo se aborda todo lo relacionado con los elementos intangibles del proyecto, como son los programas de los microcontroladores, las secuencias de control de los periféricos, los servicios de NodeJS y sus componentes, la programación de la página web, el diseño de operación de la base de datos y los protocolos de comunicación entre los elementos.

Del lado del software, también es el servidor web montado en la Raspberry Pi 3B el eje central del proyecto. Él es el encargado de recibir información de los dos microcontroladores, de enviarle órdenes de captura de nueva fotografía a uno, de solicitar la actualización del estado de los actuadores al otro, de recibir las peticiones del usuario, a través del navegador web, de ejecutar el bróker MQTT que intercomunica a los elementos del proyecto y de almacenar información en la base de datos. Un esquema general de la relación entre los diferentes elementos del software se muestra en Fig. 4-1.

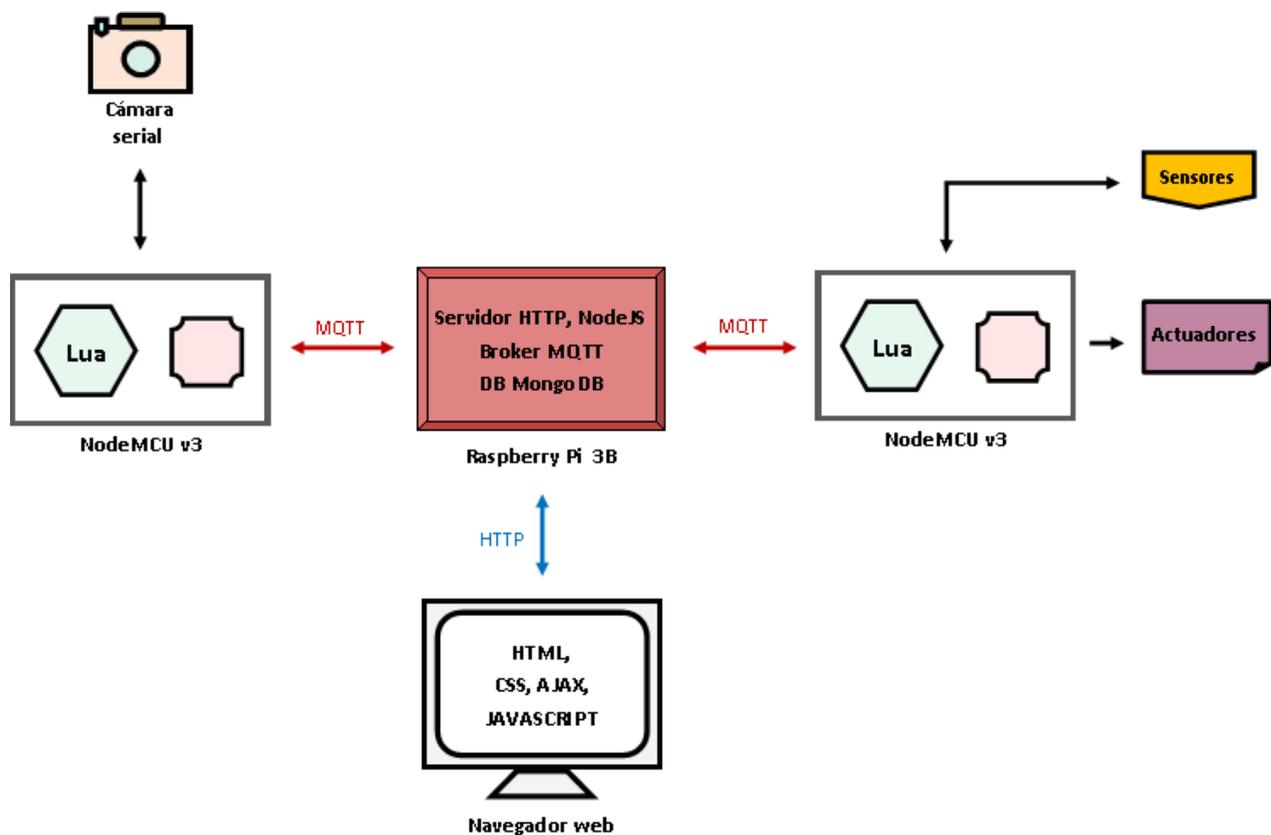


Fig. 4-1. Esquema general de la relación entre los elementos del software.

4.2. FLUJO DE DATOS

En esta sección se describe y ejemplifica con diagramas, el flujo de datos entre los distintos elementos que conforman el proyecto. En todos los diagramas, el transcurso del tiempo se representa descendiendo verticalmente en los diagramas. Cada una de las líneas gruesas verticales representa un elemento en el intercambio de información. Las flechas que van de una línea gruesa a otra, representan el intercambio de información y su dirección. El color de las flechas representa el protocolo o medio usado para transmitir la información. Dicha información relativa al color se encuentra en Fig. 4-2.

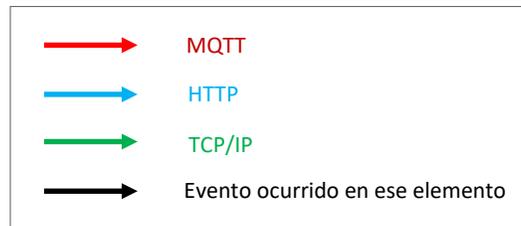


Fig. 4-2. Acotaciones de los diagramas de flujo de datos.

Se omite el bróker MQTT como intermediario en los diagramas, y se ilustra sólo el elemento de origen del mensaje y los elementos receptores finales.

4.2.1. AUTENTICACIÓN

La autenticación es el proceso que impide que personas no autorizadas hagan uso de la plataforma en línea. Si se intenta acceder a cualquier página o recurso del sitio web sin estar autenticado, el servidor web redirigirá a la página de inicio de sesión para que se introduzcan las credenciales. En caso de un inicio de sesión fructuoso (mostrado en Fig. 4-3), se autenticará la sesión del navegador actual, y permitirá ver la información, realizar peticiones y modificaciones a través de la página web, hasta que la sesión sea destruida, solicitando un cierre de sesión (como se muestra en Fig. 4-4).

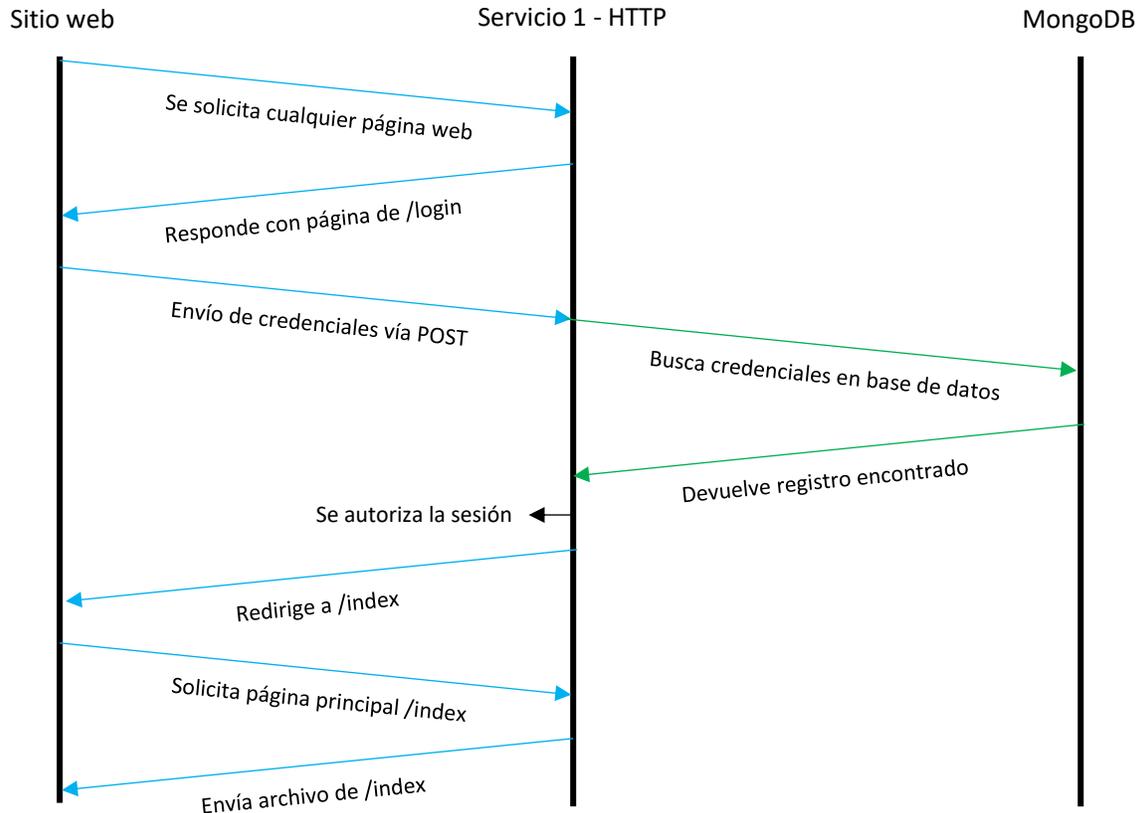


Fig. 4-3. Diagrama de flujo de datos de una autenticación fructuosa.

Si el inicio de sesión es infructuoso, se notifica al usuario que los datos introducidos no son correctos o no se encontraron registrados en la base de datos a través de la misma página web, dando como opción volver a intentarlo.

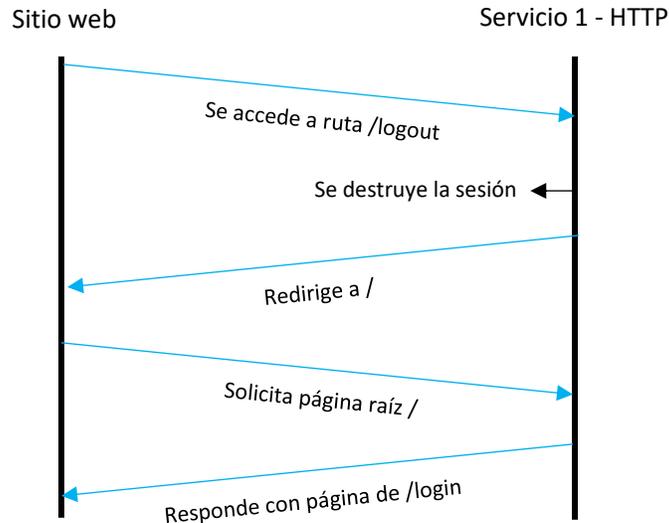


Fig. 4-4. Diagrama de flujo de datos de un cierre de sesión.

4.2.2. CAPTURA DE NUEVA FOTOGRAFÍA

En el proceso de captura de una nueva fotografía están involucrados 5 elementos dentro del intercambio de información (la cámara web se considera como un elemento periférico y no independiente, por tanto, se deja fuera del diagrama de flujo).

Todo inicia cuando el usuario solicita una nueva fotografía a través del sitio web. Entonces, el servicio 1 recibe la solicitud e indica vía MQTT al microcontrolador 2 que tome una nueva imagen. Este la obtiene de la cámara serial y la almacena en su memoria, para posteriormente comenzar a enviar por partes en paquetes MQTT, después de indicarle a los dos servicios y a sí mismo que la transmisión de paquetes inicia. Estos paquetes son recibidos por los dos servicios; el primero de ellos, calcula el porcentaje de paquetes recibidos para poder enviar esa información a la página web y mostrarle en pantalla al usuario, el estado de la carga de la nueva fotografía periódicamente; el segundo, es el responsable de almacenar los datos recibidos en cada paquete y solicitar al microcontrolador 2 que continúe con el envío del siguiente, una vez que haya finalizado con la tarea de almacenar los datos parciales.

El envío de paquetes continúa hasta que se han enviado todos los datos de la fotografía. En ese momento, el microcontrolador envía un mensaje MQTT que indica que la transmisión de paquetes ha finalizado exitosamente. En ese momento, el servicio 1 cambia su estado de recepción a completado y el servicio 2 reconstruye la fotografía a partir de los datos binarios recibidos. Posteriormente, el servicio 2 almacena la fotografía en la base de datos, empleando el cliente de mongoDB.

Cuando el sitio web es notificado de que todos los paquetes han sido recibidos con éxito, entonces solicita la última imagen almacenada en la base de datos al servicio 1. Este realiza la consulta en la base de datos y envía la foto, una vez la obtiene.

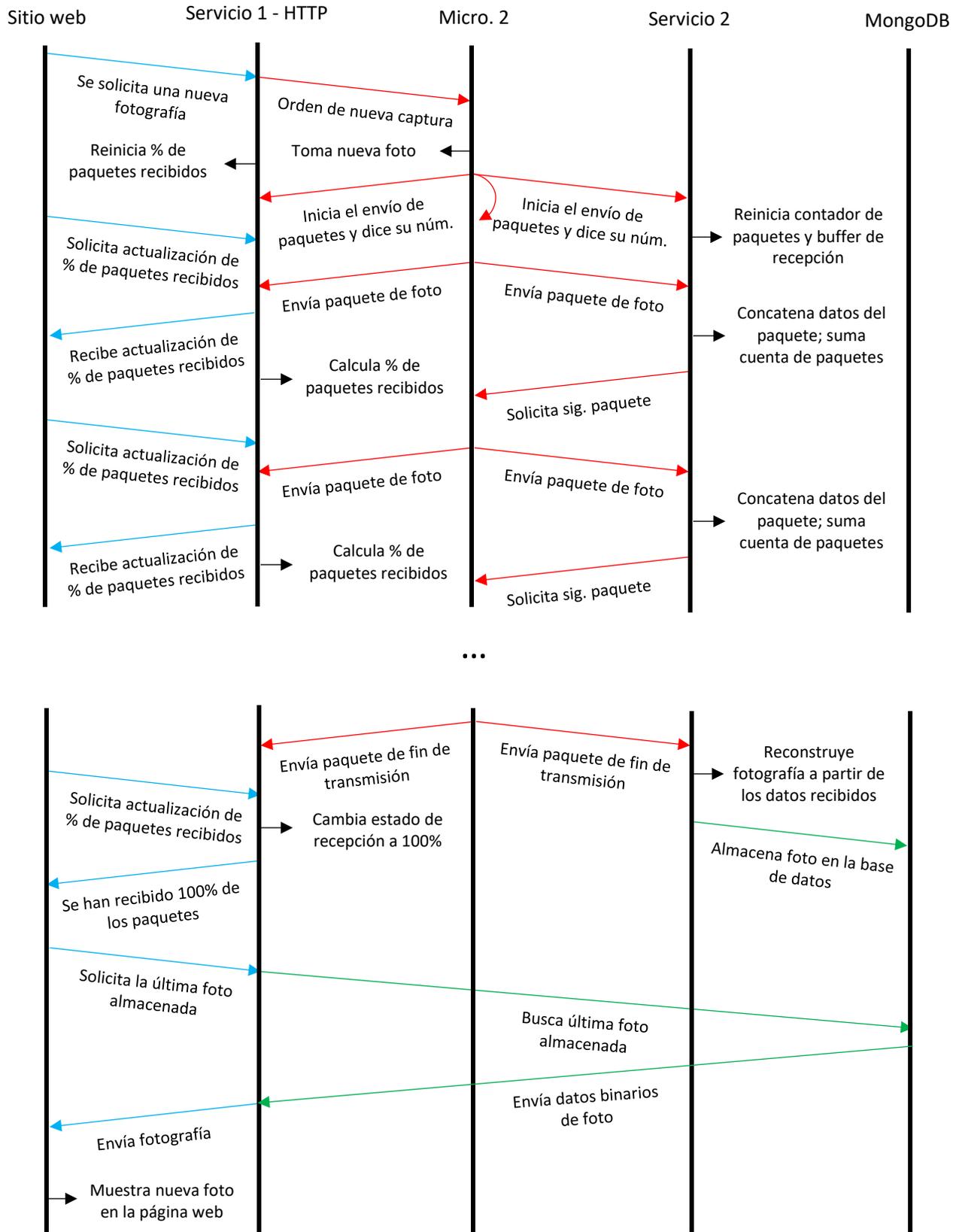


Fig. 4-5. Diagrama de flujo de datos del proceso de captura de una nueva fotografía.

Finalmente, la nueva fotografía es mostrada en la página web, tras haber sido recibida del servicio 1.

Todo el proceso de captura de nueva fotografía, se encuentra ilustrado en el diagrama de flujo de datos de Fig. 4-5.

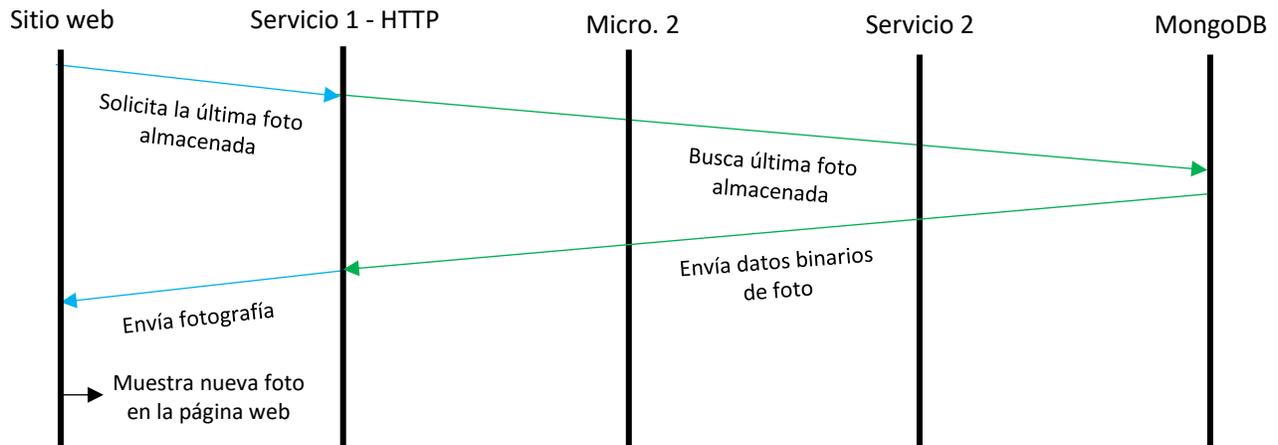


Fig. 4-6. Diagrama de flujo de datos del proceso de solicitud de la última fotografía almacenada en la base de datos.

Cuando en el sitio web se accede a la página principal, se entra por defecto en la vista *Imagen*. Cada que se entra en ese apartado, la página solicita al servicio 1 la última fotografía almacenada en la base de datos, para mostrarla en la página. El flujo de datos del proceso de solicitud de la última fotografía se muestra en Fig. 4-6.

4.2.3. RECEPCIÓN Y ALMACENAMIENTO DE DATOS DE LOS SENSORES

El microcontrolador 1 de manera periódica obtiene nuevos datos provenientes de las mediciones de los sensores con los que se comunica. Cada que obtiene un nuevo dato, lo envía vía mensaje MQTT al servicio 2, el cual adjunta la fecha y el tiempo en que fue recibido el mensaje, y posteriormente lo almacena en la base de datos como un documento. De esta forma, cada medición queda almacenada con su fecha y hora, y el tipo de variable de donde proviene el dato, en un solo documento.

Todo el proceso del flujo de datos desde que se obtiene un nuevo dato hasta que es almacenado en la base de datos, se ilustra en Fig. 4-7.

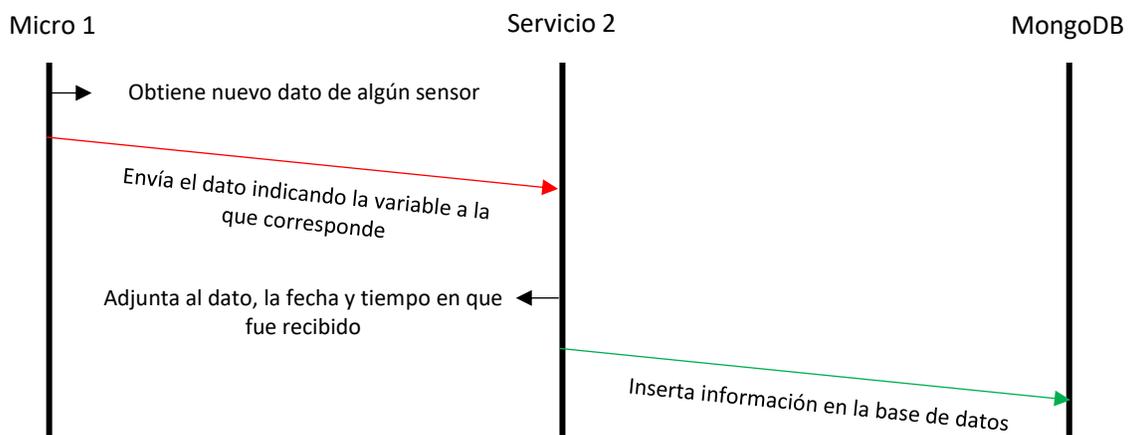


Fig. 4-7. Diagrama de flujo de datos del proceso de recepción e inserción de datos de los sensores en la base de datos.

