



“Responsabilidad con pensamiento positivo”

UNIVERSIDAD TECNOLÓGICA ISRAEL

TRABAJO DE TITULACIÓN EN OPCIÓN AL GRADO DE:
INGENIERO EN ELECTRÓNICA DIGITAL Y
TELECOMUNICACIONES

TEMA:

AUTOMATIZACIÓN DE UN CONTENEDOR DE CARGA PARA EL
CONTROL DE CLIMA E ILUMINACIÓN DE CULTIVOS
HIDROPÓNICOS.

AUTOR:

FRANKLIN PATRICIO MONTAHUANO PACHECO

TUTOR:

Ing. Luis H. Montoya L., Msc.

QUITO, ECUADOR

AÑO: 2018

UNIVERSIDAD TECNOLÓGICA ISRAEL

APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de titulación certifico:

Que el trabajo de titulación “AUTOMATIZACIÓN DE UN CONTENEDOR DE CARGA PARA EL CONTROL DE CLIMA E ILUMINACIÓN DE CULTIVOS HIDROPÓNICOS”, presentado por el Señor Franklin Patricio Montahuano Pacheco, estudiante de la carrera de Electrónica Digital y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito D.M. 16 de Agosto del 2018

TUTOR

Ing. Luis H. Montoya L., Msc.

AGRADECIMIENTOS:

Deseo que estas palabras que sirvan para mostrar el más profundo agradecimiento que siento hacia Dios, por guiarme toda la vida y traerme hasta este momento.

También, quiero agradecer con todo mi corazón a José y Luz, quienes siempre han estado a mi lado en la felicidad y en la tristeza, a ellos a los que con mucho orgullo los llamo Papá y Mamá.

Tampoco voy a olvidar a toda mi familia, a mis hermanos Luz y Jorge, así como también a mi sobrino José David por aguantarme tanto en las noches de desvelo, mi malgenio, por brindarme siempre su apoyo y entenderme a pesar de todo.

A mi novia Tania por ser durante mucho tiempo una pieza clave de apoyo, motivación en el estudio y superación personal.

A mis 3 mejores amigos Santiago B., Santiago B, y Luis que Dios los puso en mi camino y planeo conservarlos para el resto de la vida.

A mis compañeros y amigos José, Cristian, Iván y Mauricio por ser parte de este reto, que emprendimos hace casi 4 años y también por servirme de apoyo cuando los necesite.

A Todos ustedes muchas gracias desde el fondo de mi corazón.

Patricio

DEDICATORIA:

Al principio me resultó un poco difícil, como suele suceder cuando realizas la mayoría de las cosas por primera vez, pero después de reflexionar, comprendí, que las personas a las que debo agradecer este trabajo son las que siempre estuvieron ahí y las llamo con mucho cariño y amor “mi familia”.

También quiero dedicar este trabajo a Dios, porque siempre me ha brindado la fortaleza y sabiduría, se lo dedico a todos y cada uno de los miembros de mi familia, que pese a todas las dificultades siempre han estado a mi lado, apoyándome y dándome la fuerza y valor para seguir adelante.

A mis padres, José y Luz que siempre me enseñaron e inculcaron principios morales, religiosos y hábitos de estudio.

A mis hermanos, Jorge y Luz que han sido unas personas muy importantes en mi vida siempre motivándome como estudiante y persona, que siempre pasaron pendientes en el transcurso de mi carrera.

A mi sobrino José David por ser mi mayor motivación para nunca rendirme en los estudios y así poder llegar a ser un ejemplo para él.

Patricio

ÍNDICE DE CONTENIDO

Resumen.....	ix
Abstract	x
INTRODUCCIÓN	1
Antecedentes	2
Invernaderos.....	2
Estructuras y control ambiental.....	3
Cultivos en invernadero con suelo	3
Cultivos hidropónicos	4
Formulación	10
Justificación:	11
Alcance:.....	12
OBJETIVOS	13
Objetivo general:.....	13
Objetivos específicos:	13
CAPÍTULO 1	15
FUNDAMENTOS TEÓRICOS	15
1.1. Marco teórico	15
1.2. Microcontrolador.....	17
1.2.1. Arduino Uno Wi-Fi	17
1.3. Dispositivos de entrada	18
Sensor de humedad relativa y temperatura DTH22	18
1.4. Dispositivos Especiales	21
1.4.1. Reloj En tiempo real <i>RTC</i>	21
1.4.2. Modulo I2C	22
1.5. Dispositivos de salida.....	23
CAPÍTULO 2.....	30
DISEÑO DEL PROGRAMA.....	30
2.1 Diagrama de Flujo.....	30

2.2. Máquina de estados finitos.....	32
Capítulo 3	43
Diseño del Hardware.....	43
3.1. Desarrollo	44
CAPÍTULO 4.....	50
PRUEBAS Y ANÁLISIS DE RESULTADOS.....	50
4.1 Pruebas	50
4.2. Análisis de resultados.....	55
CONCLUSIONES	56
RECOMENDACIONES	58
Bibliografía	59
Anexo 1. Código Fuente:	61
Anexo 2. Datasheet ArduinoUno Wifi:.....	62
Anexo 3. Datasheet DTH22 Digital Humidity-Temperature Sensor:	63
Anexo 4. Datasheet Real Time Clock Rtc-Ds3231	64
Anexo 5. Datasheet LCD 20x4 Lcd-020n0041.....	65
Anexo 6. Datasheet I2C-LCD-Interface.....	66

TABLA DE FIGURAS (ILUSTRACIONES):