4.2.4. GRAFICACIÓN DE DATOS DE LOS SENSORES

Los datos de las variables que se monitorean con el proyecto, son mostrados al usuario de la página web a través de gráficas. Cuando el usuario selecciona la vista *Gráficos* en la página web, el sitio web solicita al servicio 1 los datos de la variable en cuestión que son graficados. El servicio 1 recibe esta solicitud y entonces se comunica con la base de datos para buscar en los registros, la información de la variable deseada y la cantidad de datos que desea obtenerse. Posteriormente, la base de datos devuelve la información solicitada al servicio 1, y éste la envía al sitio web, quien es el encargado de actualizar la gráfica con los datos recibidos.

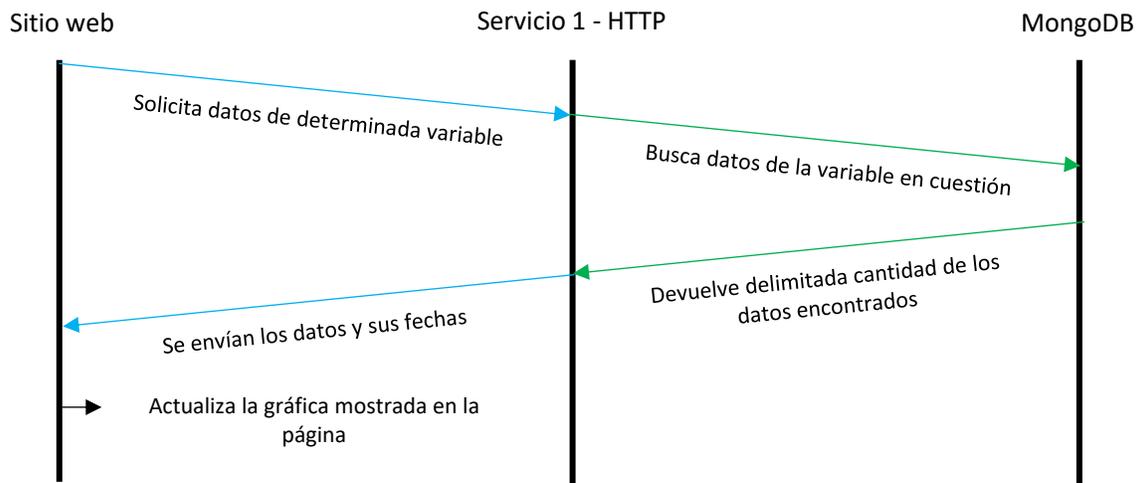


Fig. 4-8. Diagrama de flujo de datos del proceso de graficación de datos de los sensores.

Cuando se accede al modo de graficación, el sitio web está solicitando periódicamente nuevos datos para mantener actualizada la gráfica, repitiendo entonces este proceso, que se ilustra en Fig. 4-8.

La variable que se grafica, puede ser seleccionada a través de los íconos que se encuentran debajo de las gráficas en la página web. De esto se habla más detalladamente en la sección 4.5.2.

4.2.5. OBTENCIÓN DEL ESTADO DE LOS ACTUADORES

Cuando el microcontrolador 1 se reinicia, comienza un proceso de comunicación con el servicio 2 mediante el cual, obtiene el último estado configurado para cada actuador. De esta forma se garantiza que los actuadores se mantendrán en el estado deseado por el usuario.

En el periodo que existe entre el reinicio y la obtención del último estado almacenado en la base de datos, el microcontrolador 1 coloca los actuadores en un estado predeterminado, que sirve a la vez como indicador para el operador del sistema, de que se está reiniciando el sistema y que el último estado de los actuadores no ha sido recibido.

El proceso de obtención del último estado registrado en la base de datos de un actuador, se ejecuta de manera individual para cada uno de ellos. El proceso mostrado en Fig. 4-9 se ejecuta dos veces de manera consecutiva tras cada reinicio (Una por cada actuador).

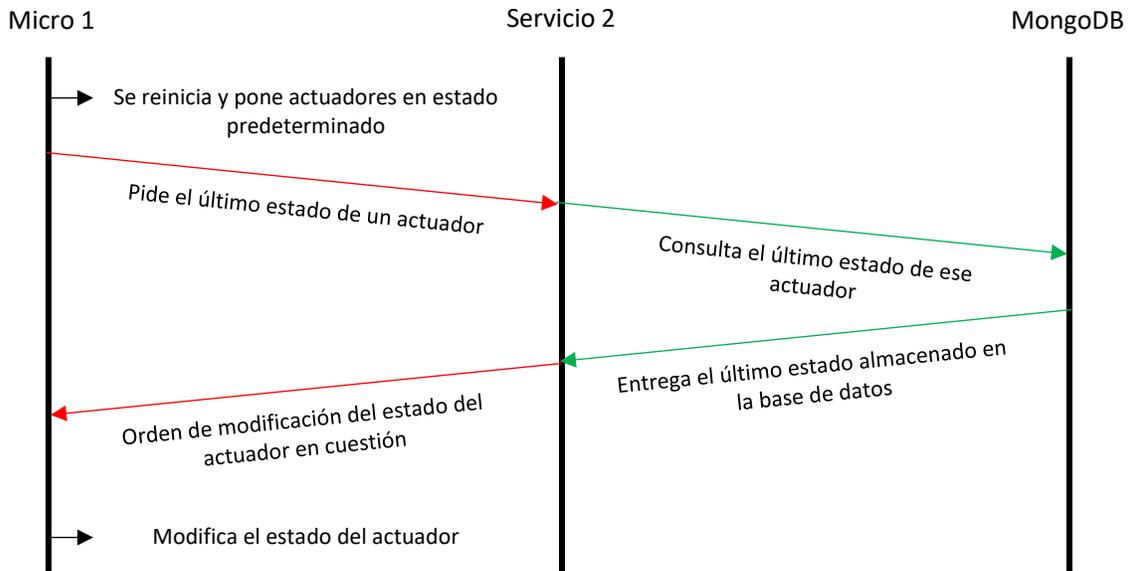


Fig. 4-9. Diagrama de flujo de datos del proceso de obtención del último estado almacenado en la base de datos para un actuador, por solicitud del microcontrolador 1.

De manera análoga, el sitio web realiza una petición al servicio 1 para conocer el estado actual de los actuadores, y presentarlo en la página web correctamente. Este proceso (ver Fig. 4-10) se ejecuta cada que la vista *Actuadores* de la página web es seleccionada haciendo clic en el botón de la barra lateral de navegación correspondiente.

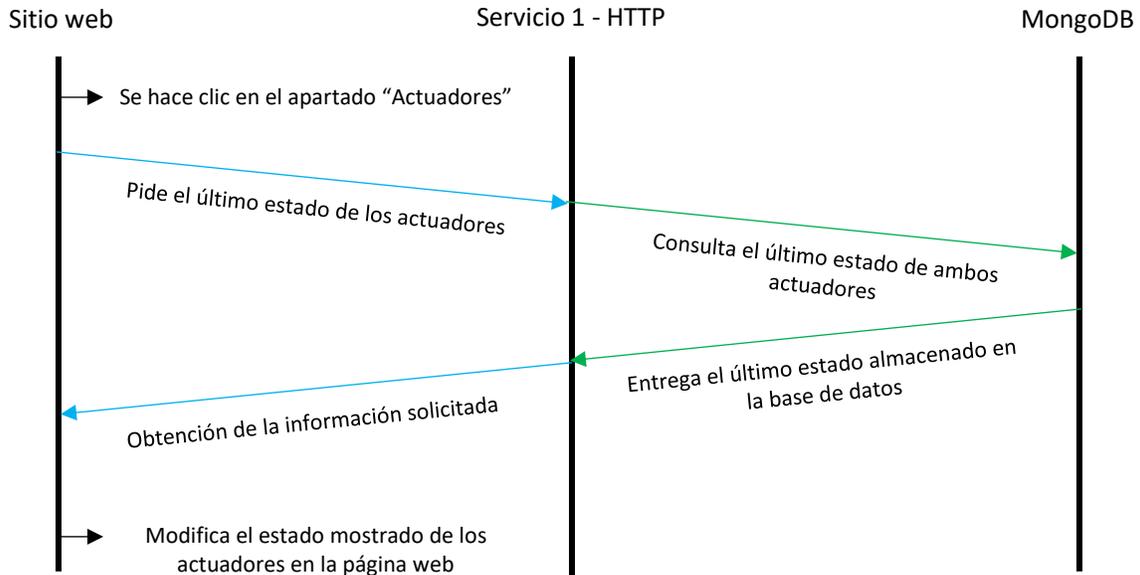


Fig. 4-10. Diagrama de flujo de datos del proceso de obtención del último estado almacenado en la base de datos de los actuadores, por solicitud de la página web.

A diferencia del proceso mostrado en Fig. 4-9, el de Fig. 4-10 se ejecuta una sola vez, puesto que se envía la información de todos los actuadores en la misma solicitud, en lugar de ejecutar el proceso una vez por actuador.

4.2.6. CAMBIO EN EL ESTADO DE UN ACTUADOR

Cuando el usuario de la página web solicita a través de ella un cambio en el estado de uno de los actuadores, se envía una petición HTTP al servicio 1, el cual, envía un mensaje MQTT, a través del bróker, con el comando de cambio para el microcontrolador 1 y al mismo tiempo, almacena ese nuevo estado en la base de datos. El microcontrolador entonces realiza el cambio solicitado al actuador seleccionado.

La utilidad de guardar el estado para los actuadores en la base de datos radica en que, de esa forma, se puede consultar para sincronizar el estado en caso de reinicio del microcontrolador 1 y para mostrar el estado actual en la página web cada vez que se solicite.

El flujo de datos del proceso que se lleva a cabo cuando se solicita un cambio en el estado de un actuador, puede verse en Fig. 4-11.

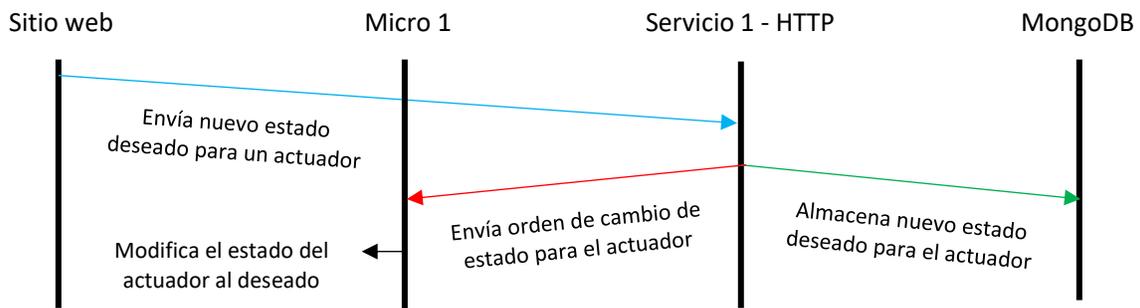


Fig. 4-11. Diagrama de flujo de datos del proceso de modificación del estado de un actuador.

4.3. SERVIDOR WEB

Como se mencionó en la sección 3.7, se utilizó una Raspberry Pi 3B como servidor web. En ella se encuentran operando dos servicios NodeJS programados en Javascript, un bróker MQTT y una base de datos MongoDB. Uno de los servicios es responsable de operar como servidor HTTP.

El sistema operativo instalado en la Raspberry Pi es Raspbian.

4.3.1. NODEJS

Concebido como un entorno de ejecución de JavaScript orientado a eventos asíncronos, Node está diseñado para construir aplicaciones en red escalables. En él, se pueden manejar conexiones concurrentes. Por cada conexión el callback es ejecutado, sin embargo, si no hay trabajo que hacer Node está durmiendo, esto contrasta con el modelo de concurrencia más común hoy en día, donde se usan hilos del Sistema Operativo. Las operaciones de redes basadas en hilos son relativamente ineficientes y son muy difíciles de usar. Además, los usuarios de Node están libres de preocupaciones sobre el bloqueo del proceso, ya que no existe. Casi ninguna función en Node realiza I/O directamente, así que el proceso nunca se bloquea. Debido a que no hay bloqueo es muy razonable desarrollar sistemas escalables en Node [93].

El logotipo de Node JS se muestra en Fig. 4-12.

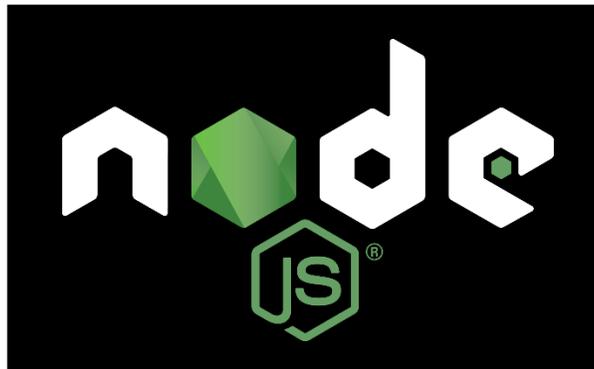


Fig. 4-12. Logotipo de NodeJS [93].

Node está influenciado por sistemas como Event Machine de Ruby ó Twisted de Python, sin embargo, lleva el modelo de eventos un poco más allá: presenta un bucle de eventos como un entorno en vez de una biblioteca. En otros sistemas siempre existe una llamada que bloquea para iniciar el bucle de eventos. El comportamiento es típicamente definido a través de callbacks al inicio del script y al final se inicia el servidor mediante una llamada de bloqueo. En Node no existe esta llamada. Node simplemente ingresa el bucle de eventos después de ejecutar el script de entrada. Node sale del bucle de eventos cuando no hay más callbacks que ejecutar. Se comporta de una forma similar a JavaScript en el navegador: el bucle de eventos está oculto al usuario [93].

4.3.1.1. NPM

JavaScript ha pasado de ser un lenguaje utilizado para brindar interactividad a las páginas web, a ser utilizado ahora también en el lado de los servidores gracias a Node.js. Dentro de Node, el código se organiza por módulos o paquetes, así que al momento de trabajar con él va a ser necesario agregar más módulos, es aquí donde entra npm [94].



Fig. 4-13. Logotipo de npm [95].

Node Package Manager o simplemente npm (cuyo logotipo se muestra en Fig. 4-13) es un gestor de paquetes que hace más fácil la programación al momento de trabajar con Node, ya que gracias a él, se puede tener cualquier biblioteca disponible con sólo una línea de código. Npm ayuda a administrar módulos, distribuir paquetes y agregar dependencias de una manera sencilla [94].

Cuando se instalan nuevos paquetes lo que hace npm es instalarlo de manera local en el proyecto dentro de la carpeta *node_modules*, pero se le puede indicar que lo instale de manera global de ser necesario [94].

Además, existen módulos que ya vienen instalados por defecto en Node.js. A estos módulos se les conoce como “módulos nativos” [94].

4.3.2. MONGODB

MongoDB es una base de datos de documentos que ofrece una gran escalabilidad, flexibilidad, y que además contiene un modelo de consultas e indexación avanzado. Almacena datos en documentos JSON flexibles, es decir, cada documento puede contener diferentes campos y las estructuras de datos se pueden ir modificando. Su logotipo puede verse en Fig. 4-14. [96].



Fig. 4-14. Logotipo de MongoDB [96].

El modelo de documentos concuerda con los objetos del código de la aplicación, lo que facilita trabajar con datos. Las consultas ad-hoc, la indexación y la adición en tiempo real permiten acceder a los datos y analizarlos con gran eficacia [96].

En el diagrama de Fig. 4-15 se observa una representación del almacenamiento de documentos, provenientes de una computadora, en una base de datos Mongo DB.



Fig. 4-15. Diagrama de inserción de documentos a base de datos MongoDB [96].

MongoDB es una base de datos distribuida, por lo que es fácil de usar y proporciona una elevada disponibilidad, escalabilidad horizontal y distribución geográfica [96].

El modelo de documentos de MongoDB resulta muy fácil de aprender y usar, y proporciona a los desarrolladores todas las funcionalidades que necesitan para satisfacer los requisitos más complejos a cualquier escala. Se proporcionan drivers para más de diez lenguajes, y la comunidad ha desarrollado varias decenas más [96].

La información se introduce en forma de “documentos”, que son almacenados dentro de una “colección” que, a su vez, está dentro de una “base de datos”. Los documentos pueden contener información en formato JSON o BSON.

En el proyecto, se utilizó sólo una base de datos, la cual contiene 4 colecciones distintas: Actuadores, Sensores, Usuarios y Fotos. En las colecciones se almacenan los estados actuales de los actuadores con sus fechas de cambio correspondientes, los datos recabados por los sensores del sistema, los usuarios y sus credenciales existentes para ingresar al sitio web y las fotografías capturadas por la cámara, respectivamente. La estructura de la base de datos se ilustra en Fig. 4-16.

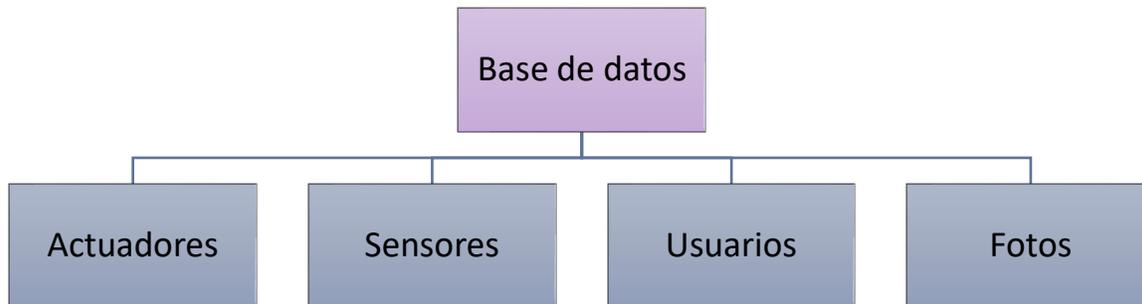


Fig. 4-16. Diagrama de organización interna de la base de datos implementada.

Dentro de cada colección, se encuentran documentos que contienen la información de interés, a modo de campo y dato, en formato JSON.

4.3.3. SERVICIOS

El servidor web, en este caso la Raspberry Pi 3B, tiene en ejecución permanente dos servicios NodeJS. Los microcontroladores interactúan a través de estos servicios con la base de datos y el cliente en un navegador web; sirviendo como intermediario entre todas las partes. Los servicios se encargan de solicitar datos a otra parte del proyecto, almacenarlos y redirigirlos.

Se utilizan paquetes npm en ambos servicios que diversifican las prestaciones de cada uno de ellos, permitiendo así desempeñar tareas adicionales necesarias para el funcionamiento del proyecto.

4.3.3.1. SERVICIO 1 - HTTP

Este servicio, de aquí en adelante *servicio 1*, es el encargado de atender las peticiones HTTP del navegador. Emplea varios paquetes npm que le permiten realizar diversas tareas:

- Crear y utilizar un cliente MQTT.
- Manejar sesiones y autenticaciones de usuarios.
- Montar un servidor HTTP y atender este tipo de solicitudes.
- Conectarse a una base de datos mongoDB.
- Obtener información proveniente del navegador web mediante peticiones POST.

El servicio 1 es el encargado de administrar los archivos que se envían al navegador en función de las solicitudes HTTP, y de manejar el proceso de autenticación. El flujo de datos del inicio de sesión se puede encontrar en Fig. 4-3. Entre sus funciones están:

- Envía el archivo html de la página de inicio de sesión si no hay ninguna sesión activa con ese navegador.
- Recibe los datos de autenticación proporcionados por el usuario y se conecta a la base de datos para contrastarlos con los registros existentes.

- Si la autenticación es infructuosa, devuelve la información del error a la página web solicitando al usuario que verifique la información introducida.
- Si el usuario y la contraseña proporcionados son correctos, entonces crea una sesión y dirige el navegador a la página principal del sitio web.

El servicio 1 es también el intermediario directo entre la página web y los recursos almacenados en la base de datos. Cuando la página web en el navegador solicita una imagen, un conjunto de datos de las variables medidas por los sensores del proyecto o el estado actual de los actuadores (Fig. 4-6, Fig. 4-8 y Fig. 4-10, respectivamente), también es el responsable de buscar los datos solicitados en la colección correspondiente de la base de datos y enviárselos al navegador para ser procesados y visualizados en la página web. Estos flujos de datos entre el navegador, el servicio 1 y los demás elementos involucrados, se documentan a detalle en la sección 4.2. Otra de sus tareas es registrar en la base de datos las peticiones de cambios de estado de los actuadores, que el usuario realiza a través de la página web. Se encarga, así mismo, de enviar las órdenes de actualización al microcontrolador 1 para que modifique los estados de los actuadores. De esta manera, el cambio es ejecutado en el microcontrolador correspondiente y el estado actual del actuador queda registrado en la base de datos. El proceso y el flujo de datos de la actualización del estado de los actuadores se detallan más a fondo en la sección 4.2.6.

Finalmente, el servicio 1 es también quien envía las órdenes de captura de nueva fotografía al microcontrolador 2 y va recibiendo cada uno de los paquetes que éste le envía mediante el protocolo MQTT, para contabilizar la cantidad de datos recibidos. Durante este proceso, además registra y responde las solicitudes HTTP de actualización del porcentaje completado de la recepción de la fotografía, que el servidor web emite. El proceso y flujo de datos de la solicitud de nueva fotografía, se explican a detalle en la sección 4.2.2.

4.3.3.2. SERVICIO 2

Este servicio, de aquí en adelante *servicio 2*, es el encargado de recibir los nuevos datos de los sensores, que el microcontrolador 1 envía vía MQTT, y de registrarlos en la base de datos. También tiene como encomiendas:

- Recibir vía MQTT la solicitud de los últimos estados de los actuadores, hecha por microcontrolador 1.
- Buscar en la base de datos y enviar la orden de actualización correspondiente como respuesta.
- Recibir los paquetes MQTT con los datos de la fotografía que envía microcontrolador 2 tras cada captura.
- Enviar una respuesta MQTT a microcontrolador 2 tras cada paquete recibido, solicitando que continúe con la transmisión de datos.
- Reconstruir la foto al término de la transmisión de los paquetes.
- Almacenar la foto reconstruida, en la base de datos.

Este servicio emplea únicamente dos paquetes npm: uno que le permite crear y utilizar un cliente MQTT y otro para poder conectarse a una base de datos mongoDB.

El flujo de datos del proceso de recepción de los paquetes que contienen información parcial de la fotografía, su reconstrucción y posterior almacenamiento, se encuentran incluidos en Fig. 4-5. El proceso para almacenar los datos de los sensores y la solicitud de microcontrolador 1 del último estado de los actuadores, se muestran en Fig. 4-7 y Fig. 4-9, respectivamente.

4.4. SYSTEM ON CHIP (SOC)

Los microcontroladores contenidos en los SoC ESP8266EX utilizados, se programan utilizando el lenguaje Lua a través del IDE⁴⁰ ESPlorer, con el firmware previamente programado.

Se utilizaron dos ESP8266 debido a tres razones:

1. Porque así se cuenta con más pines disponibles por si desea escalarse el proyecto en etapas posteriores.
2. Ya que colocar la cámara en una ubicación donde se puedan observar el sistema completo y los sensores, cerca de la ubicación de los puntos de medición, brinda una mayor flexibilidad.
3. Debido a las limitaciones de funcionamiento del microcontrolador como tal (Se profundiza más sobre el tema en la sección 5.2).

4.4.1. ESPLORER

Con el firmware programado previamente en los dispositivos, se procede a usar la IDE ESPlorer (mostrada en Fig. 4-17) para poder comunicarnos con los ESP8266, enviarles comandos, desarrollar y cargar los programas que serían utilizados en el proyecto.

ESPlorer es una herramienta multiplataforma (escrita en Java) que incluye las herramientas esenciales para desarrollar software, incluyendo Lua para el NodeMCU y MicroPython. También soporta comandos AT [97].

⁴⁰ Entorno de Desarrollo Integrado, por sus siglas en inglés

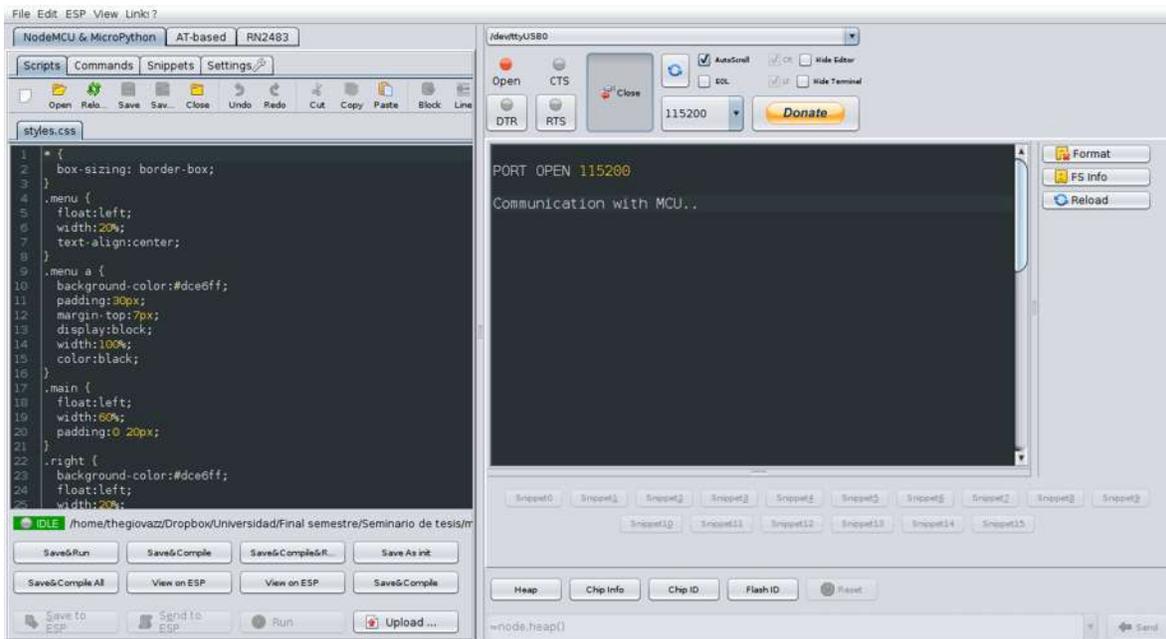


Fig. 4-17. Pantalla principal de ESPlorer.

La sección de la izquierda permite ver, editar o escribir archivos. También cuenta con botones que permiten enviar archivos al NodeMCU. La sección de la derecha es la consola, donde se muestran los mensajes de comunicación serial con la tarjeta de desarrollo. Se puede modificar el *baudrate*⁴¹ al valor apropiado (115200 para el NodeMCU V3), reiniciar la tarjeta, iniciar o terminar la comunicación, entre otros.

4.4.2. LENGUAJE LUA

El firmware del NodeMCU busca un archivo llamado *init.lua* en la raíz de su memoria y en caso de que exista, inicia su ejecución como parte de su secuencia de arranque. Comúnmente este archivo se utiliza para configurar la conexión WiFi y continúa hasta que ésta se encuentre configurada [98].

Lua (cuyo logo se ve en Fig. 4-18) nació en 1993 en el Instituto *TeCGraf* de la *Pontificia Universidad Católica do Rio de Janeiro*, con el objetivo de aumentar las capacidades de otros lenguajes de programación también desarrollados ahí. Tuvo como principio fundamental la simplicidad, partiendo de un lenguaje pequeño, y evolucionando hasta lo que es hoy en día: un lenguaje usado internacionalmente en áreas tan diversas como la creación de videojuegos, el desarrollo de aplicaciones de escritorio, la robótica, etcétera [99].

⁴¹ Número de unidades de señal por segundo.



Fig. 4-18. Logotipo de Lua [99].

Lua **es un lenguaje de extensión** porque no puede ejecutarse por sí mismo. Debe incluirse dentro de otro lenguaje que lo use [99].

Lua **es interpretado** porque no se produce un archivo ejecutable a partir de él, sino que sus instrucciones se ejecutan a través de otro programa, conocido como *intérprete* [99].

Lua **tiene manejo de memoria automático**. El programador no debe preocuparse por liberar la memoria RAM que su programa ya no está usando [99].

Lua **es un lenguaje de tipado dinámico** ya que asigna automáticamente el tipo de dato correcto a cada variable, pudiendo ser un número, un texto o una estructura de datos [99].

Lua **es portable**, porque puede funcionar en diversidad de sistemas operativos y dispositivos; Por ejemplo, Windows, Mac, Linux, Android, iOS Windows Phone, etcétera [99].

Lua **es libre**. Pertenece a la filosofía del software libre, es gratuito y su código puede ser visto o modificado por cualquier interesado [99].

4.4.3. PROGRAMAS DE LOS MICROCONTROLADORES

Se emplearon dos SoC para llevar a cabo todas las tareas requeridas para el proyecto. A cada uno de ellos se le asignan tareas diferentes; por tal motivo, su programación se realiza de manera independiente.

4.4.3.1. MICROCONTROLADOR DEDICADO A SENSORES Y ACTUADORES

El microcontrolador responsable de manipular y recibir la información proveniente de los sensores, así como de efectuar cambios en los actuadores del sistema (de aquí en adelante, *microcontrolador 1*), cuenta con una comunicación vía WiFi con el cliente de la página web, a través del protocolo MQTT; por este medio el cliente solicita las modificaciones a los actuadores. Las solicitudes llegan al microcontrolador, quien interpreta la información para modificar después los estados de los actuadores. También tiene conexión con dos servicios NodeJS que se ejecutan en el servidor web: Del servicio 1 recibe la orden de actualización del estado de un actuador, que se origina desde la página web, mientras que el servicio 2 acepta las peticiones para informar acerca del estado del sistema basándose en la información almacenada en la base de datos. Esta información es muy útil principalmente al arranque del sistema, pues permite asegurarse de configurar el último estado deseado y establecido por el usuario.

Este servicio también recibe los datos provenientes de los sensores para almacenarlos en la base de datos.

Las mediciones de los sensores inician por los eventos generados en los diferentes temporizadores del microcontrolador. Existe un temporizador por cada uno de los sensores. Cada temporizador (o timer) se ejecuta cada cierto tiempo, dependiendo de la naturaleza de la magnitud a medir, y asegurándose de no activarse de manera simultánea.

Las lecturas llegan al microcontrolador por uno de sus pines digitales (en el caso del sensor de temperatura y humedad) y por el ADC (en el caso de la iluminación, la CE y el pH). Como sólo se tiene un canal de entrada del convertidor A/D del microcontrolador, se utiliza un multiplexor analógico para seleccionar la señal del sensor de interés que se enviará al ADC.

En cada evento de un timer, se llama a la rutina correspondiente para la lectura del sensor. Después de la lectura, el dato obtenido se envía vía mensaje MQTT junto con la información del sensor del que proviene, al servicio 2 NodeJS. Este es el responsable de recibirlo e introducirlo en la base de datos.

Las modificaciones a los actuadores se realizan cuando se ejecuta el evento de recepción de MQTT y entre la información recibida, se encuentra la instrucción de actualizar el estado del actuador, junto con el nuevo valor o estado deseado.

Al iniciar el programa principal del microcontrolador 1, se configura el estado inicial de la bomba como apagado y el de los LED al 1% de intensidad con el fin de tener un indicador visual en el sistema físico de que el arranque está en proceso, y evitar riegos indeseados en caso de reinicio o desconexión del microcontrolador, enseguida, se configuran e inicializan los PWM's y las variables que van a utilizarse en el programa. Posteriormente, se inicializan los temporizadores para cada uno de los sensores, se define la configuración de la conectividad WiFi y se intenta realizar una conexión: en caso de no ser exitosa, se intenta nuevamente; de lo contrario, se configura el cliente MQTT y se realiza la conexión con el servidor MQTT. Para finalizar, se solicita al servicio 2, el último estado almacenado en la base de datos de la bomba y los LED, para configurarlo en el sistema. El flujo del programa principal del microcontrolador 1 se muestra en Fig. 4-19.

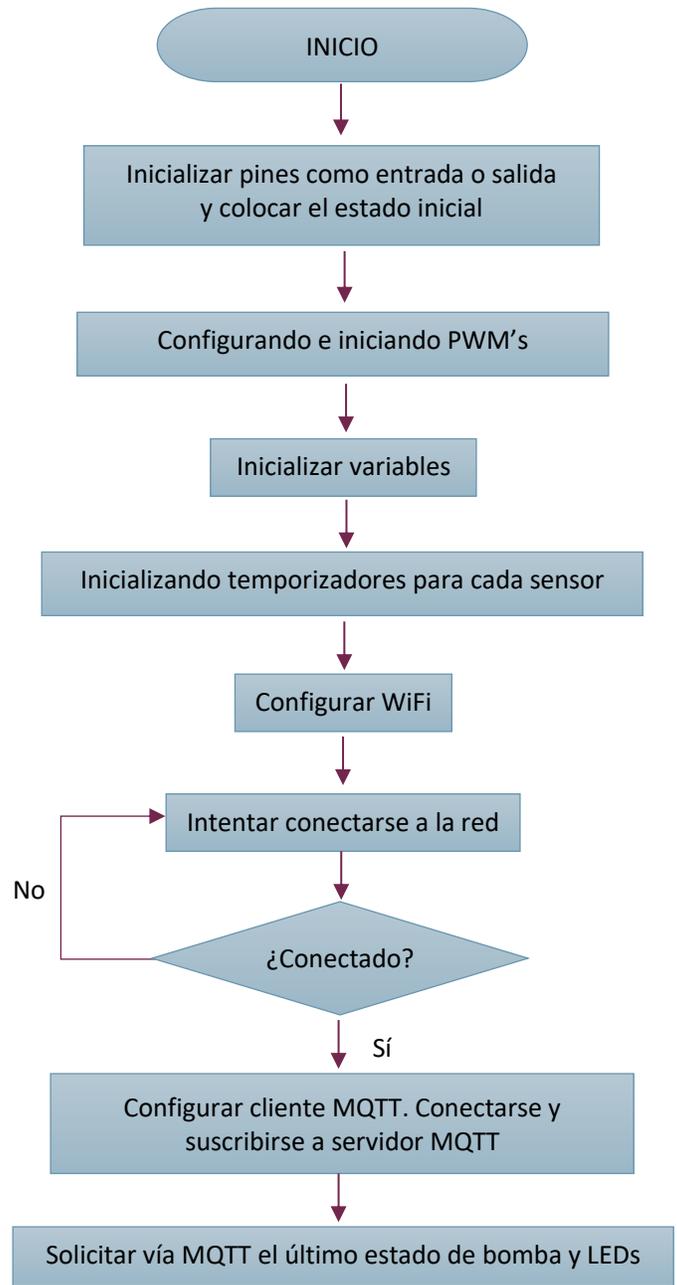


Fig. 4-19. Diagrama de flujo del programa principal del microcontrolador 1.

Los diagramas de flujo correspondientes a las ejecuciones asíncronas de los eventos iniciados al recibir un comando por MQTT o por alguno de los timers, se muestran en Fig. 4-20 y Fig. 4-21, respectivamente.

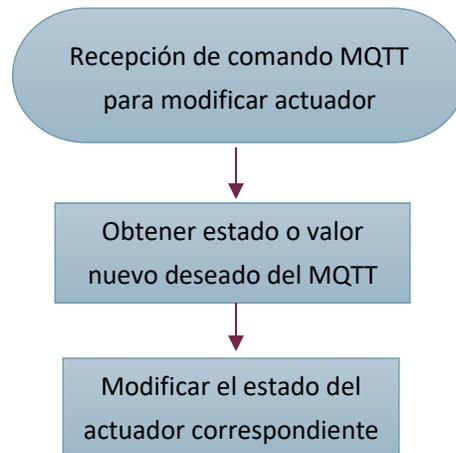


Fig. 4-20. Diagrama de flujo de rutina asíncrona ejecutada ante mensaje MQTT de actualizar estado de actuador.

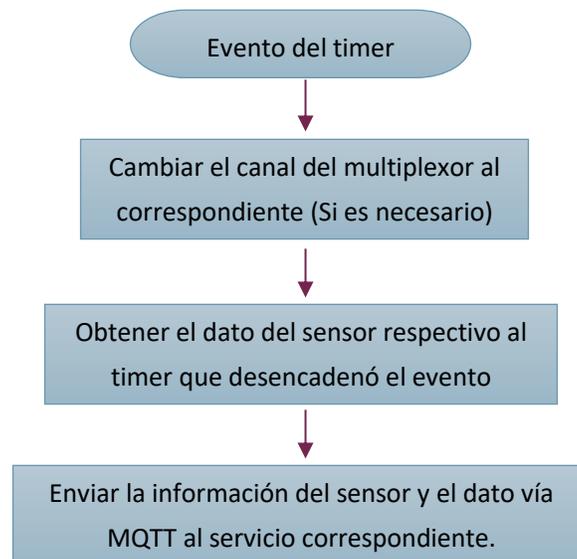


Fig. 4-21. Diagrama de flujo de rutina asíncrona ejecutada ante temporizador.

4.4.3.2. MICROCONTROLADOR DEDICADO A DISPOSITIVOS PERIFÉRICOS

El microcontrolador dedicado a dispositivos periféricos (de aquí en adelante, *microcontrolador 2*) se dedica exclusivamente a la administración de la cámara serial para obtener una nueva fotografía. El usuario puede solicitar que se tome una fotografía a través del portal web del sistema.

El comando viaja del navegador al servicio 1 NodeJS en el servidor web, el cual, envía una orden de captura de nueva fotografía a este microcontrolador vía MQTT. Una vez recibida la orden de captura, el

microcontrolador comienza a ejecutar la rutina de captura de nueva fotografía, y emplea los comandos AT para controlar la cámara vía comunicación serial.

Al tomar la fotografía, ésta es almacenada en la memoria del microcontrolador y, posteriormente, comienza el proceso de envío vía MQTT al servidor web. El proceso de envío inicia con un mensaje que indica la cantidad de datos del archivo a enviar. Se van tomando fragmentos de 900 bytes del archivo y se envía cada uno de ellos en un mensaje MQTT; además se anexa el número de paquete correspondiente, al servicio encargado de recibirlos e ir almacenando la información para su posterior reestructuración.

Al término del envío de cada paquete, el microcontrolador espera un mensaje vía MQTT proveniente del servicio 2, que indica que puede continuar con el envío. Una vez recibido dicho mensaje, el envío del siguiente paquete se realiza. Esta acción se repite hasta que finaliza el envío del último paquete del archivo, entonces se envía un último mensaje MQTT que indica al servicio que la transmisión ha finalizado.

A pesar de que el protocolo MQTT especifica que en cada uno de sus *payloads* (contenido del mensaje) pueden enviarse hasta 256 MB, el módulo MQTT utilizado por el NodeMCU especifica que el máximo tamaño de datos que pueden enviarse en un mensaje MQTT (incluyendo cabeceras, topic y toda la información) es de 1024 bytes. El flujo de datos del proceso de captura y envío de una nueva fotografía, se muestra en Fig. 4-5. [100].

El flujo del programa principal del microcontrolador 2 se ilustra en Fig. 4-22.

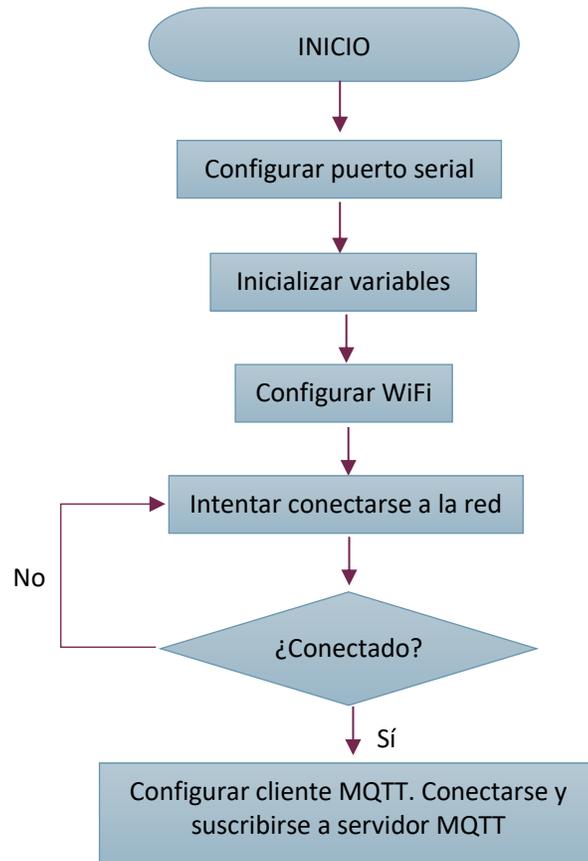


Fig. 4-22. Diagrama de flujo del programa principal del microcontrolador 2.

Los diagramas de flujo correspondientes a las ejecuciones asíncronas de los eventos por la recepción de comandos por MQTT, se muestran en Fig. 4-23.