Figura 1. 1. Diagrama de bloques	16
Figura 1. 2. Arduino Uno Wi-Fi.....	17
Figura 1. 3. Sensor de humedad relativa y temperatura DTH22.....	18
Figura 1. 4. Esquema de conexión DTH22	21
Figura 1. 5. Módulo RTC de Arduino.....	22
Figura 1. 6. Módulo I2C de Arduino.....	22
Figura 1. 7. Distribución espectral de luz de los LED para crecimiento.....	25
Figura 1. 8. Plantas con Iluminación LED para crecimiento	26
Figura 1. 9. Pantalla LCD	26
Figura 1. 10. Módulo de relé +5V.....	28
Figura 1. 11. Relé normalmente abierto (NO).	28
Figura 1. 12. Relé normalmente cerrado (NC).....	29
Figura 2. 13 Diagrama de flujo	31
Figura 2. 14. Sistema de control retroalimentado	32
Figura 2. 15. Definición de botones y pines.....	32
Figura 2. 16. Botones para el control Automático.	33
Figura 2. 17. Definición de los estados	33
Figura 2. 18. Flanco de bajada y subida.....	34
Figura 2. 19. Rutina para controlar el flanco de subida.	34
Figura 2. 20. Pantalla principal de MENU.....	35
Figura 2. 21. Estado inicial de HOME y SET TEMP.	35
Figura 2. 22. Estado inicial SET TEMP.....	36
Figura 2. 23. MENU de la temperatura de operación	36
Figura 2. 24. Estado inicial SET HUM.....	37
Figura 2. 25. MENU de la humedad de operación.....	37
Figura 2. 26. Estado inicial SET ILIM.....	38
Figura 2. 27. MENU del control de la iluminación.....	38
Figura 2. 28. Estado inicial SET TIME.....	39
Figura 2. 29. MENU del control del tiempo de riego.	40
Figura 2. 30. Máquina de estados completa.	41
Figura 2. 31. MENU del control del intervalo de tiempo para riego.	41
Figura 3. 32. Simulación en Proteus.	43
Figura 3. 33. circuito de prueba en protoboard.	44
Figura 3. 34. Cálculo del ancho de banda de una pista PCB.....	45
Figura 3. 35. Diseño de la PCB Fuente:.....	45

Figura 3. 36. PCB (Placa de circuito impreso).....	46
Figura 3. 37. Soldadura de elementos en la PCB (Placa de circuito impreso).....	47
Figura 3. 38. Montaje de elementos en la PCB (Placa de circuito impreso).....	47
Figura 3. 39. Montaje de los pulsadores en la placa acrílica.....	48
Figura 3. 40. Armado del contenedor a escala.	49
Figura 3. 41. Proyecto finalizado.	49
Figura 4. 42. Prueba de funcionamiento del circuito una vez ensamblado.	50
Figura 4. 43. Comparación de valores medidos por el circuito.....	51
Figura 4. 44. Luces LED para crecimiento.	52
Figura 4. 45. Planta creciendo con luz LED.	52
Figura 4. 46. Prueba de funcionamiento de la bomba de agua.....	53
Figura 4. 47. Prueba de funcionamiento del aire acondicionado.	53
Figura 4. 48. Crecimiento de hortalizas en ambiente controlado.....	54
Figura 4. 49. Plantas cultivadas con control automático.....	54

CONTENIDO DE TABLAS

Tabla 1. 1. Características eléctricas del DTH22	19
Tabla 1. 2. Bits de datos del DTH22	20
Tabla 1. 3. Módulos y pines I ² C.....	23
Tabla 1. 4. Contenido espectral de los LED para crecimiento	24
Tabla 1. 5. Especificaciones de los LED's de crecimiento	25
Tabla 1. 6. Pines del LCD 20x4	27

Resumen

El presente proyecto presenta el diseño e implementación de un sistema de control automático y manejo de cultivo hidropónico para la producción de cultivos dentro de contenedores, teniendo en cuenta que los resultados estén dentro del rango de producción esperado.

El sistema de control realiza el monitoreo de los parámetros y permite su visualización en una pantalla LCD. También entrega datos bidireccionales para la comunicación entre el microcontrolador y un computador, dando al usuario la capacidad de controlar de forma automática y manual todo el sistema.

El proceso del diseño del prototipo, construcción, programación y pruebas de laboratorio se detallan a lo largo del presente documento, así como las dificultades y desventajas que se pueden presentar en los cultivos tradicionales, pruebas y comparaciones en cuanto a crecimiento, rendimiento y calidad, finalizando con un detalle de conclusiones y recomendaciones.

Palabras Clave:

Automatización

Wifi

Hidroponía

Clima

Abstract

In this project the design and implementation of a system of automatic control and management of hydroponic crops for the production within containers is presented, taking into account the results within the expected production range.

The control system performs the monitoring of the parameters and allows it to be shown on a LCD screen. It also delivers bidirectional data for communication between the microcontroller and a computer, giving the user the ability to control the entire system both automatically and manually.

The process of prototype design, construction, programming and laboratory tests are detailed throughout the document, as well as the difficulties and disadvantages that may arise compared to traditional crops, tests and comparisons in terms of growth, performance and quality, ending with a detail of conclusions and recommendations.

Keywords:

Automation

Wifi

Hydroponics

Weather

INTRODUCCIÓN

El cultivo de hortalizas en invernaderos es una de las formas más comunes en todas las empresas agrícolas a nivel mundial. En una combinación con los invernaderos, la hidroponía se está volviendo cada vez más popular, especialmente en los países más desarrollados del mundo como Estados Unidos, Canadá, Japón y algunos otros países en Europa. Esta tecnología es altamente productiva, ya que conserva el agua y la tierra lo que genera que se proteja el medio ambiente.

La hidroponía es común para la producción de verduras, hierbas, cultivos verticales (tomate, pepino y pimientos). Mediante la utilización de sistemas de control automático se operan cientos de dispositivos dentro de un invernadero mediante la medición de parámetros de entrada, para mantener el entorno de crecimiento ideal. La tecnología de producción de alimentos actualmente está cambiando rápidamente ya que los nuevos sistemas de control existentes producen rendimientos nunca alcanzados, por lo que se espera que en un futuro muy cercano su utilización sea a escala mundial (Dalrymple DG, 1973).