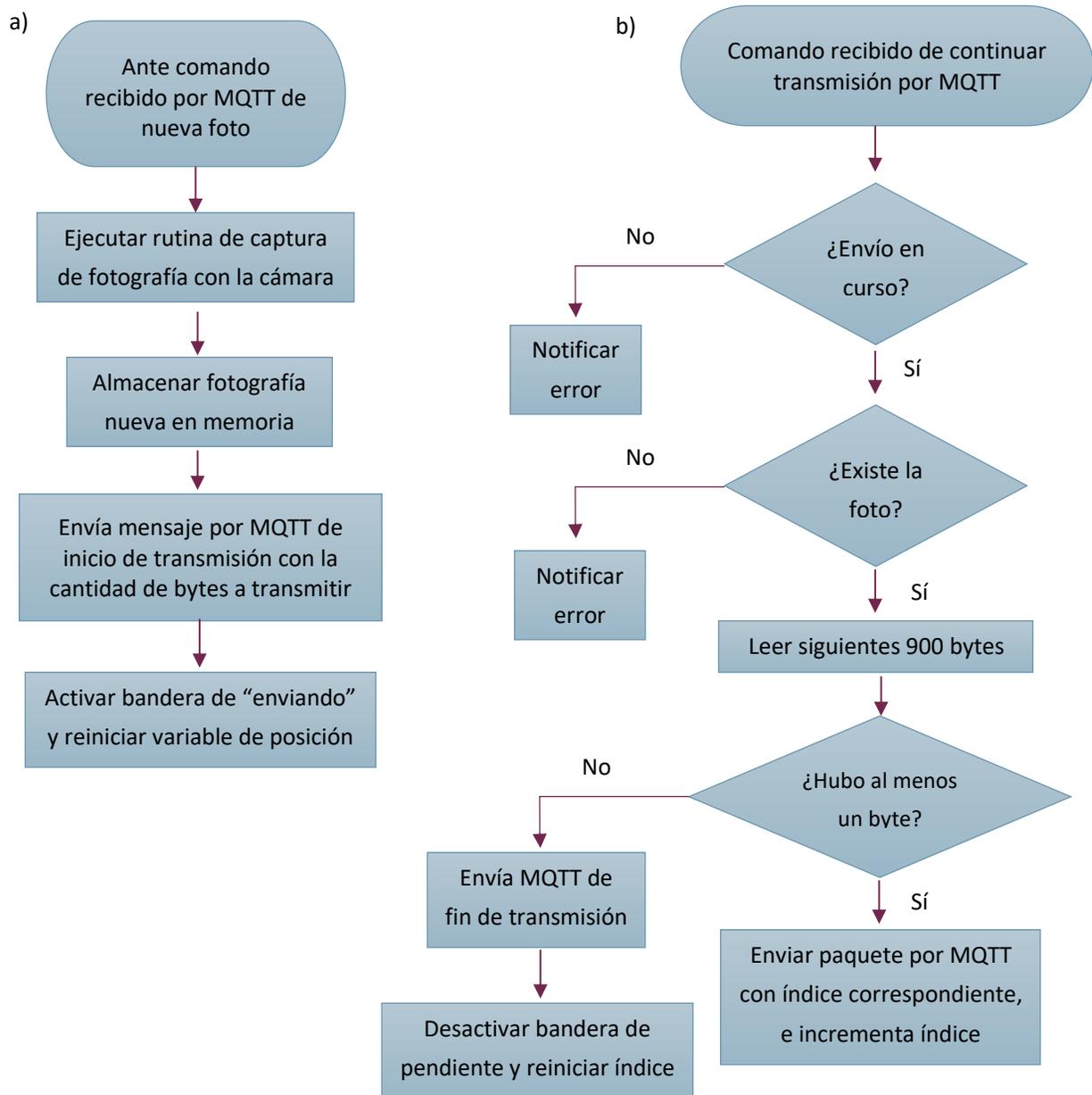


Fig. 4-23. Diagramas de flujo de ejecuciones asíncronas ante instrucciones MQTT. a) Ante instrucción de nueva fotografía. b) Ante instrucción de continuar con la transmisión de la fotografía.

El diagrama de flujo de la rutina de comandos que el microcontrolador envía para capturar una fotografía con la cámara, se muestra en Fig. 4-24.

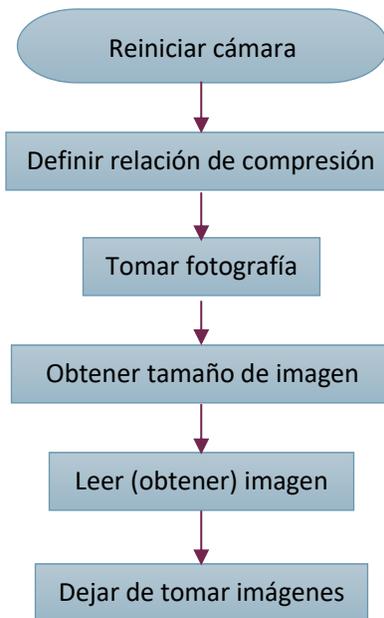


Fig. 4-24. Diagrama de flujo de rutina de instrucciones enviadas a la cámara serial para obtener una imagen.

La cámara requiere un periodo de tiempo de entre 2 y 3 segundos después de recibir el comando de confirmación de reinicio, antes de enviarle el de tomar fotografía.

4.5. SITIO WEB

El sitio web está construido utilizando Javascript, CSS, HTML, peticiones AJAX y dos servicios NodeJS. Consta de dos páginas web distintas. La primera de ellas es la página de autenticación y la segunda es la página principal, la cual, modifica su contenido en función del apartado seleccionado.

El sitio web es accesible, siempre que el servidor web se encuentre activo, a través de la dirección feliche.tk:3000, lo que significa que se accede a través del puerto 3000. La página que se recibe primero es la página de autenticación (véase la sección 4.5.1). Una vez realizada la autenticación, se recarga para mostrar la página principal (de la que se habla en la sección 4.5.2).

4.5.1. PÁGINA DE AUTENTICACIÓN

Es la página que se carga por defecto cuando no se tiene una sesión iniciada. Cuenta con dos bloques para introducir información y un botón de enviar, en esos bloques se introducen las credenciales para autenticarse y, de esta forma, iniciar sesión. La vista general de esta página se muestra en Fig. 4-25.

Del proceso de autenticación y el flujo de datos correspondiente se habla en la sección 4.2.1.



Fig. 4-25. Vista de la página de autenticación en un navegador.

4.5.2. PÁGINA PRINCIPAL

Después de iniciar sesión exitosamente, a través de la página de autenticación, el sitio web redirige a la página principal.

Está conformada por varios elementos que permiten una interacción con el usuario de diferentes maneras. Dichos elementos se explican a detalle a continuación:

I. Barra de título

Esta barra se muestra siempre en la parte central superior de la página y sirve para indicar mediante texto, el contenido que se está visualizando en el contenedor central.

II. Barra lateral de navegación

La barra de navegación lateral (que se ilustra en Fig. 4-26) está compuesta de 4 polígonos negros (de aquí en adelante, botones), distribuidos en la parte izquierda de la página, que emergen hacia la derecha y

cambian el cursor por la “manita” de clic a un hipervínculo cuando éste pasa por encima de ellos. Cada uno tiene una leyenda escrita y un ícono representativo de la vista a la que dan acceso al ser cliqueados.



Fig. 4-26. Barra lateral de navegación de la página web principal.

La leyenda de un polígono sólo es visible cuando el cursor se posa sobre él, mientras que su ícono es visible todo el tiempo, como se muestra en Fig. 4-27.

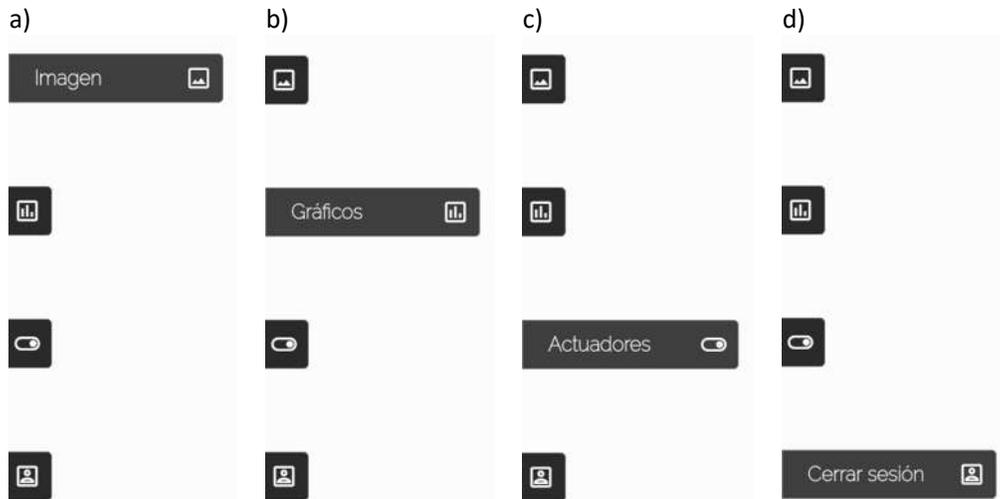


Fig. 4-27. Comportamiento expansivo de respuesta de los íconos de la barra de navegación lateral al cursor. a) ícono Imagen. b) ícono Gráficos. c) ícono Actuadores. d) ícono Cerrar sesión.

Listando de la parte superior de la página a la inferior, se encuentran los botones de Imagen, Gráficos, Actuadores y Cierre de sesión, respectivamente.

La vista por defecto al cargarse la página principal, es la Vista *Imagen*. El botón inferior de la barra lateral de navegación sirve exclusivamente para cerrar la sesión actual. Accionarlo, redirige al navegador a la página de autenticación y destruye la sesión del navegador, haciendo imposible acceder a la página principal sin volver a introducir las credenciales correctamente.

III. Contenedor central

El contenedor central, como su nombre lo indica, se encuentra en el centro de la página. Dentro de él, se encuentran los elementos de interés de cada una de las vistas: La imagen para la vista *Imagen*, los gráficos para la vista *Gráficos* y los íconos de los actuadores, con sus indicadores de estado correspondientes, para la vista *Actuadores*. El contenedor central se encuentra destacado con bordes de color rojo, en la imagen de Fig. 4-28.

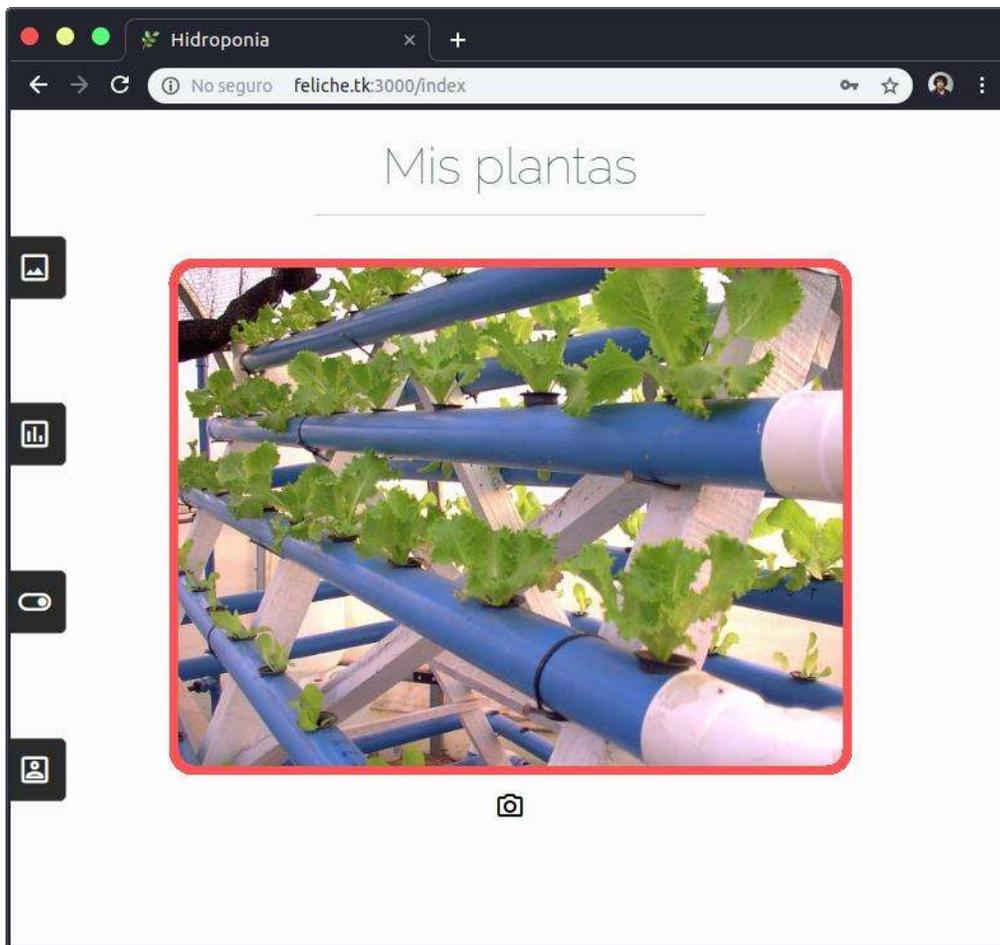


Fig. 4-28. Contenedor central de la página principal.

IV. Barra inferior de opciones

La barra inferior de opciones (véase Fig. 4-29) es una barra que se ubica bajo el contenedor central que contiene íconos que permiten seleccionar las opciones disponibles en la vista *Gráficos*. Esta barra desaparece durante la vista *Actuadores*.



Fig. 4-29. Barra inferior de opciones de la página principal.

Cada ícono cambia ligeramente de tonalidad cuando el cursor pasa sobre él, cambia el cursor por la “manita” y despliega un texto informativo respecto a la opción de la que se trate. Un ejemplo de este comportamiento puede verse en Fig. 4-30.



Fig. 4-30. Comportamiento de respuesta al cursor, de los íconos de la barra inferior de opciones.

La opción disponible durante la vista *Imagen* es la captura de una nueva fotografía; las opciones disponibles durante la vista *Gráficos* son Iluminación, Temperatura, Humedad, pH y Concentración de nutrientes.

4.5.2.1. VISTA IMAGEN

La vista *Imagen* permite la visualización de las imágenes capturadas por la cámara web. Cuando esta vista es cargada, el sitio web solicita a la base de datos la última fotografía almacenada y, después de recibirla, la muestra en el contenedor central. En esta vista, a través de la barra inferior de opciones, puede solicitarse la captura de una nueva fotografía que se muestra en el contenedor central luego de ser capturada, almacenada en la base de datos, y recibida por el servicio correspondiente. El flujo de datos de los procesos de captura de nueva fotografía y de solicitud de la última almacenada, se describe en la sección 4.2.2.

Una captura de pantalla de la página web con la vista *Imagen* completa se muestra en Fig. 4-31.

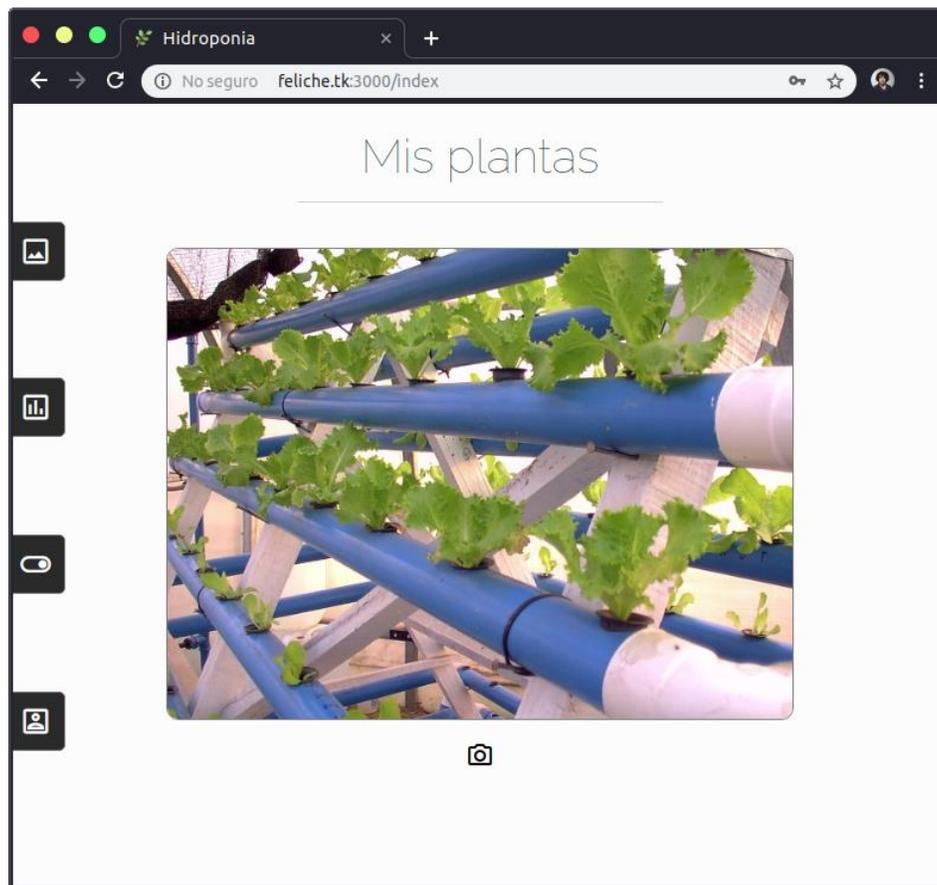


Fig. 4-31. Vista *Imagen*, de la página principal.

4.5.2.2. VISTA GRÁFICOS

La vista *Gráficos* (mostrada en Fig. 4-32) permite ver los datos recabados por los sensores que están almacenados en la base de datos, a través de gráficos que se muestran en el contenedor central. Los últimos 10 datos de la variable correspondiente son mostrados.

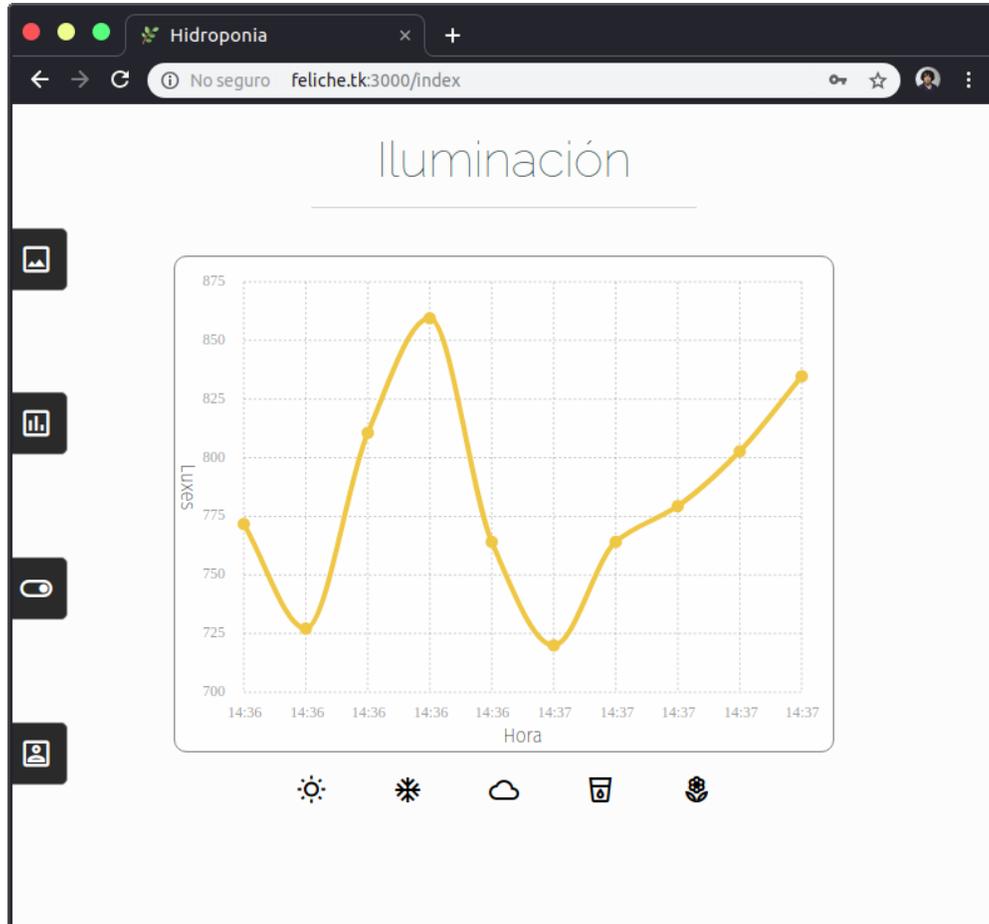


Fig. 4-32. Vista *Gráficos*, de la página principal.

La barra inferior de opciones, contiene en esta vista 5 íconos. Cada uno de ellos cambiará el gráfico que se esté visualizando en pantalla por el correspondiente a cada una de las 5 variables medidas por el sistema, así como el contenido de la barra de título, que indicará de qué gráfico se trata. De izquierda a derecha, aparecerán los íconos de Iluminación, Temperatura, Humedad, pH y Concentración de nutrientes. Cada uno de los gráficos se muestra en diferente color para facilitar su identificación. El gráfico seleccionado por defecto, cuando se cambia a la vista *Gráficos*, es el de Iluminación.

Siempre que la vista *Gráficos* se encuentre en pantalla, el sitio web estará solicitando periódicamente al servidor web, los últimos datos almacenados mediante peticiones AJAX. Esto con el fin de mantener el gráfico en constante actualización. Acerca de este proceso se habla más a detalle en la sección 4.2.4.

Algunos de los datos recabados por los sensores y posteriormente almacenados en la base de datos, se encuentran en unidades del ADC del microcontrolador. Es entonces el navegador, a través del programa

de JavaScript, el encargado de convertirlos a las unidades correspondientes y mostrarlos posteriormente en los gráficos de manera que el usuario pueda interpretarlos con facilidad.

4.5.2.3. VISTA ACTUADORES

La vista *Actuadores*, que puede verse en Fig. 4-33, muestra de manera sencilla los actuadores disponibles en el sistema (la bomba y la iluminación LED) y su estado actual en el contenedor central. Permite también modificar su estado actual: con un interruptor en el caso de la bomba y con un selector de color (en notación hexadecimal RGB) en el caso de los LED. Una explicación más detallada del proceso de cambio en el estado de un actuador y su flujo de datos, puede encontrarse en la sección 4.2.6.

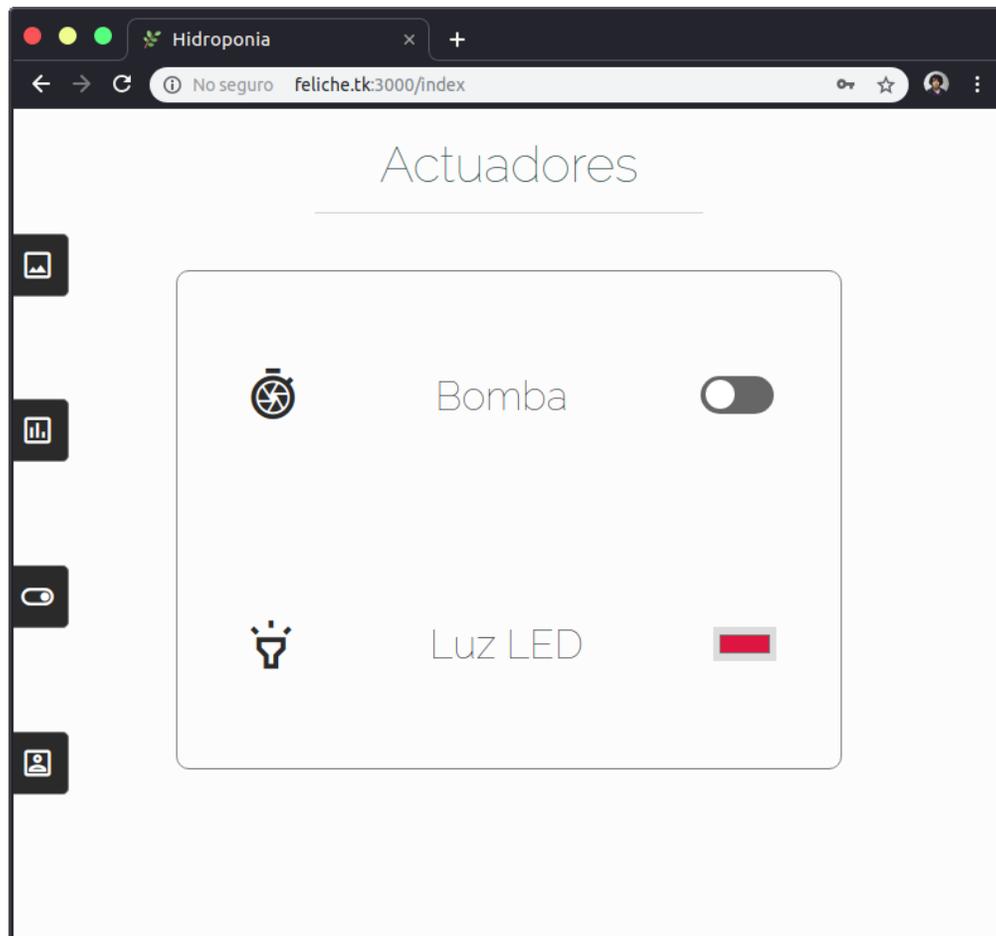


Fig. 4-33. Vista *Actuadores*, de la página principal.

Cada que esta vista es seleccionada, el sitio web envía una solicitud al servidor web para que éste, a su vez, solicite a la base de datos el último estado almacenado de los actuadores, con el fin de visualizarlo correctamente en la página web. Todo este proceso se describe en la sección 4.2.5.

CAPÍTULO 5. PRUEBAS Y RESULTADOS

En este capítulo pueden encontrarse algunas pruebas realizadas al sistema o a los elementos que lo conforman. Se muestran también capturas de pantalla de la interacción con el sitio web.

Finalmente, pueden encontrarse las justificaciones de algunas decisiones tomadas a lo largo del desarrollo del proyecto.

5.1. PRUEBAS CON LA CÁMARA SERIAL

Para probar que la cámara funciona correctamente y realizar pruebas con ella, previamente a su integración, se hizo uso de un convertidor USB-Serial (el mostrado en Fig. 5-1) para poder realizar conexión con una computadora. La conexión entre la cámara y el convertidor se muestra en Fig. 5-2.



Fig. 5-1. Convertidor USB-Serial utilizado en el proyecto.

El software utilizado para la prueba fue el LS-Y201 proporcionado para Windows, por LinkSprite. Dicho programa se encuentra disponible para su descarga en:

<https://cdn.sparkfun.com/datasheets/Sensors/LightImaging/LSY201.exe>

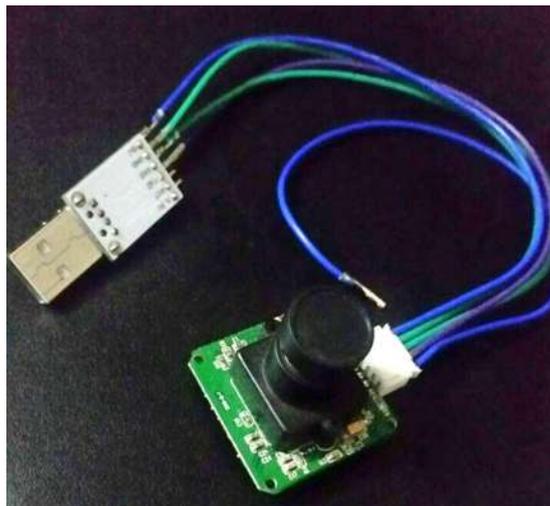


Fig. 5-2. Cámara serial conectada al convertidor USB-Serial.

La interfaz principal del programa luce como se ilustra en Fig. 5-3. Una vez conectada la cámara, se seleccionó el puerto COM que la computadora le asignó y el baud rate a utilizar. El baud rate predefinido para este modelo es de 38400.

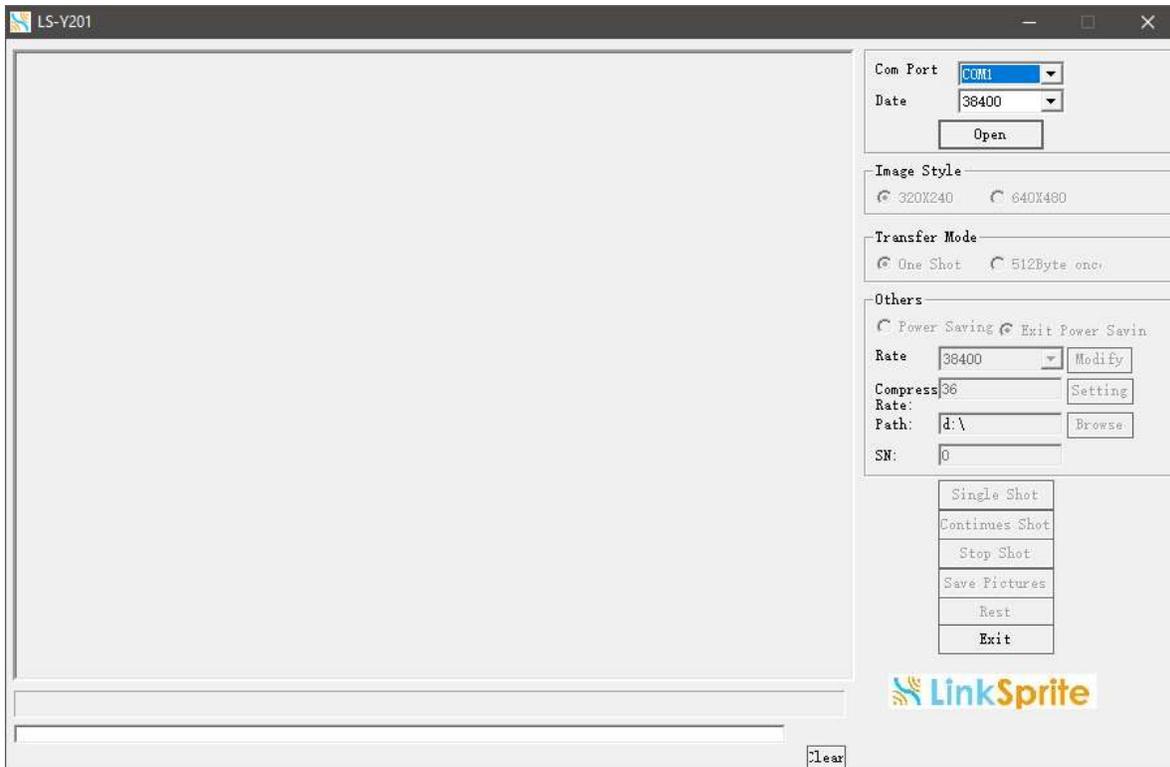


Fig. 5-3. Pantalla principal del software LS-Y201.

En este caso, fue el puerto COM4 (vea Fig. 5-4).

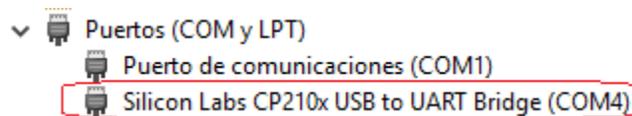


Fig. 5-4. Puerto COM asignado por la computadora, al convertidor USB-Serial.

Se inició la comunicación con la cámara, con el botón *Open*, como se puede observar en Fig. 5-5.

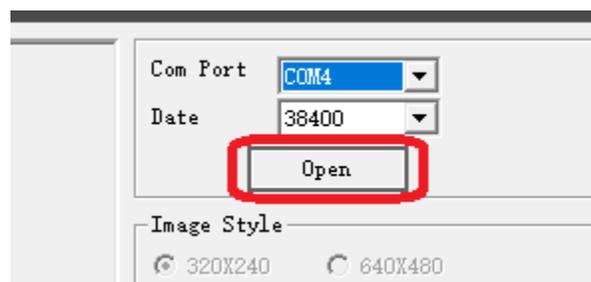


Fig. 5-5. Estableciendo conexión con cámara serial.

El software permite seleccionar entre dos posibles resoluciones para las imágenes a capturar, el modo de transferencia, si se quiere emplear el modo de ahorro de energía o no, el factor de compresión, una ruta en el equipo para almacenar las imágenes capturadas, entre otras opciones.

Para finalizar la prueba, luego de haber seleccionado las configuraciones deseadas (que bien pueden quedarse como están por defecto), se hace click en el botón *Single Shot*, para obtener una imagen de la cámara y visualizarla en el programa (ver Fig. 5-6). Es posible entonces tomar una nueva fotografía, pulsando el mismo botón, o elegir guardar la imagen recibida con el botón *Save Pictures*.

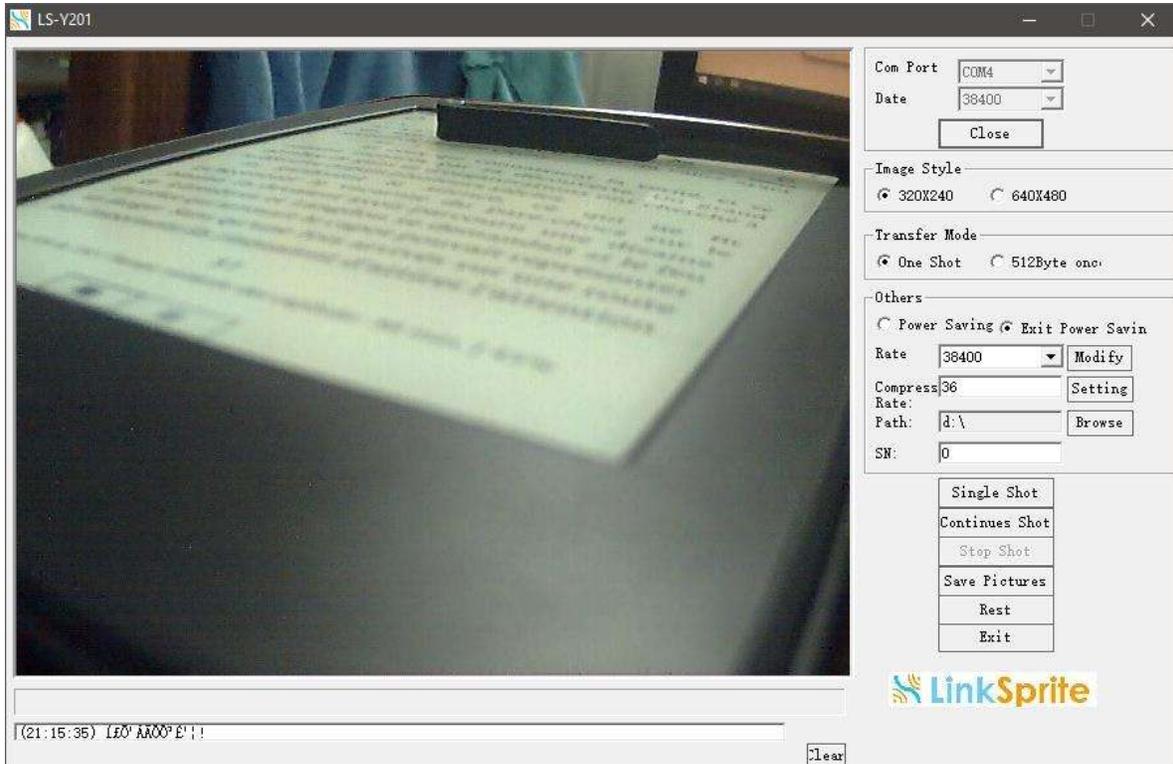


Fig. 5-6. Visualizando imagen tomada con la cámara serial, en el programa LS-Y201.

De esta forma, se procede a realizar pruebas de captura de imágenes, variando los parámetros hasta encontrar los óptimos para ser usados en el proyecto.

5.2. USO DE DOS SOC

Durante una gran parte del desarrollo del proyecto, se intenta que todas las tareas las realizara un único microcontrolador. Esto, buscando aprovechar al máximo sus capacidades. El desarrollo del proyecto se realiza por etapas: se desarrolla una característica a la vez y al final se integran todas para conjuntar el proyecto completo. Fue en esa última etapa cuando se comenzaron a encontrar fallas en la programación del microcontrolador. Era capaz de desempeñar todas sus tareas si las ejecutaba de forma individual, pero cuando se integran todas las tareas a realizar, de forma simultánea, comienza a tener

comportamientos inesperados e impredecibles como reinicios, fallas en la ejecución del programa, eliminación de datos de variables, problemas con la memoria, entre otros.

Se realizaron una meticulosa investigación al respecto en internet y una gran cantidad de pruebas, hasta que se encontró que existían otras personas que también habían tenido problemas similares con ese mismo modelo de SoC, como lo detalla el artículo disponible en [101] (Mismo en el que puede encontrarse un comentario mío). En este artículo detallan cómo se enfrentaron a esos mismos problemas (así como un comportamiento extraño del ADC) al desarrollar con este SoC y el lenguaje Lua, y proponen como alternativa el uso de la IDE de Arduino y el lenguaje de esa plataforma. Concluyen también, que Lua es un lenguaje excesivamente de alto nivel para un chip tan pequeño como el del ESP8266 y sus 4MB de memoria, los cuales son suficientes para un programa grande y complejo en lenguaje C, pero no para uno en lenguaje Lua. Mencionan también, que programando con la IDE de Arduino, que se traduce a lenguaje C para el ESP8266, pueden conocer exactamente la cantidad de SRAM⁴² que están utilizando y cuánta les queda disponible para seguir codificando; y que no tienen necesidad de desconectar y reconectar el dispositivo, reiniciarlo o recargarle el firmware durante el desarrollo.

Al final, desarrollar en dos SoC resultó ser beneficioso, dada la naturaleza del proyecto, porque así se cuenta con más pines disponibles. De esta manera, el proyecto se hace más fácilmente escalable, y permite añadir otras funcionalidades en una posterior etapa de desarrollo. Además, permite una instalación más sencilla, ya que es necesario colocar la cámara y su respectivo microcontrolador, ubicada lo suficientemente lejos de la instalación hidropónica de PVC como para que las fotografías tengan una perspectiva adecuada.

5.3. INICIO DE SESIÓN

En esta sección se muestran capturas de pantalla del proceso de inicio de sesión en el sitio web. Todo comienza ingresando a la dirección feliche.tk:3000 con los servidores debidamente puestos en marcha de manera previa. Entonces se carga la página de inicio de sesión, mostrada en Fig. 4-25.

⁴² Memoria estática de acceso aleatorio, por sus siglas en inglés.



Fig. 5-7. Introduciendo credenciales a la página de autenticación.

Se introducen las credenciales correctas, y se presiona la tecla Intro o el ícono de la palomita de la página, como se muestra en Fig. 5-7. Si el usuario y la contraseña son correctas, se redirigirá entonces a la página principal.

5.4. SOLICITUD DE NUEVA FOTOGRAFÍA

La página principal muestra la vista *Imagen* con la última fotografía almacenada en la base de datos, tras ser completamente cargada. Para solicitar la captura de una nueva fotografía, hay que pulsar sobre el ícono de la cámara, como se muestra en Fig. 5-8.

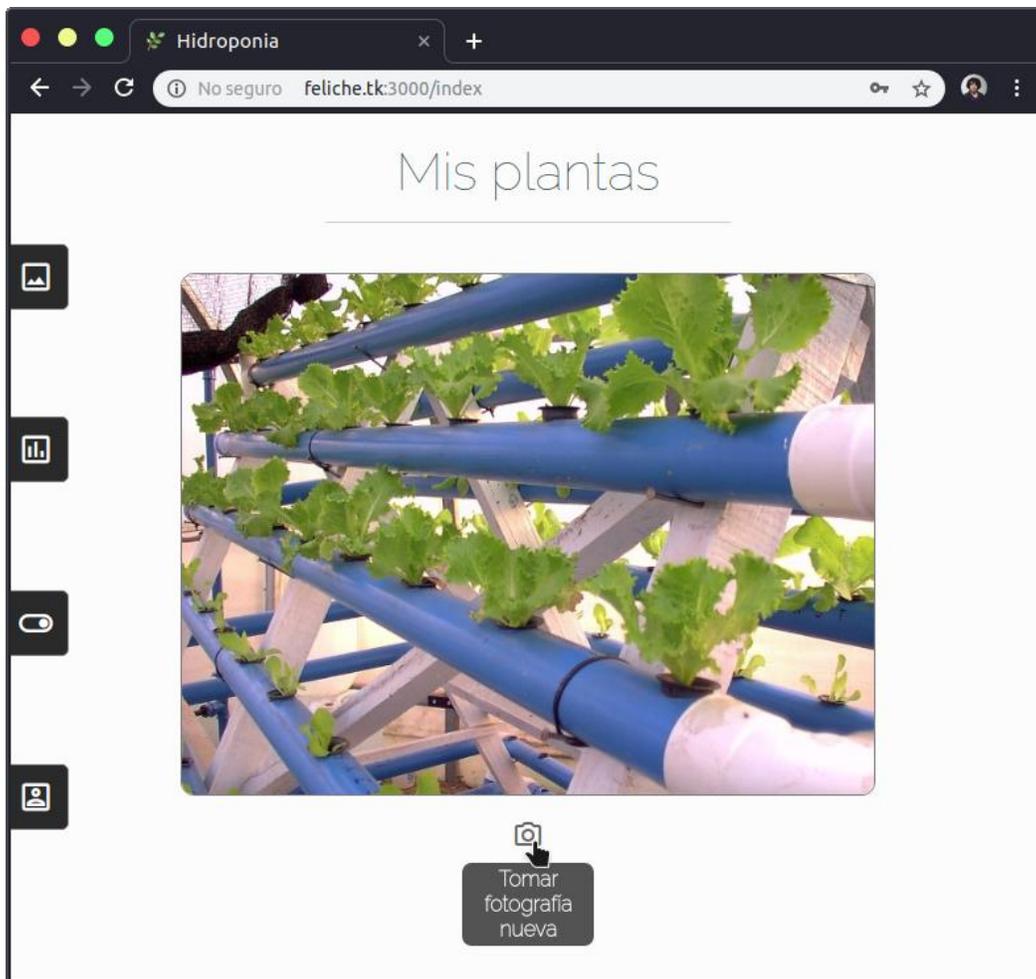


Fig. 5-8. Solicitando nueva captura de fotografía desde la vista *Imagen* de la página principal.