La tecnología para la producción de alimentos en invernaderos ha avanzado muchísimo en los últimos 30 años. La producción de alimentos en invernadero a menudo se denomina agricultura de ambiente controlado (*Controlled Environment Agriculture*) y por lo general va acompañada con la hidroponía. Los cultivos hidropónicos posiblemente son el método que más demanda tiene en la producción de la industria agrícola actual, ya que requiere habilidades básicas de agricultura.

Hay muchos tipos de sistemas hidropónicos para la producción agrícola en ambientes controlados, estos tienen lugar dentro de recintos diseñados para controlar el aire, temperatura, luz, agua, nutrición de las plantas y clima. Cada componente de la agricultura en ambientes controlados tiene la misma importancia, ya sea el diseño estructural, el control ambiental o el sistema de crecimiento. No todos los sistemas son rentables ya que con bastante frecuencia se utilizan solo una o dos de las principales variables y si no se presta una atención adecuada, ningún sistema hidropónico resultaría económicamente sustentable.

Antecedentes

Invernaderos

Los invernaderos a partir de los años 60 se han convertido en más que un protector de plantas, hoy en día se los pueden ver como una especie de fábricas de vegetales. En casi todos los aspectos de la producción el sistema está 100% automatizado, ya que cuenta con un entorno artificial controlado por una computadora casi en su totalidad.

Cuando el sistema de invernaderos se encuentra en un entorno de investigación en un sistema totalmente cerrado se le denomina cámara de crecimiento o un fitotrón, que es un laboratorio equipado para estudiar las condiciones físicas y químicas en las que se desarrollan las plantas.

La agricultura ambiental controlada ha ganado muchísima importancia en el campo hortícola, no solo por la producción de cultivos vegetales y ornamentales, sino también en la producción de plántulas de plantas, ya sea desde la semilla o mediante procedimientos de cultivo de tejidos. En los últimos 15 años ha habido un interés creciente en el uso de técnicas hidropónicas para producir cultivos hortícolas en invernaderos.

El futuro crecimiento en invernadero mediante la agricultura ambiental controlada generará que la hidroponía se utilice para la producción de hortalizas a gran escala y depende en gran medida del desarrollo de sistemas de producción que sean más competitivos en términos de costos en comparación con la agricultura de campo abierto. La economía en la producción de alimentos en invernaderos es equilibrada en relación con el alto capital y los costos operacionales de los invernaderos, ya que es significativamente mayor la productividad de la agricultura de ambiente controlado en comparación con la agricultura de campo abierto (OFA). (Jensen, 2017)

Estructuras y control ambiental

Para lograr un ambiente de cultivo más uniforme, sin fluctuaciones de temperatura rápidas, se está utilizando un mayor volumen de espacio dentro del área dedicada a un invernadero, donde hoy en día los soportes de las estructuras de invernadero alcanzan una altura de más de 5 m sobre el nivel del suelo. Los materiales más nuevos como los policarbonatos y los acrílicos se han vuelto mucho más comunes, aunque su popularidad aún no ha sido compensada por los altos costos de producción.

Hace algunos años había un gran énfasis en la investigación para el uso de energía solar como fuente de alimentación y calor del invernadero ya que es técnicamente factible, pero se ha demostrado que no es económica debido a los altos costos de recolección y almacenamiento. En los últimos diez años, ha habido un gran interés en el desarrollo de plantas de cogeneración, donde pequeñas plantas eléctricas utilizan el calor residual que desprenden los generadores eléctricos para calentar hortalizas de invernadero o agua para la producción de pescado.

Cualquiera que sea la fuente de energía, es importante conservarla dentro del invernadero. En el futuro los invernaderos tendrán techos retráctiles que permitirán la entrada de luz y ventilación si es deseado mediante sistemas de control automático. (Jensen, 2017).

Cultivos en invernadero con suelo

El método estándar para cultivar vegetales de invernadero en todo el mundo está en el suelo y un agricultor exitoso generalmente tiene un buen conocimiento de horticultura, suelos, plantas, patología, entomología y fisiología vegetal, así como la capacidad de ingeniería para proporcionar el ambiente más adecuado para el crecimiento de la planta.

Muchos agricultores que hacen cultivos en invernadero, la mayoría de las veces fallan porque carecen de la educación y la capacitación adecuada. Además, un problema importante en los cultivos tradicionales son las enfermedades transmitidas por el suelo, plantas creciendo continuamente sin rotación de cultivos o interrupción en la producción.

Actualmente existen muchas restricciones ambientales y de salud por la falta de fungicidas para suelo disponibles para uso en invernaderos. Este problema sumado al alto

costo de esterilizar el ambiente con vapor está creando la necesidad de utilizar métodos alternativos de agricultura como la hidropónica en ambientes controlados. (Dalrymple DG, 1973) (Jensen, 2017).

Cultivos hidropónicos

Durante los últimos años, ha habido un interés creciente en la hidroponía o cultivos sin suelo para poder producir técnicas de cultivos hortícolas en invernaderos. El futuro de la hidroponía depende en gran medida del desarrollo de sistemas de producción que sean competitivos en costos con agricultura de campo.

Hay varios tipos de sistemas hidropónicos, así como algunos diseños de estructuras para invernadero y muchos métodos de control del medio ambiente pero no todos los sistemas son rentables en cada ubicación. Si bien las técnicas de cultivo hidropónico en los trópicos pueden ser bastante similares a los utilizados en las regiones templadas, las propias estructuras de los invernaderos y los métodos del control pueden diferir en gran medida.

La Hidroponía es una tecnología para cultivar plantas en soluciones de nutrientes (agua y fertilizantes) con el uso de suelo como medio artificial (por ejemplo, arena, grava, cascara de arroz, lana de roca) para proporcionar el soporte mecánico de las plantas. También existen los sistemas hidropónicos líquidos que tienen un medio sólido de soporte (bases plásticas) para las raíces de las plantas.