Hecho esto, el contenedor central modifica su contenido a una barra de progreso y un texto informativo, que indica el estado de la nueva captura, como se aprecia en Fig. 5-9 y Fig. 5-10.

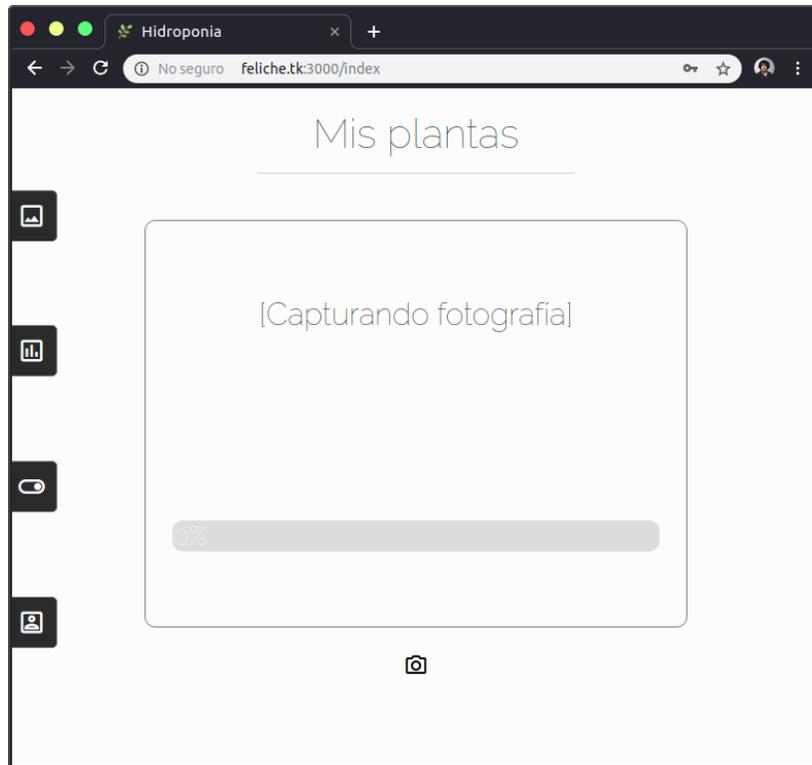


Fig. 5-9. Contenedor central que se muestra durante el proceso de captura de la nueva fotografía, en la vista *Imagen* en la página principal.

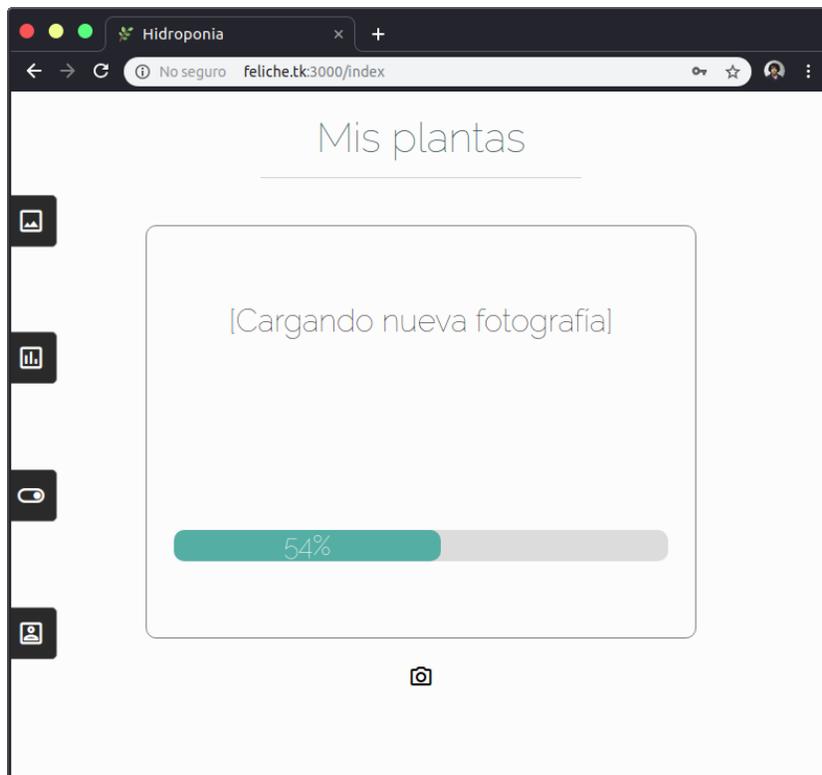


Fig. 5-10. Contenedor central que se muestra durante el proceso de recepción de la nueva fotografía, en la vista *Imagen* en la página principal.

Una vez que la nueva imagen esté lista, será almacenada en la base de datos y mostrada en el contenedor central de la página. Véase Fig. 5-11.

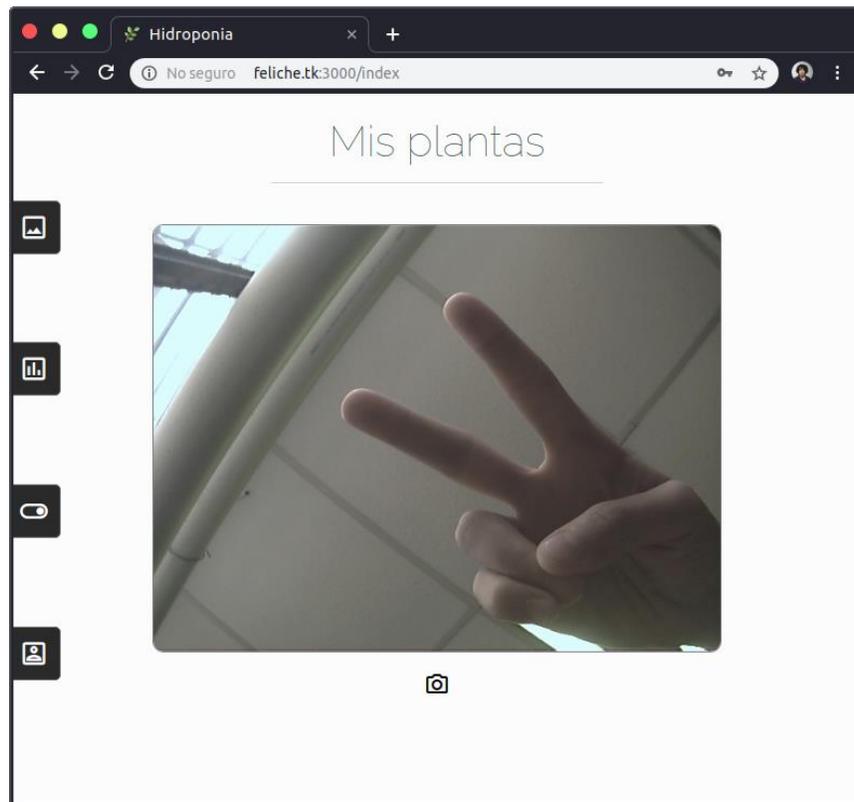


Fig. 5-11. Visualización de la nueva imagen capturada por la cámara tras la ejecución de la captura solicitada.

5.5. VISUALIZACIÓN DE DATOS EN UN GRÁFICO

Para acceder a los gráficos, hay que ir a la vista *Gráficos*, pulsando el botón correspondiente en la barra de navegación lateral. Hecho esto, se muestra el gráfico de Iluminación, como se ve en Fig. 4-32.

Si se desea observar el gráfico de otra variable, como la temperatura, hay que pulsar sobre el ícono de la variable de interés, disponible en la barra inferior de opciones, tal como se muestra en Fig. 5-12, donde se pulsa el ícono para ver el gráfico de la temperatura.

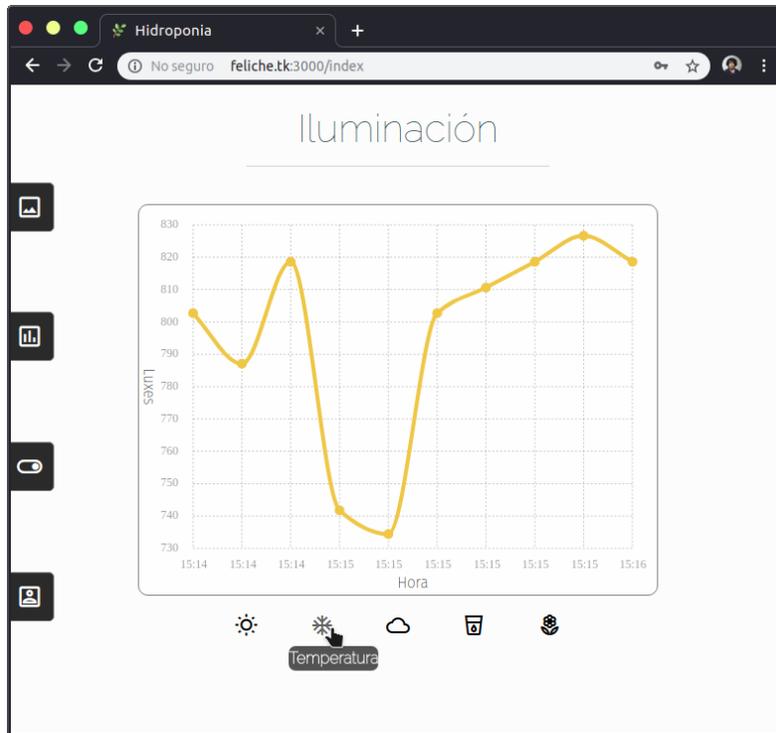


Fig. 5-12. Procedimiento para cambiar el gráfico en visualización por el de temperatura, a través de la barra inferior de opciones.

De esta forma, el gráfico se recargará, el título de la barra de título cambiará y se mostrará ahora la gráfica de los datos de la variable seleccionada. Ver Fig. 5-13.

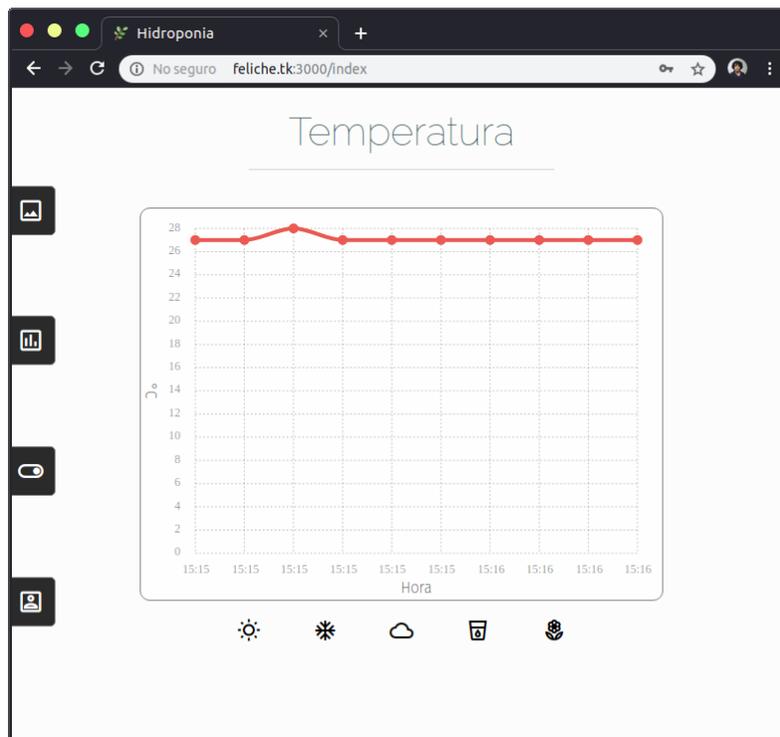


Fig. 5-13. Gráfico de temperatura, mostrado en la vista Gráficos de la página principal.

5.6. CONSULTA Y CONTROL DEL ESTADO DE LOS ACTUADORES

Para conocer el estado actual de los actuadores, hay que ir a la vista *Actuadores*, disponible haciendo clic en el botón correspondiente en la barra lateral de navegación. Entonces, el contenedor central muestra los actuadores, representados por íconos, y su estado actual a través de los selectores al lado derecho de las descripciones de cada actuador. Véase Fig. 4-33.

Para modificar el estado de la bomba, basta con hacer clic en el interruptor junto a su nombre. Este cambia su estado, indicando que el cambio fue realizado con éxito.

Para modificar el color o intensidad de la iluminación LED, hay que pulsar sobre el rectángulo de selección de color. Este abre una ventana con toda una gama de colores en su notación hexadecimal, como puede observarse en Fig. 5-14.

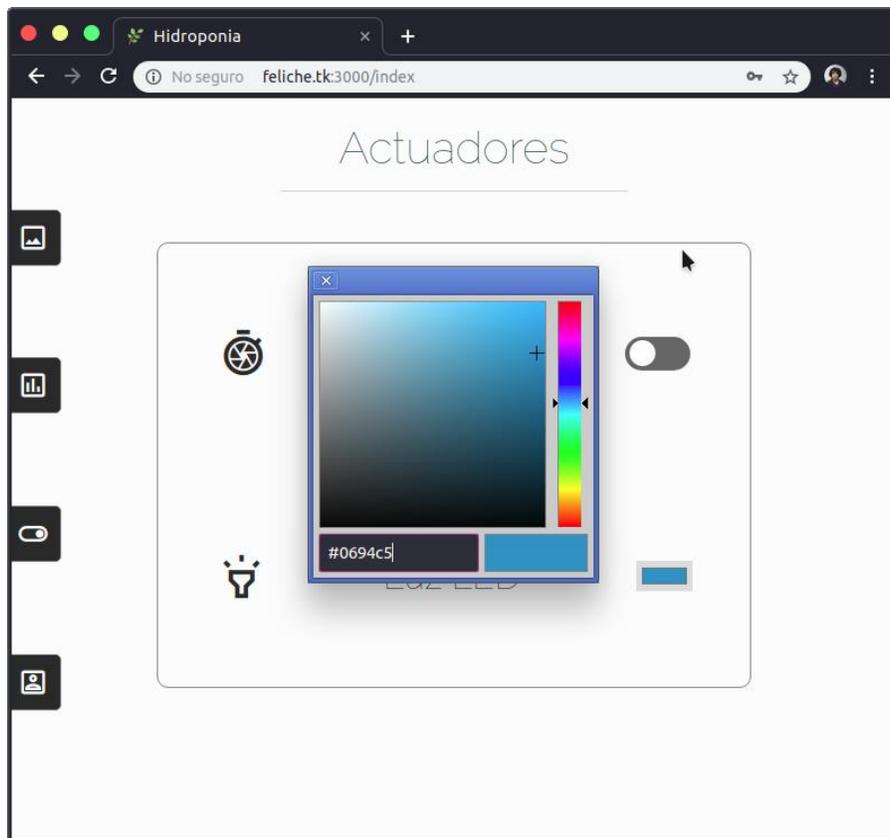


Fig. 5-14. Ventana emergente de selección de color en formato hexadecimal, para cambiar el tono y brillo de la iluminación LED.

Se puede hacer clic en el color de interés, o introducir directamente el valor hexadecimal RGB deseado, y pulsar en la cruz para cerrar la ventana emergente de selección de color y enviar la orden de cambio al microcontrolador correspondiente.

El propósito de introducir la opción de cambiar el color e intensidad luminosa de los LED, es permitir al usuario configurar las condiciones óptimas de iluminación, dependiendo del tipo de cultivo y la etapa de crecimiento en la que este se encuentre. Numerosas investigaciones (como [102] y [103]) concluyen que

la iluminación juega un rol vital en el crecimiento y desarrollo de las plantas. Se ha demostrado que modificar la longitud de onda de la iluminación artificial utilizada para cultivar lechugas y otros alimentos, repercute en el tamaño del cultivo final, el tiempo de crecimiento de la planta, la duración de cada una de las etapas de crecimiento del cultivo, entre otras. El propósito de este proyecto no es analizar cuáles son los parámetros más óptimos para los cultivos de lechugas hidropónicas, pero sí crear las herramientas necesarias para hacerlo. Por esa razón, se implementa la posibilidad de seleccionar una longitud de onda diferente a la correspondiente al color blanco.

5.7. CIERRE DE SESIÓN

Para cerrar la sesión, basta con hacer clic en el botón Cerrar sesión, de la barra lateral de navegación (véase Fig. 5-15). Hecho esto, el sitio web redirigirá a la página de inicio de sesión.

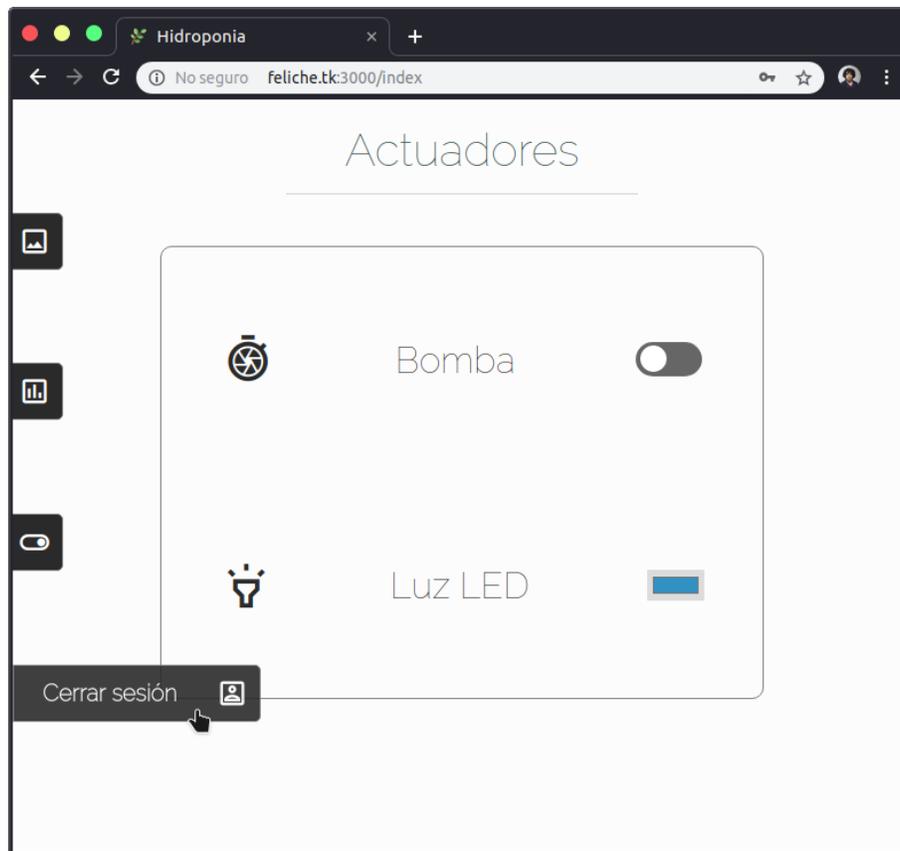


Fig. 5-15. Cerrando sesión a través de la barra lateral de navegación.

CONCLUSIONES Y TRABAJOS FUTUROS

Se logró diseñar y construir un sistema para el monitoreo y control remoto, vía internet, de cultivos hidropónicos. Este sistema cuenta con dos microcontroladores que se encargan de recabar y enviar a la base de datos, información sobre las variables físicas del sistema, a través de los sensores; recibir los comandos para modificar el estado de los actuadores desde la página web; modificar dicho estado de los actuadores; controlar la cámara web para tomar nuevas fotografías; enviar esas mismas fotografías a la base de datos cuando el usuario lo solicita desde la página web. Cuenta con un servidor web, montado en una Raspberry Pi, en el que se encuentra la base de datos, un bróker MQTT y los servicios, escritos en NodeJS, que hacen que funcione el sitio web diseñado. También cuenta con una plataforma web desde donde el usuario puede ver la información sobre el sistema hidropónico, así como solicitar imágenes nuevas y modificar el estado de los actuadores.

Los actuadores que se utilizan en el sistema son una tira de LED RGB y una bomba sumergible de agua. Los sensores empleados (y en algunos casos, contruidos) son para medir la temperatura, la humedad, la iluminación, el pH y la concentración de nutrientes en la solución hidropónica que se usará en el sistema de cultivos hidropónicos NFT. Tanto los actuadores como los sensores se conectan a uno de los dos microcontroladores utilizados.

La cámara web empleada es una cámara serial, que se conecta al Microcontrolador 2.

Se desarrolló una plataforma web de diseño simple, minimalista e intuitivo, que consta de dos páginas, que requiere de autenticación para su uso. Desde ella, se muestra el estado de los actuadores, la información de los sensores a través de gráficos y las imágenes tomadas por el microcontrolador con la cámara web.

El alcance del proyecto fue construir un sistema de semiautomatización para cultivos hidropónicos. No obstante, lo consecuente sería utilizarlo con plántulas de algún cultivo para controlar y medir su desarrollo en el sistema hidropónico. El proyecto podría, así mismo, utilizarse para realizar investigación en el área biológica o de ingeniería agrícola donde se determine la configuración óptima de iluminación artificial, efectos de distintas longitudes de onda en los cultivos, periodos de riego, temperatura, entre otros.

Para mejorar el proyecto actual, podría utilizarse un microcontrolador más potente que no tenga las limitaciones del empleado en este. Un microcontrolador con más prestaciones permitiría añadirle más sensores, de las mismas naturalezas o de distintas que los utilizados, así como más actuadores. Podría entonces controlarse la temperatura, la humedad, el pH y la conductividad eléctrica, y podría llevarse su utilización a invernaderos hidropónicos.

Otra mejora que podría hacerse, es utilizar un servidor web con más sistemas conectados. Lo que permitiría integrar el prototipo en sistemas más complejos. Podría también utilizarse un servicio de servidor web en la nube para suplir al servidor web que se implementó en la Raspberry Pi. Esta mejora

generaría un gasto fijo mensual para su utilización, pero ahorraría al usuario los gastos de electricidad o mantenimiento del servidor propio.

Una característica adicional que se podría añadir al sistema, es la posibilidad de tener las configuraciones y parámetros óptimos para distintos tipos de cultivo en la base de datos, como el porcentaje de nutrientes, la iluminación y temperaturas necesarias. Esto permitiría que el usuario final, en la página web, pueda seleccionar el tipo de cultivo que quiere tener en su sistema hidropónico, y el sistema se configuraría automáticamente para mantener las variables físicas del sistema que puede controlar, dentro de los rangos necesarios para la planta en cuestión; así como mostrar advertencias en el caso de que el usuario deba comprobar el sistema si una variable física que no pueda controlarse automáticamente necesita atención.

El diseño actual es modular, lo que permite su escalabilidad. Una etapa posterior del proyecto podría ser contar con más sistemas hidropónicos en los que se tengan cultivos diferentes y conectarlo al mismo servidor web. Podría entonces hacerse que la página web permitiera seleccionar cualquiera de los sistemas hidropónicos disponibles y conectados al servidor web, para monitorearlo o controlarlo de la misma manera en la que se hace para un solo sistema hidropónico.

APÉNDICES

Apéndice 1. HTTP

El protocolo HTTP utiliza cadenas de texto con palabras reservadas en sus transmisiones que permiten a los clientes y servidores comunicarse y realizar diferentes acciones. En este apéndice se muestra una lista de los comandos, encabezados de petición, encabezados de respuesta y códigos de respuesta existentes, del protocolo HTTP.

LISTA DE COMANDOS HTTP

Comando	Descripción
GET	Solicita el recurso ubicado en la URL especificada
HEAD	Solicita el encabezado del recurso ubicado en la URL especificada
POST	Envía datos al programa ubicado en la URL especificada
PUT	Envía datos a la URL especificada
DELETE	Borra el recurso ubicado en la URL especificada

ENCABEZADOS HTTP

Nombre del encabezado	Descripción
Accept	Tipo de contenido aceptado por el navegador (por ejemplo, texto/html)
Accept-Charset	Juego de caracteres que el navegador espera
Accept-Encoding	Codificación de datos que el navegador acepta
Accept-Language	Idioma que el navegador espera (de forma predeterminada, inglés)
Authorization	Identificación del navegador en el servidor
Content-Encoding	Tipo de codificación para el cuerpo de la solicitud
Content-Language	Tipo de idioma en el cuerpo de la solicitud
Content-Length	Extensión del cuerpo de la solicitud
Content-Type	Tipo de contenido del cuerpo de la solicitud (por ejemplo, texto/html)
Date	Fecha en que comienza la transferencia de datos
Forwarded	Utilizado por equipos intermediarios entre el navegador y el servidor
From	Permite especificar la dirección de correo electrónico del cliente
From	Permite especificar que debe enviarse el documento si ha sido modificado desde una fecha en particular
Link	Vínculo entre dos direcciones URL
Orig-URL	Dirección URL donde se originó la solicitud
Referer	Dirección URL desde la cual se realizó la solicitud
User-Agent	Cadena con información sobre el cliente, por ejemplo, el nombre y la versión del navegador y el sistema operativo

ENCABEZADOS DE RESPUESTA HTTP

Nombre del encabezado	Descripción
Content-Encoding	Tipo de codificación para el cuerpo de la respuesta
Content-Language	Tipo de idioma en el cuerpo de la respuesta
Content-Length	Extensión del cuerpo de la respuesta
Content-Type	Tipo de contenido del cuerpo de la respuesta (por ejemplo, texto/html)
Date	Fecha en que comienza la transferencia de datos
Expires	Fecha límite de uso de los datos
Forwarded	Utilizado por equipos intermediarios entre el navegador y el servidor
Location	Redireccionamiento a una nueva dirección URL asociada con el documento
Server	Características del servidor que envió la respuesta

CÓDIGOS DE RESPUESTA HTTP

Código	Mensaje	Descripción
10x	Mensaje de información	Estos códigos no se utilizan en la versión 1.0 del protocolo
20x	Éxito	Estos códigos indican la correcta ejecución de la transacción
200	OK	La solicitud se llevó a cabo de manera correcta
201	CREATED	Sigue a un comando POST e indica el éxito, la parte restante del cuerpo indica la dirección URL donde se ubicará el documento creado recientemente.
202	ACCEPTED	La solicitud ha sido aceptada, pero el procedimiento que sigue no se ha llevado a cabo
203	PARTIAL INFORMATION	Cuando se recibe este código en respuesta a un comando de GET indica que la respuesta no está completa.
204	NO RESPONSE	El servidor ha recibido la solicitud, pero no hay información de respuesta
205	RESET CONTENT	El servidor le indica al navegador que borre el contenido en los campos de un formulario
206	PARTIAL CONTENT	Es una respuesta a una solicitud que consiste en el encabezado <i>range</i> . El servidor debe indicar el encabezado <i>content-Range</i>
30x	Redirección	Estos códigos indican que el recurso ya no se encuentra en la ubicación especificada
301	MOVED	Los datos solicitados han sido transferidos a una nueva dirección
302	FOUND	Los datos solicitados se encuentran en una nueva dirección URL, pero, no obstante, pueden haber sido trasladados
303	METHOD	Significa que el cliente debe intentarlo con una nueva dirección; es preferible que intente con otro método en vez de GET
304	NOT MODIFIED	Si el cliente llevó a cabo un comando GET condicional (con la solicitud relativa a si el documento ha sido modificado desde la

		última vez) y el documento no ha sido modificado, este código se envía como respuesta.
40x	Error debido al cliente	Estos códigos indican que la solicitud es incorrecta
400	BAD REQUEST	La sintaxis de la solicitud se encuentra formulada de manera errónea o es imposible de responder
401	UNAUTHORIZED	Los parámetros del mensaje aportan las especificaciones de formularios de autorización que se admiten. El cliente debe reformular la solicitud con los datos de autorización correctos
402	PAYMENT REQUIRED	El cliente debe reformular la solicitud con los datos de pago correctos
403	FORBIDDEN	El acceso al recurso simplemente se deniega
404	NOT FOUND	Un clásico. El servidor no halló nada en la dirección especificada. Se ha abandonado sin dejar una dirección para redireccionar...
50x	Error debido al servidor	Estos códigos indican que existe un error interno en el servidor
500	INTERNAL ERROR	El servidor encontró una condición inesperada que le impide seguir con la solicitud (una de esas cosas que les suceden a los servidores...)
501	NOT IMPLEMENTED	El servidor no admite el servicio solicitado (no puede saberlo todo...)
502	BAD GATEWAY	El servidor que actúa como una puerta de enlace o proxy ha recibido una respuesta no válida del servidor al que intenta acceder
503	SERVICE UNAVAILABLE	El servidor no puede responder en ese momento debido a que se encuentra congestionado (todas las líneas de comunicación se encuentran congestionadas, inténtelo de nuevo más adelante)
504	GATEWAY TIMEOUT	La respuesta del servidor ha llevado demasiado tiempo en relación al tiempo de espera que la puerta de enlace podía admitir (excedió el tiempo asignado...)

Apéndice 2. ESPECIFICACIONES DEL ESP8266EX

En este apéndice se muestra una tabla con la información detallada de las características del SoC ESP8266EX: las de su módulo WiFi, de sus partes físicas y las del software.

Categories	Items	Parameters
Wi-Fi	Certification	Wi-Fi Alliance
	Protocols	802.11 b/g/n
	Frequency Range	2.4G – 2.5G (2400M – 2483.5M)
	Tx Power	802.11 b: +20 dBm
		802.11 g: +17 dBm
		802.11 n: +14 dBm
	Rx Sensitivity	802.11 b: -91 dbm (11 Mbps)
802.11 g: -75 dbm (54 Mbps)		
802.11 n: -72 dbm (MCS7)		
Antenna	PCB Trace, External, IPEX Connector, Ceramic Chip	
Hardware	CPU	Tensilica L106 32-bit processor
	Peripheral Interface	UART/SDIO/SPI/I2C/I2S/IR Remote Control
		GPIO/ADC/PWM/LED Light & Button
	Operating Voltage	2.5V – 3.6V
	Operating Current	Average value: 80 mA
	Operating Temperature Range	-40°C – 125°C
	Storage Temperature Range	-40°C – 125°C
	Package Size	QFN32-pin (5 mm x 5 mm)
External Interface	-	
Software	Wi-Fi Mode	Station/SoftAP/SoftAP+Station
	Security	WPA/WPA2
	Encryption	WEP/TKIP/AES
	Firmware Upgrade	UART Download / OTA (via network)
	Software Development	Supports Cloud Server Development / Firmware and SDK for fast on-chip programming
	Network Protocols	IPv4, TCP/UDP/HTTP/FTP
	User Configuration	AT Instruction Set, Cloud Server, Android/iOS App

Apéndice 3. DIAGRAMAS DE PINES DEL NODEMCU V2 Y V3

En este apéndice se incluyen los diagramas de pines de la tarjeta de desarrollo NodeMCU v2 y v3. Dichos diagramas, muestran las funciones que tiene disponible cada uno de los pines de esos dos modelos de tarjeta de desarrollo.

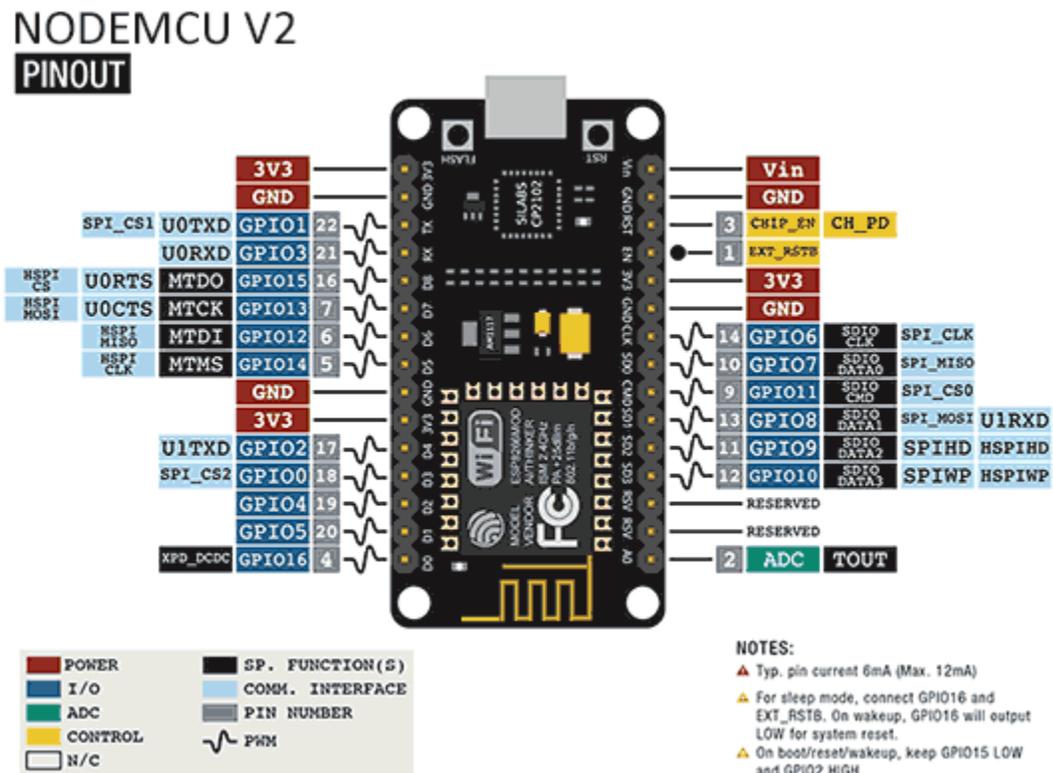


Fig. 0-1. Diagrama de pines del NodeMCU V2 [76].

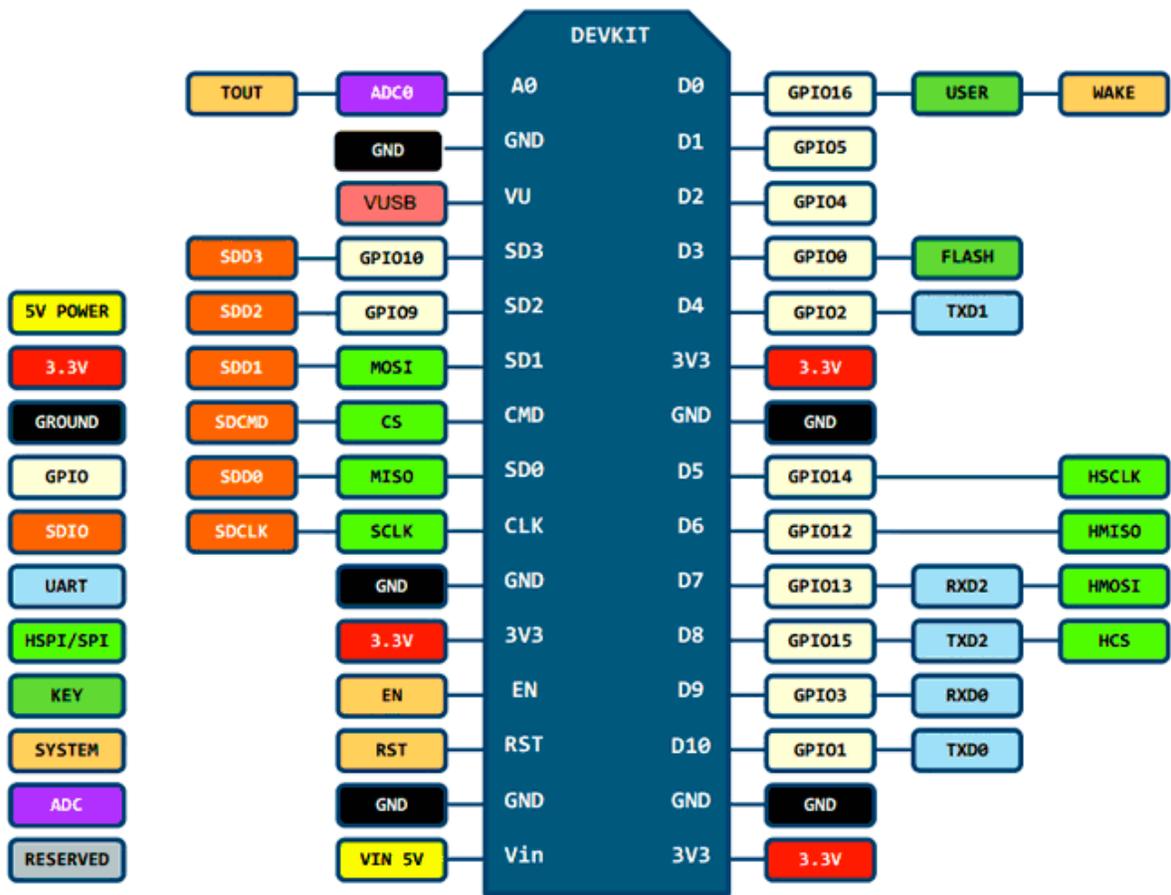


Fig. 0-2. Diagrama de pines del NodeMCU V3 [76].

REFERENCIAS

- [1] National Geographic, «Genographic Project,» National Geographic, [En línea]. Available: <https://genographic.nationalgeographic.com/development-of-agriculture/>. [Último acceso: 13 Mayo 2018].
- [2] A. Carretón, «Arqueoblog,» [En línea]. Available: <http://arqueoblog.com/que-es-el-neolitico/>. [Último acceso: 23 Mayo 2018].
- [3] «Breve historia de la agricultura,» Monsanto Company, [En línea]. Available: <http://www.monsantoglobal.com/global/lan/mejorar-la-agricultura/Pages/breve-historia-de-la-agricultura.aspx>. [Último acceso: 7 Mayo 2018].
- [4] T. Martínez Saldaña, «Historia de la agricultura en México,» de *III Taller Latinoamericano "Prevención de Riesgos en el Uso de Plaguicidas"*, Xalapa, Veracruz, México, 1983.
- [5] M. León-Portilla, «La conquista».
- [6] «Horse-hoeing husbandry,» 1762.
- [7] «Front page of The Abilene reflector,» Kansas, 1884.
- [8] ASALE, RAE, «Diccionario de la lengua española - Edición del Tricentenario,» [En línea]. Available: <http://dle.rae.es/?id=M3AhIMC>. [Último acceso: 8 Mayo 2018].
- [9] J. C. López Hernández y J. Pérez-Parra, «Evolución de las estructuras de invernadero,» Estación Experimental de la Fundación Cajamar "Las Palmerillas", [En línea]. Available: <http://www.publicacionescajamar.es/pdf/series-tematicas/centros-experimentales-las-palmerillas/evolucion-de-las-estructuras.pdf>. [Último acceso: 8 Mayo 2018].
- [10] Diliff. [En línea]. Available: <https://commons.wikimedia.org/w/index.php?curid=7392055>. [Último acceso: 8 Mayo 2018].
- [11] C. von Zabeltitz, «Greenhouse structures,» de *Ecosystems of the world, Greenhouse Ecosystems*, Ed Elsevier, pp. 1-15.
- [12] N. Castilla, *Invernaderos de plástico tecnología y manejo*, Segunda ed., Madrid, Barcelona: Ediciones Mundi-Prensa, 2007.
- [13] Productos Químicos Andinos S. A., «Piña Buga,» Parque Industrial Juanchito.
- [14] E. Rizo, «Inforural,» [En línea]. Available: <https://www.inforural.com.mx/agricultura-protégida-evolucion-y-exito-en-mexico/>. [Último acceso: 8 Mayo 2018].
- [15] Replantable, «Replantable,» [En línea]. Available: <https://replantable.com/>. [Último acceso: 10 Mayo 2018].
- [16] Smartcultiva Corp, «Smartcultiva Advance Microfarming Sensing Technologies,» [En línea]. Available: <http://smartcultiva.com/index.html>. [Último acceso: 10 Mayo 2018].
- [17] WWAP (Programa Mundial de las Naciones Unidas de Evaluación de los Recursos Hídricos)/ONU-Agua, «Informe Mundial de las Naciones Unidas sobre el Desarrollo de los Recursos Hídricos 2018: Soluciones basadas en la naturaleza para la gestión del agua,» UNESCO, París, 2018.
- [18] L. Farràs Pérez, «La Vanguardia,» 21 Mayo 2018. [En línea]. Available: <http://www.lavanguardia.com/natural/20180520/443665527312/agua-sequia-ciudad-del-cabo-cambio-climatico.html>. [Último acceso: 23 Mayo 2018].
- [19] Ecoosfera, «La Agricultura Biodinámica Una Técnica De Cultivo Que Beneficia La Tierra,» [En línea]. Available: <https://ecoosfera.com/2015/12/la-agricultura-biodinamica-una-tecnica-de-cultivo-que-beneficia-la-tierra/>. [Último acceso: 9 Mayo 2018].