Los sistemas hidropónicos también se clasifican en abiertos y cerrados. En los sistemas abierto una vez que se entrega la solución de nutrientes a las raíces de la planta, esta no se reutiliza y en los sistemas cerrados la solución nutritiva sobrante se recupera y se reabastece al sistema nuevamente (Jensen, 2017).

Algunos productores y agencias de publicaciones continúan limitando la definición de hidroponía a sistemas líquidos solamente y esta exclusión de la hidroponía la utilizan para desenfocar datos estadísticos que pueden llevar a una subestimación del alcance de la tecnología y sus implicaciones económicas.

Prácticamente todos los sistemas hidropónicos del mundo están encerrados en estructuras de tipo invernadero para proporcionar control de temperatura, reducir la pérdida de agua por evaporación, reducir infestaciones de enfermedades, plagas y proteger los cultivos contra los elementos del clima, como el viento y lluvia. (Dalrymple DG, 1973).

Las principales ventajas del cultivo hidropónico incluyen el máximo rendimiento del cultivo con una alta densidad de producción, es decir en donde no exista un suelo adecuado se puede cultivar, además son virtualmente indiferentes a la temperatura ambiente, también realizan un uso más eficiente del agua y de los fertilizantes adicionalmente es altamente adecuado para la automatización y el control de las enfermedades.

Otra gran ventaja de la hidroponía, en comparación con el cultivo de las plantas en el suelo, es el aislamiento que tiene el cultivo, ya que a menudo el suelo tiene muchos problemas de enfermedades, salinidad, carencia de nutrientes y drenaje. Una de las tareas más costosas y que requieren muchísimo tiempo es la esterilización, la cual es innecesaria en los sistemas hidropónicos, asimismo otra ventaja importante si se desea realizar un cambio de cultivos se lo logra fácil y rápidamente.

Las principales desventajas de la hidroponía, en relación con la agricultura convencional, son los altos costos de los insumos, la energía requerida y el alto grado de gestión para lograr una producción exitosa. Los costos iniciales pueden ser excesivos si las estructuras son calentadas artificialmente y enfriadas por evaporación mediante sistemas de ventiladores o control ambiental, aunque no siempre son necesarios en los trópicos. Debido a sus costos significativamente más altos, las aplicaciones exitosas de la tecnología hidropónica se limitan a cultivos de alto valor económico para regiones específicas, a menudo confinadas a épocas específicas del año.

Por su naturaleza, los sistemas de cultivo hidropónicos son sistemas cerrados en donde las raíces de las plantas están expuestas directamente a la solución de nutrientes sin otro medio de cultivo y esta solución se reutiliza nuevamente. Existen varios sistemas en esta categoría, como la técnica *Nutrient Film Technique (NFT)* y *Deep Flow Hydroponics* (Hidroponía De Flujo Profundo). Este último, se ha convertido posiblemente en el sistema hidropónico más popular para producción de vegetales de hojas y hierbas. (Graves CJ, 1983) (Arnon, 1938) (Collins, 1985) (Winsor, 1979).

- ***Deep Flow Hydroponics (DFT)***. Es un método para cultivar lechuga y otras verduras, fueron desarrolladas de forma independiente por Jensen (Jensen, Collins 1985) en Arizona y Massantini (1976) en Italia para la producción a gran escala y se han popularizado en muchos países del mundo. Por ejemplo, en el Caribe, la producción de lechuga ha sido posible combinando este sistema de hidroponía con el enfriamiento de la solución nutriente. La solución de nutrientes es monitoreado,

reabastecido, recirculado y aireado. En la tabla 1 se observa una comparación entre las ventajas y desventajas.

Tabla 1 Ventajas y desventajas Deep Flow Hydroponics (DFT).

VENTAJAS	DESVENTAJAS
<ul style="list-style-type: none"> • Ayudan a mantener la temperatura adecuada de la solución de nutrientes para que las plantas crezcan y se desarrollen. • La solución nutritiva circula alrededor de las raíces de la planta utilizando una bomba o gravedad. • Es especialmente útil en la distribución de nutrientes esenciales como el calcio y el potasio. • Las raíces de las plantas se sumergen constantemente en la solución de nutrientes. 	<ul style="list-style-type: none"> • Es probable que se necesite un enfriador para la solución de nutrientes. • Puede formar espuma cuando usa productos orgánicos. • Existen fluctuaciones de pH por lo que se requiere que este sea siempre monitoreado. • Las raíces pueden crecer tanto que pueden obstruir las líneas de irrigación.

Fuente: (Hawthorne Gardening, 2017)

- **Nutrient Film Technique (NFT).** Esta técnica se desarrolló a finales de 1960 por el Dr. Allan Cooper en el *Glasshouse Crops Research Institute en Littlehampton* en Inglaterra. NFT ha dado lugar a varios sistemas modificados que se utilizan para la producción de vegetales principalmente lechuga. Para cultivos como los tomates, la solución de nutrientes se la realiza a través de canales plásticos que contienen las raíces de las plantas. Las paredes de los canales son flexibles lo que permite que se junten alrededor de la base de cada planta, excluyendo la luz lo que genera que se prevenga la evaporación.

La solución de nutrientes se bombea al extremo superior de cada canal y fluye por gravedad pasado por las raíces en las tuberías de captación y un sumidero. La solución se controla para la reposición de sales y agua antes de que se recicle. Una ventaja principal del sistema *NFT* en comparación con otros es que requiere mucho menos solución de nutrientes. En la tabla 2 se aprecia una comparación entre las ventajas y desventajas.

Tabla 2. Ventajas y Desventajas de Nutrient Film Technique (NFT).

VENTAJAS	DESVENTAJAS
<ul style="list-style-type: none"> - El costo de construcción de un sistema <i>NFT</i> es bajo. - El movimiento constante del agua evita que el agua se estanque. - Se pueden cultivar muchos de plantas. - Puede usar cualquier medio de cultivo para mantener la planta. - No se necesita un temporizador para que fluya la solución nutritiva porque el agua siempre fluye por gravedad. 	<ul style="list-style-type: none"> - Existen fluctuaciones de pH por lo que se requiere que este sea siempre monitoreado. - Si la bomba se daña, las plantas morirán rápidamente ya que no hay medios para contener el agua. - Las plantas con raíces cortas necesitarán ser alimentadas por la parte superior o incorporar líneas de goteo, hasta que las raíces crezcan hasta el fondo del PVC.

Fuente: (Hawthorne Gardening, 2017)

- ***Aeroponics***. En una aplicación inusual de hidroponía de sistema cerrado, en donde las plantas se cultivan en agujeros de poliestireno expandido u otro material. Las raíces de las plantas están suspendidas en el aire y encerrado en una caja sellada de pulverización para que las raíces estén en oscuridad (para evitar el crecimiento de algas) y en la humedad de saturación. Un sistema de nebulización rocía la solución nutritiva sobre las raíces periódicamente.

El sistema normalmente está encendido por solo unos pocos segundos cada 2-3 minutos. Esto es suficiente para mantener las raíces húmedas y la solución de nutrientes gaseado. Los sistemas fueron desarrollados en Arizona para lechuga, espinacas e incluso tomates, aunque se consideró que el último no era económicamente viable. No se conocen operaciones aeropónicas comerciales a gran escala, aunque varias pequeñas empresas comercializan sistemas para uso doméstico. En la tabla 3 se detalla una comparación entre las ventajas y desventajas.

Tabla 3 Ventajas y desventajas de Aeroponics.

VENTAJAS	DESVENTAJAS
<ul style="list-style-type: none"> - Alta relación entre el oxígeno y el agua. - Existe una reutilización el agua. - Existe un aumento de la absorción de la solución de nutrientes y un crecimiento acelerado. - 	<ul style="list-style-type: none"> - Tiene las más altas fluctuaciones de pH. - Las raíces necesitan ser regadas con más frecuencia, por cortos períodos de tiempo, por lo tanto, se pueden secar muy rápido si no se les da suficiente solución nutritiva. - El rociador se obstruye fácilmente, por lo que no es bueno para los cultivos orgánicos.

Fuente: (Hawthorne Gardening, 2017)

- **Sistemas Hidropónicos Agregados.** En los sistemas hidropónicos agregados un medio sólido e inerte proporciona soporte para las plantas. Al igual que en los sistemas líquidos, la solución de nutrientes se administra directamente a las raíces de las plantas. Los sistemas agregados pueden ser abiertos o cerrados. Los sistemas abiertos no reciclan la solución de nutrientes y los sistemas cerrados si lo hacen.

En la mayoría de los sistemas hidropónicos abiertos, se recupera el exceso de solución de nutrientes, pero el excedente no se recicla, sino que se desecha en estanques de evaporación o se utiliza para regar las plantaciones adyacentes y debido a que la solución de nutrientes no se recicla, dichos sistemas abiertos son menos sensibles a la composición del medio utilizado o a la salinidad del agua. En la tabla 4 se aprecia una comparación entre las ventajas y desventajas.

Tabla 4 Ventajas y desventajas de los Sistemas Hidropónicos Agregados

VENTAJAS	DESVENTAJAS
<ul style="list-style-type: none"> - El costo de implementación es bajo. - Puede tener una alimentación de la solución de nutrientes constante o mediante un sistema temporizado. 	<ul style="list-style-type: none"> - Puede obstruirse fácilmente, por lo que no es bueno para los cultivos orgánicos. - Necesita mucho más tiempo para su configuración inicial.

<p>- Se puede optar por las opciones de reutilizar o no la solución de nutrientes. Está constantemente limpiando la solución de arriba hacia abajo, por lo que no tiene mucha acumulación de sal.</p>	<p>- Dado que usualmente usas más medios que en comparación con otros métodos hidropónicos, la tasa de absorción es más lenta.</p>
---	--

Fuente: (Hawthorne Gardening, 2017)

Formulación

Según la FAO (Organización de las Naciones Unidas para la Alimentación y la Agricultura) que es una Organización sin ánimo de lucro, se estima que para el año 2050 la población mundial será de 9100 millones de personas, un 34 % superior a la de hoy en día, y prácticamente la totalidad de este incremento de la población tendrá lugar en los países en desarrollo. La urbanización continuará a un ritmo acelerado, aproximadamente el 70 % de la población mundial será urbana (en la actualidad esta cifra es del 49 %) y para poder alimentar a esta población tan numerosa, la producción de alimentos deberá aumentar aproximadamente en un 70%.

Si se considera que el rendimiento de la producción se incrementa linealmente en el ámbito mundial como es la tendencia registrada durante los últimos 50 años, ese provee que para las próximas décadas la producción de alimentos no será suficiente para satisfacer las necesidades alimentarias del mundo. (Organización de las Naciones Unidas para la alimentación y la agricultura, 2017).

Por lo tanto, el desafío actual es que con la utilización de tecnología de punta en automatización y nuevas técnicas de cultivo alternativo en ambientes controlados como la hidroponía se pueda solucionar en parte la producción de alimentos.

Justificación:

En la actualidad existen muchos métodos de cultivo, los invernaderos tradicionales y los cultivos hidropónicos cada uno con sus pros y contras. El método hidropónico tiene muchas más ventajas que los cultivos tradicionales ya que es un sistema totalmente confiable, con un bajo nivel mantenimiento, que produce cultivos de mejor calidad y una tasa menor de enfermedades.

La continua investigación y el desarrollo de sistemas automáticos pueden llevar en un futuro a mejorar la relación costo-beneficio entre la energía y los materiales utilizados, logrando así que los nuevos cultivos se los realicen en ambientes controlados más apropiados para tener un mejor control.

También existe un alto interés para impulsar el desarrollo tecnológico de automatización para mejorar la producción de los invernaderos en la agricultura, ya que utilizando la tecnología más avanzada se puede solucionar los problemas actuales de producción debido a que los alimentos en invernaderos son altamente productivos.