- [20] INTAGRI, «Acuaponía para la Producción de Plantas y Peces,» *Serie Horticultura Protegida No. 32*, p. 6, 2017.
- [21] Hidroponia.mx, «Hidroponia.mx,» [En línea]. Available: <http://hidroponia.mx/aeroponia-una-tecnica-de-otro-nivel/>. [Último acceso: 9 Mayo 2018].
- [22] Asociación Hidropónica Mexicana A.C., [En línea]. Available: <http://hidroponia.org.mx/cultivo-hidroponico/que-es-la-hidroponia/>. [Último acceso: 8 Mayo 2018].
- [23] Comercializadora Hydro Environment S.A. de C.V., «HydroEnvironment,» [En línea]. Available: https://hydroenv.com.mx/catalogo/index.php?main_page=page&id=27. [Último acceso: 8 Mayo 2018].
- [24] M. van Heemskerck, Artist, *Hanging Gardens of Babylon*. [Art].
- [25] electronicaestudio.com, «¿Qué es un microcontrolador?,» electronicaestudio.com, [En línea]. Available: <http://www.electronicaestudio.com/microcontrolador.htm>. [Último acceso: 10 Junio 2018].
- [26] MCI Electronics, «¿Qué es un microcontrolador?,» MCI Electronics, [En línea]. Available: <http://cursos.mcielectronics.cl/que-es-un-microcontrolador-1/>. [Último acceso: 10 Junio 2018].
- [27] Solvetic Sistemas, «Solvetic,» Solvetic, 16 Junio 2017. [En línea]. Available: <https://www.solvetic.com/page/recopilaciones/s/profesionales/que-es-soc-system-on-a-chip-caracteristicas>. [Último acceso: 1 Octubre 2019].
- [28] L. del Valle Hernandez, «NodeMCU tutorial paso a paso desde cero,» Programarfácil.com, [En línea]. Available: <https://programarfácil.com/podcast/nodemcu-tutorial-paso-a-paso/>. [Último acceso: 17 Junio 2018].
- [29] J. G. Cobo, «Hardware Libre,» 3 Noviembre 2014. [En línea]. Available: <https://www.hwlibre.com/que-es-una-placa-sbc/>. [Último acceso: 20 Mayo 2019].
- [30] G. Halfacree, «Wikipedia,» 14 Marzo 2018. [En línea]. Available: [https://es.wikipedia.org/wiki/Archivo:Raspberry_Pi_3_B%2B_\(39906369025\).png#metad_ata](https://es.wikipedia.org/wiki/Archivo:Raspberry_Pi_3_B%2B_(39906369025).png#metad_ata). [Último acceso: 20 Mayo 2019].
- [31] ControlReal, «Sensores. Definición y características,» ControlReal, 18 Agosto 2015. [En línea]. Available: <http://controlreal.com/es/sensores-definicion-y-caracteristicas/>. [Último acceso: 10 Junio 2018].
- [32] Omega, «Sensor de temperatura,» Omega, [En línea]. Available: <https://es.omega.com/prodinfo/medicion-temperatura.html#>. [Último acceso: 10 Junio 2018].
- [33] Motorlan, «Sondas térmicas en motores eléctricos: PTC, NTC, RTD... ¿Qué las diferencia?,» Motorlan, [En línea]. Available: <http://www.motorlan.es/es/sondas-termicas-motores-electricos-ptc-ntc-rtd-las-diferencia/>. [Último acceso: 16 Junio 2018].
- [34] Omega, «Introducción a los termómetros infrarrojos,» Omega, [En línea]. Available: <https://mx.omega.com/prodinfo/termometro-infrarrojo.html>. [Último acceso: 10 Junio 2018].
- [35] Omega, «Medidores de pH,» Omegas, [En línea]. Available: <https://mx.omega.com/prodinfo/medidores-de-ph.html>. [Último acceso: 10 Junio 2018].
- [36] Venemedia, «Definición de pH,» Condeptdefinición.de, 23 Septiembre 2016. [En línea]. Available: <http://conceptdefinicion.de/ph/>. [Último acceso: 10 Junio 2018].

- [37] Infoagro, «El Conductividad Eléctrica (CE),» Infoagro Systems, [En línea]. Available: http://www.infoagro.com/instrumentos_medida/doc_conductividad_electrica.asp?k=53. [Último acceso: 10 Junio 2018].
- [38] Zelian, «Zelian - Equipamiento de laboratorio,» Zelian.com.ar, [En línea]. Available: <https://www.zelian.com.ar/conductivimetro-de-sobremesa-marca-hanna-modelo-edge-hi2003--det--PCO-00484>. [Último acceso: 10 Junio 2018].
- [39] E. Mandado Pérez, J. Marcos Acevedo, C. Fernández Silva y J. I. Armesto Quiroga, Autómatas programables y sistemas de automatización, Barcelona: marcombo, 2009.
- [40] Robot Diyari, «Robot Diyari,» [En línea]. Available: <https://robotdiyari.com/image/cache/catalog/URUNLER/LDR-800x800-1600x1600.jpg>. [Último acceso: 16 Mayo 2019].
- [41] Ecured, «Sensor de humedad,» Ecured.cu, [En línea]. Available: https://www.ecured.cu/Sensor_de_Humedad. [Último acceso: 10 Junio 2018].
- [42] Talos Electronics, «Talos Electronics,» [En línea]. Available: <https://www.taloselectronics.com/products/sensor-de-temperatura-y-humedad-analogico-amt1001>. [Último acceso: 16 Mayo 2019].
- [43] GRobotronics, «GRobotronics - The Ultimate Makerstore,» [En línea]. Available: <https://grobotronics.com/rht03-dht22.html>. [Último acceso: 16 Mayo 2019].
- [44] Definiciona, «Definiciona,» 2019. [En línea]. Available: <https://definiciona.com/psicrometria/>. [Último acceso: 2019 Octubre 1].
- [45] 330ohms, «¿Qué son los actuadores electrónicos?,» 330ohms, 14 Octubre 2013. [En línea]. Available: <http://blog.330ohms.com/2013/10/14/que-son-los-actuadores-electronicos/>. [Último acceso: 10 Junio 2018].
- [46] Wikipedia, «Actuador,» Wikipedia, 7 Junio 2018. [En línea]. Available: https://es.wikipedia.org/wiki/Actuador#Actuadores_neum%C3%A1ticos. [Último acceso: 10 Junio 2018].
- [47] Unet.edu.ve, «Marco teórico,» Unet.edu.ve, [En línea]. Available: <http://www.unet.edu.ve/~maqflu/doc/LAB-1-95.htm>. [Último acceso: 10 Junio 2018].
- [48] Directores Industriales, «Automatización y Electrónica,» Directores Industriales, 2 Julio 2014. [En línea]. Available: https://www.dirind.com/dae/monografia.php?cla_id=18. [Último acceso: 10 Junio 2018].
- [49] ilutop, «ilutop,» 11 Julio 2019. [En línea]. Available: <https://www.ilutop.com/blog/lamparas-de-diseno-famosas/>. [Último acceso: 1 Octubre 2019].
- [50] Cefire, «3.1 Tipos de lámparas,» cefire.edu.gva.es, [En línea]. Available: http://cefire.edu.gva.es/pluginfile.php/199806/mod_resource/content/0/contenidos/009/luminotecnia/31_tipos_de_lmparas.html. [Último acceso: 10 Junio 2018].
- [51] Definicion.mx, «Definición de Dispositivos Periféricos,» Definicion.mx, [En línea]. Available: <https://definicion.mx/dispositivos-perifericos/>. [Último acceso: 10 Junio 2018].
- [52] Wikipedia, «Cámara web,» Wikipedia, 25 Abril 2018. [En línea]. Available: https://es.wikipedia.org/wiki/C%C3%A1mara_web. [Último acceso: 10 Junio 2018].
- [53] J. Angeles Cruz, «Jcacweb cam,» 2008.
- [54] C. Vialfa, «El concepto de red,» CCM, 18 Enero 2018. [En línea]. Available: <https://es.ccm.net/contents/252-el-concepto-de-red>. [Último acceso: 29 Mayo 2018].
- [55] C. Vialfa, «LAN (red de área local),» CCM, 8 Marzo 2017. [En línea]. Available: <https://es.ccm.net/contents/253-lan-red-de-area-local>. [Último acceso: 29 Mayo 2018].

- [56] C. Vialfa, «Redes de área local,» CCM, 26 Septiembre 2017. [En línea]. Available: https://es.ccm.net/contents/295-redes-de-area-local#simili_main. [Último acceso: 29 Mayo 2018].
- [57] C. Vialfa, «VLAN - Redes virtuales,» CCM, 13 Septiembre 2017. [En línea]. Available: <https://es.ccm.net/contents/286-vlan-redes-virtuales>. [Último acceso: 29 Mayo 2018].
- [58] Jeff, «Internet,» CCM, 16 Octubre 2008. [En línea]. Available: <https://es.ccm.net/contents/266-internet>. [Último acceso: 29 Mayo 2018].
- [59] C. Vialfa, «Módem,» CCM, 18 Octubre 2016. [En línea]. Available: <https://es.ccm.net/contents/675-modem>. [Último acceso: 29 Mayo 2018].
- [60] ADSLZONE, «Router : Toda la información,» ADSLZONE, [En línea]. Available: https://www.adslzone.net/adsl_router-faq.html. [Último acceso: 29 Mayo 2018].
- [61] Cadena SER, «Cadena SER,» [En línea]. Available: https://cadenaser00.epimg.net/ser/imagenes/2018/11/07/radio_elda/1541609655_699980_1541611194_noticia_normal_recorte1.jpg. [Último acceso: 16 Mayo 2019].
- [62] C. Vialfa, «Protocolo de comunicación,» CCM, 17 Enero 2018. [En línea]. Available: <https://es.ccm.net/contents/275-protocolo-de-comunicacion>. [Último acceso: 23 Mayo 2018].
- [63] C. Vialfa, «El protocolo HTTP,» CCM, 17 Enero 2018. [En línea]. Available: <https://es.ccm.net/contents/264-el-protocolo-http>. [Último acceso: 23 Mayo 2018].
- [64] C. Vialfa, «El protocolo IP,» CCM, 19 Octubre 2017. [En línea]. Available: <https://es.ccm.net/contents/274-el-protocolo-ip#q=protocolo+ip&cur=1&url=%2F>. [Último acceso: 27 Mayo 2018].
- [65] Jeff, «Protocolo IPv6,» CCM, 16 Octubre 2008. [En línea]. Available: <https://es.ccm.net/contents/268-protocolo-ipv6>. [Último acceso: 27 Mayo 2018].
- [66] C. Vialfa, «Protocolo TCP,» CCM, 10 Julio 2017. [En línea]. Available: <https://es.ccm.net/contents/281-protocolo-tcp>. [Último acceso: 28 Mayo 2018].
- [67] LCC, «Herramientas WEB para la enseñanza de Protocolos de Comunicación,» LCC, [En línea]. Available: <http://neo.lcc.uma.es/evirtual/cdd/tutorial/aplicacion/cliente-servidor.html>. [Último acceso: 29 Mayo 2018].
- [68] T. Ylonen, «SSH - Secure Login Connections over the Internet,» de *Proceedings of the 6th USENIX Security Symposium*, 1996.
- [69] Geeky Theory , «Geeky Theory,» [En línea]. Available: <https://geekytheory.com/que-es-mqtt>. [Último acceso: 29 Mayo 2019].
- [70] M. Marqués, «UNIVERSITAT JAUME I DE CASTELLÓ,» Departamento de Ingeniería y Ciencia de la Computación, Enero 2009. [En línea]. Available: http://www3.uji.es/~mmarques/apuntes_bbdd/apuntes.pdf. [Último acceso: 30 Mayo 2019].
- [71] A. Pat, «SQL vs NOSQL. Conoce sus diferencias y cuando utilizarlas,» Mexlike, 22 Mayo 2018. [En línea]. Available: <https://mexlike.io/sql-vs-nosql-conoce-sus-diferencias-de-cada-tipo/>. [Último acceso: 30 Mayo 2019].
- [72] acenswhitepapers, «Bases de datos NoSQL. Qué son y tipos que nos podemos encontrar,» acenswhitepapers, [En línea]. Available: <https://www.acens.com/wp-content/images/2014/02/bbdd-nosql-wp-acens.pdf>. [Último acceso: 30 Mayo 2019].
- [73] Mercadolibre.com, «Solución Nutritiva Sales Hidropónica 500 L Envío Gratis,» Mercadolibre.com.mx, [En línea]. Available: <https://articulo.mercadolibre.com.mx/MLM->

- 589522376-solucion-nutritiva-sales-hidroponica-500-l-envio-gratis-_JM?searchVariation=20873510665. [Último acceso: 17 Junio 2018].
- [74] Hortaflor, *Sobre de 1g de Semillas de Lechuga Italiana Black Seeded Simpson*, Francia, 2018.
- [75] Espressif Inc., «ESP8266EX Datasheet,» Espressif Inc., 2018.
- [76] L. Llamas, «NodeMCU, la popular placa de desarrollo con ESP8266,» luisllamas.es, 1 Junio 2018. [En línea]. Available: <https://www.luisllamas.es/esp8266-nodemcu/>. [Último acceso: 18 Junio 2018].
- [77] NodeMCU, «NodeMCU Documentation Building the firmware,» nodemcu.readthedocs.io, [En línea]. Available: <https://nodemcu.readthedocs.io/en/master/en/>. [Último acceso: 19 Junio 2018].
- [78] NodeMCU, «NodeMCU Documentation Flashing the Firmware,» nodemcu.readthedocs.io, [En línea]. Available: <https://nodemcu.readthedocs.io/en/master/en/flash/>. [Último acceso: 19 Junio 2018].
- [79] Sparkfun, «LinkSprite JPEG Color Camera TTL Interface,» Sparkfun.com, [En línea]. Available: <https://www.sparkfun.com/products/retired/10061>. [Último acceso: 15 Junio 2018].
- [80] DiferenciasEntre, «Diferencias entre JPG y JPEG,» diferenciasentre.org, [En línea]. Available: <https://diferenciasentre.org/jpg-y-jpeg/>. [Último acceso: 15 Junio 2018].
- [81] J.-F. Pillou, «CMM,» 16 Octubre 2008. [En línea]. Available: <https://es.ccm.net/contents/729-compresion-jpeg>. [Último acceso: 18 Noviembre 2019].
- [82] LinkSprite Technologies, Inc, «LinkSprite JPEG Color Camera Serial UART Interface User Manual,» 2012.
- [83] AOSONG, «Temperature and humidity module DHT11 Product Manual,» Akizuki Denshi Tsusho Co..
- [84] Green Forest, «Green Forest,» Servicios Forestales S.A.S., [En línea]. Available: <https://greenforest.com.co/producto/medidor-de-conductividad-tdsec-hold/>. [Último acceso: 28 Mayo 2019].
- [85] Diymore, «Diy More,» Diymore, [En línea]. Available: https://www.diymore.cc/products/diymore-liquid-ph-value-detection-detect-sensor-module-monitoring-control-for-arduino-m?_pos=1&_sid=75be5e74b&_ss=r. [Último acceso: 25 Julio 2019].
- [86] Diymore, «Diy More,» Diymore, [En línea]. Available: <https://www.diymore.cc/collections/free-shipping-1/products/bnc-electrode-probe-connector-hydroponic-for-ph-aquarium-controller-meter-sensor>. [Último acceso: 25 Julio 2019].
- [87] Amazon.com, «Yosoo Solar DC 12 V 24 V agua caliente bomba de circulación bomba de agua Motor sin escobillas 3 m 5 m de poco ruido csf (DC 24 V),» Amazon.com, [En línea]. Available: https://www.amazon.com.mx/Yosoo-Solar-caliente-circulaci%C3%B3n-escobillas/dp/B00DWORE6E/ref=sr_1_1?ie=UTF8&qid=1529261077&sr=8-1&keywords=bomba+de+agua+12v. [Último acceso: 17 Junio 2018].
- [88] laser&led, «Tira de LEDs exterior RGB,» [En línea]. Available: https://tienda.laserandled.com/fichas_tecnicas_laser_and_led/TMU5SOLA.pdf. [Último acceso: 17 Junio 2018].
- [89] Raspberry Pi Foundation, «Raspberry Pi,» [En línea]. Available: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>. [Último acceso: 12 Junio 2019].

- [90] Fairchild Semiconductor, «CD4051BC • CD4052BC • CD4053BC,» Noviembre 1983. [En línea]. Available: <https://www.mouser.mx/datasheet/2/308/4051BC-1190287.pdf>. [Último acceso: 11 Junio 2019].
- [91] ON Semiconductor, «BC546B, BC547A, B, C,,» [En línea]. Available: <https://www.mouser.mx/datasheet/2/308/bc546-d-1190617.pdf>. [Último acceso: 11 Junio 2019].
- [92] STMicroelectronics, «L293B,» [En línea]. Available: <https://www.st.com/resource/en/datasheet/l293b.pdf>. [Último acceso: 31 Julio 2019].
- [93] Node.js Foundation, «NodeJS,» [En línea]. Available: <https://nodejs.org/es/about/>. [Último acceso: 12 Junio 2019].
- [94] A. G. Benites, «DevCode,» [En línea]. Available: <https://devcode.la/blog/que-es-npm/>. [Último acceso: 12 Junio 2019].
- [95] Boboss74, «Vector logo of npm,» 4 Septiembre 2014. [En línea]. Available: <https://commons.wikimedia.org/wiki/File:Npm-logo.svg>. [Último acceso: 19 Septiembre 2019].
- [96] MongoDB, Inc., «mongo DB,» [En línea]. Available: <https://www.mongodb.com/es/what-is-mongodb>. [Último acceso: 12 Junio 2019].
- [97] NodeMCU, «NodeMCU Documentation ESPlorer,» nodemcu.readthedocs.io, [En línea]. Available: <https://nodemcu.readthedocs.io/en/master/en/upload/>. [Último acceso: 19 Junio 2018].
- [98] NodeMCU, «NodeMCU Documentation Uploading Code,» nodemcu.readthedocs.io, [En línea]. Available: <https://nodemcu.readthedocs.io/en/master/en/upload/>. [Último acceso: 19 Junio 2018].
- [99] W. Vargas, «Lua, el lenguaje de programación creado en Brasil,» GCF AprendeLibre, 26 Agosto 2015. [En línea]. Available: https://www.gcfaprendelibre.org/blog/lua_el_lenguaje_de_programacion_creado_en_brasil/1.do. [Último acceso: 19 Junio 2018].
- [100] T. P. Stephen Robinson, «NodeMCU Documentation,» 23 Enero 2015. [En línea]. Available: <https://nodemcu.readthedocs.io/en/master/modules/mqtt/>. [Último acceso: 30 Mayo 2019].
- [101] Cyril, «foobarflies,» foobarflies.io, 24 Febrero 2016. [En línea]. Available: <https://www.foobarflies.io/reassessing-lua-on-nodemcu/>. [Último acceso: 17 Mayo 2019].
- [102] T. e. a. Han, «Improving “color rendering” of LED lighting for the growth of lettuce,» 2017.
- [103] K.-H. L. e. al., «The effects of red, blue, and white light-emitting diodes on the growth development, and edible quality of hydroponically grown lettuce (*Lactuca sativa* L. var. capitata),» *Scientia Horticulturae*, 2012.
- [104] C. Vialfa, «MIME (Extensiones Multipropósito de Correo de Internet),» CCM, 9 Enero 2018. [En línea]. Available: <https://es.ccm.net/contents/118-mime-extensiones-multiproposito-de-correo-de-internet#q=mime&cur=1&url=%2F>. [Último acceso: 23 Mayo 2018].
- [105] Nvidia, «Nvidia,» 13 Mayo 2008. [En línea]. Available: https://www.nvidia.com/docs/CP/45121/nforce_680i_sli_overclocking.pdf. [Último acceso: 28 Mayo 2019].

[106] B. L. Takeyas, «Funciones Hash o Hashing,» [En línea]. Available: http://www.itnuevolaredo.edu.mx/takeyas/Apuntes/Administracion_Archivos/Apuntes/Hashing.PDF. [Último acceso: 27 Mayo 2019].