Sin embargo, parece que no hay una base para el uso de una tecnología mucho más amplia y rápida de lo que está ocurriendo actualmente para el crecimiento en ambientes controlados. La agricultura depende en gran medida del desarrollo de sistemas de producción que son competitivos en costos con los de la agricultura tradicional (Castilla, 1995) (Jensen, 2017).

Alcance:

En el siguiente proyecto se plantearán las bases teóricas, el programa y el algoritmo para el funcionamiento de un prototipo de un control automático para un contenedor de carga para lograr una funcionalidad básica basada en crear un ambiente controlado en donde se puedan sembrar cultivos y verificar su crecimiento.

Funciones avanzadas como: poder modificar y calibrar remotamente los parámetros de temperatura, humedad, iluminación, tiempo e intervalo de riego, no serán realizadas. Tampoco se creará alguna aplicación de software para PC, Tablet y Smartphone.

Al ser un prototipo se creará un contenedor a escala donde se implementará el circuito de control para verificar su funcionamiento y este a su vez podrá ser utilizado como base para en un futuro desarrollar un circuito comercial mucho más elaborado, con una interfaz de control remoto para uso mediante el internet y también el manejo de otros parámetros como el control de pH, temperatura de la solución nutritiva, los niveles de los estanques, los cuales fueron incluidos en el diseño.

OBJETIVOS

Objetivo general:

Desarrollar un sistema de automatización de un contenedor de carga para el control de clima e iluminación de cultivos hidropónicos.

Objetivos específicos:

- Definir los parámetros de control de temperatura, humedad, iluminación nutrición y riego.
- Diseñar el sistema de automatización de un contenedor de carga para el control de clima y crecimiento de cultivos hidropónicos según los parámetros establecidos en el objetivo general.
- Desarrollar el algoritmo de control y automatización para el control automático.
- Construir el sistema automatizado y circuito electrónico para controlar temperatura, humedad, nutrición, riego e iluminación para cultivos hidropónicos.
- Realizar pruebas de funcionamiento y validación de resultados para cultivos hidropónicos.

Descripción del contenido de los capítulos:

Capítulo 1: en este capítulo se presenta el marco teórico relacionado con el dispositivo desde un punto de vista de elaboración, se hará una breve explicación de los componentes y principios sobre los cuales funciona el prototipo.

Capítulo 2: en este capítulo se evalúan las opciones de diseño e implementación del programa y se analiza la máquina de estados finitos.

Capítulo 3: en este capítulo se documenta el proceso de elaboración y las opciones para la construcción del circuito.

Capítulo 4: en este capítulo se resumen las pruebas realizadas de operación y se documentan las experiencias de implementación.

Conclusiones y recomendaciones: se presentan los resultados de las pruebas realizadas durante la implementación del prototipo. Se presentarán algunas recomendaciones de mejora para trabajos futuros.

CAPÍTULO 1

FUNDAMENTOS TEÓRICOS

En este Capítulo se detalla el diagrama de bloques, los fundamentos teóricos del funcionamiento del microprocesador, los sensores para adquirir los datos de humedad/temperatura, también los elementos externos como RTC, el módulo I²C, las luces Led para horticultura y circuitos para controlar la etapa de potencia.

El concepto de garantizar el suministro de alimentos en el futuro no consiste solo en sembrar en un buen lugar los cultivos, lo que realmente necesitaremos será una granja de última tecnología, con una gran cantidad de técnica de cultivo y una eficiente automatización que nos permita proporcionar las condiciones óptimas para que las plantas se desarrollen rápido y sanamente. Esto lo lograremos controlando los parámetros más importantes para el crecimiento como la humedad relativa, temperatura, el tipo de Iluminación, tiempos de riego e intervalos de alimentación.

1.1.Marco teórico

Hay varios tipos de microcontroladores e interfaces accesibles en el mercado, pero debido al conocimiento recibido en las materias de Microprocesadores y Control Automático para desarrollar este proyecto integrador de carrera se eligió como controlador al módulo Arduino UNO Wifi que es una interfaz de electrónica de código abierto fundamentada en hardware y con una programación simple de utilizar.

La arquitectura del sistema se muestra en el diagrama de bloques de la figura 1.1, donde se puede observar que consta con dos sensores de entrada, uno de humedad y otro de temperatura, también se utiliza el módulo de Arduino UNO Wifi como controlador para procesar las entradas y salidas a través de conversores analógicos/digitales, el cual incorpora un módulo Wifi ESP8266 que genera una red Wifi local (servidor) y también sirve para conectarse al Internet a través de una dirección IP pública programada en el Arduino Uno Wifi. Mediante la interface del módulo Wifi se puede acceder de forma remota mediante

una computadora, *tablets* o *smartphones* (*Android* o *iOS*) visualizar como varían los valores de los parámetros de los sensores tanto de temperatura como de humedad, verificar que la iluminación luz LED de crecimiento está encendida o apagada, ver el tiempo programado de riego y también los intervalos programados para que el riego se lo realice periódicamente.

Una de las salidas de módulo Arduino UNO Wifi controlan una pantalla LCD (*liquid-crystal display*) monocromática y alfanumérica de 20 caracteres por 4 filas (20x4) que es controlada por un módulo Arduino I²C *Inter-Integrated Circuit*.

Las otras salidas del módulo Arduino controlan la etapa de potencia las cuales nos sirven para manejar los dispositivos electrónicos de alto voltaje mediante relés que funcionan como un interruptor para activar las luces LED de crecimiento, el calefactor, aire acondicionado el humidificador y el deshumificador.

En la figura 1.1 se detalla en general el diagrama de bloques que se utiliza para el funcionamiento del sistema de automatización.

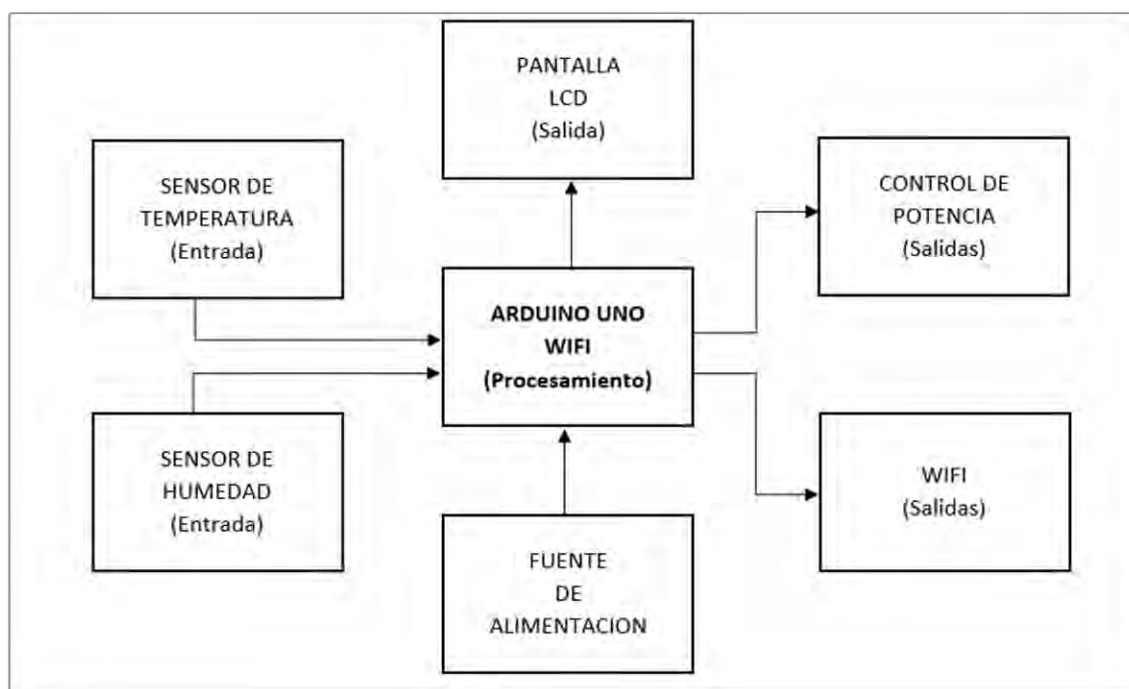


Figura 1. 1. Diagrama de bloques

Fuente: Elaborado por el autor

A continuación, se describirá el componente teórico de cada bloque, así como cada uno de sus componentes. Las hojas técnicas (*datasheets*) de cada componente se lo presentan en la sección de Anexos.

1.2. Microcontrolador

1.2.1. Arduino Uno Wi-Fi

El Arduino Uno Wi-Fi es similar al Arduino Uno Rev.3 pero con un módulo de comunicación inalámbrica Wi-Fi integrado. La placa se basa en ATmega328P con un módulo de Wi-Fi ESP8266 integrado. Tiene 14 pines digitales de IN/OUT (de los cuales 6 pueden usarse como salidas *PWM*), 6 entradas analógicas, un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un encabezado *ICSP* y un botón de reinicio (*Reset*), para su alimentación solo hace falta conectarlo a una computadora con un cable USB, un adaptador de CC de 12VCD o a una batería. El módulo Arduino UNO Wifi se detalla en la figura 1. 2.

El Módulo Wi-Fi ESP8266 es un *SoC (System on a Chip)* autónomo con pila de protocolos TCP / IP integrados que puede brindar acceso a una red Wi-Fi con la ventaja de poder actuar como un punto de acceso.



Figura 1. 2. Arduino Uno Wi-Fi
(Arduino, 2016)

1.3. Dispositivos de entrada

Sensor de humedad relativa y temperatura DTH22

El sensor de humedad relativa y temperatura DHT22, usa la tecnología de recolección de datos digitales exclusivamente, asegurando su confiabilidad y seguridad. Todos sus elementos están conectados con un solo chip de 8 bits, cada sensor internamente está compensado por temperatura, para lo cual el fabricante utiliza una cámara de calibración y el resultado de este proceso que es un coeficiente de temperatura se almacena en el programa de la memoria interna *OTP*. En la figura 1. 3. Se detalla el sensor de humedad y temperatura DTH22.



Figura 1. 3. Sensor de humedad relativa y temperatura DTH22.

Fuente: (Rduinostar.com, 2015)

Por su tamaño reducido, bajo consumo de energía y transmisión de datos de larga distancia aproximadamente de 20 metros, hace que el módulo DHT22 sea adecuado para el diseño e implementación del proyecto ya que el módulo esta empaquetado en una placa de tres pines, lo que hace que la conexión sea más simple y versátil.

Cuando se suministra energía al sensor, este no envía ninguna información dentro del primer segundo hasta poder pasar del estado inestable al estable. Se recomienda colocar un capacitor de 100nF entre VDD y GND para realizar un filtro en la alimentación del sensor. En la tabla 1.1 se describe las características eléctricas del Sensor DTH22.

Tabla 1. 1. Características eléctricas del DTH22

DESCRIPCION	CONDICIÓN	MÍNIMO	TÍPICO	MÁXIMO	UNIDADES
Fuente Alimentación	DC	3.3	5	6	[V]
Consumo Energía	Medición	1	N/A	1.5	[mA]
	Stand-by	40	N/A	50	[uA]
Tiempo Medida	Segundos	N/A	2	N/A	[s]

Fuente: (Adafruit.com, 2018, pág. 5)

El sensor DTH22 de acuerdo con las especificaciones del fabricante tiene las siguientes características para medir la temperatura:

- **Resolución Decimal:** que son los valores de humedad y temperatura representados con una cifra decimal.
- **Tiempo de muestreo:** que es de 2 segundos, sólo puede entregar datos correctos cada 2 segundos.
- **Rango de valores:** que va desde los -40°C hasta los 80°C de temperatura.
- **Precisión:** que es de $\pm 0.5^{\circ}\text{C}$, y hasta $\pm 1^{\circ}\text{C}$ cuando se encuentra en condiciones adversas.
- **Tiempo de respuesta:** es menor a 10 segundos, es decir que tarda menos de 10 segundos en mostrar un cambio de temperatura en el entorno.

Adicionalmente el sensor DTH22 para medir la humedad relativa tiene las siguientes características:

- **Rango de valores:** que va desde el 0% hasta el 99.9% de Humedad Relativa.
- **Precisión:** que es de $\pm 2\%$ de Humedad Relativa, a una temperatura de 25°C .
- **Tiempo de respuesta:** que es menor a 5 segundos, es decir que tarda menos de 5 segundos en mostrar un cambio de humedad relativa en el entorno y adicionalmente la velocidad del aire debe ser de 1 [m/s] . (Adafruit.com, 2018, pág. 1).

Para la comunicación se utiliza un bus simple de datos entre el Módulo Arduino y el sensor DHT22. El tiempo requerido para la comunicación es de 5 ms y los datos tienen una longitud de 40 bits los cuales se dividen en dos partes, una para la parte entera y la otra para la parte decimal. Para el proceso de transmisión de los datos se envía primero un bit de datos en alto y luego la trama de datos de 40 bits la cual se divide de la siguiente manera:

- Datos de humedad relativa de los valores enteros en formato de 8 bits.
- Datos de humedad relativa de los valores decimales en formato de 8 bits.
- Datos de temperatura de los valores enteros en formato de 8 bits.
- Datos de temperatura de los valores decimales en formato de 8 bits.
- *Checksum* (suma de chequeo) de verificación de 8 bits.

Si la transmisión de la trama de datos es correcta, la suma de verificación (*Checksum*) debe ser los últimos 8 bits de datos enteros humedad relativa + 8 bits de humedad relativa decimales + 8 bits de temperatura datos enteros + 8 bits de datos de temperatura decimales.

Cuando el Módulo Arduino envía la señal de inicio, el DHT22 pasa del modo de bajo consumo de energía al modo de desempeño y enviará la señal de respuesta de datos de 40 bits con la información de la temperatura y humedad relativa. Mediante el siguiente ejemplo se explicará como con los 40 bits de datos que envía el sensor DHT22, el módulo Arduino interpreta los valores, esto se muestra en la tabla 1.2.

Tabla 1. 2. Bits de datos del DHT22

0000 0010 1000 1100	0000 0001 0101 1111 1110	1110
16 Bits Humedad Relativa (HR)	16 Bits Temperatura (T)	Checksum

Fuente: (Adafruit.com, 2018, pág. 3)

Primero se convierte los datos de Humedad Relativa (HR) de 16 bits del sistema binario al sistema decimal, **0000 0010 1000 1100** → **652**

$$HR = 652/10 = 65.2\%$$

Luego se convierte los datos de Temperatura (T) de 16 bits del sistema binario al sistema decimal, **0001 0101 1111** → **351**

$$T = 351/10 = 35.1\text{ °C}$$

Y al final se realiza la suma total para verificar el *Checksum* (Suma de Chequeo)

$$\text{Suma} = 0000\ 0010 + 1000\ 1100 + 0000\ 0001 + 0101\ 1111 = \mathbf{1110\ 1110}$$

El *checksum* es igual a los últimos 8 bits de Suma = 1110 1110

Nota: Cuando el bit más alto de temperatura es 1, significa que la temperatura está por debajo de 0 grados Celsius. Ejemplo: 1000 0000 0110 0101, $T = -10.1\text{ }^{\circ}\text{C}$.

En la figura 1.4. se observa cómo se conecta el módulo DHT22 al Arduino UNO y como la trama de datos a través del pin digital es enviada.

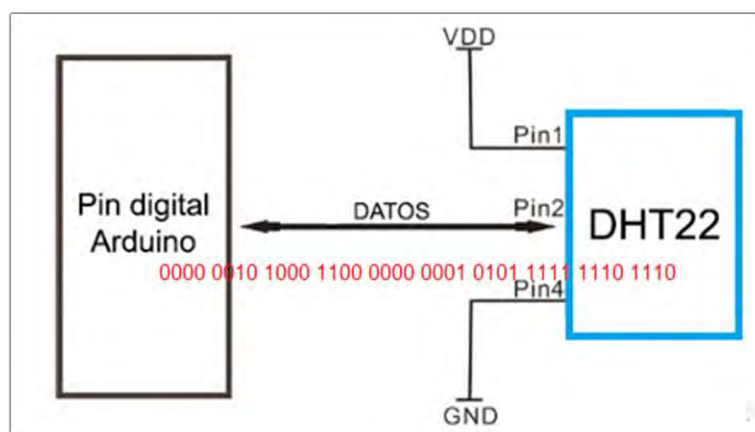


Figura 1. 4. Esquema de conexión DTH22

Fuente: (Adafruit.com, 2018, pág. 2)

1.4. Dispositivos Especiales

1.4.1. Reloj En tiempo real *RTC*

El *RTC* (*Real Time Clock*). de Arduino es un módulo que sirve para realizar un seguimiento de la hora actual y que se puede utilizar para programar acciones en un momento determinado. La mayoría de los *RTC* usan un oscilador de cristal cuya frecuencia es 32.768 [kHz] que es la misma frecuencia utilizada en relojes de cuarzo.

Como esta frecuencia es igual a 2^{15} ciclos por segundo, esta velocidad es muy conveniente para utilizarlo con circuitos de contador binarios. Además, el *RTC* puede continuar funcionando en modo de suspensión o *Sleep*, por lo que se puede utilizarlo para activar el dispositivo desde cualquier parte del programada.

Cada vez que él modulo se enciende, el *RTC* se restablece y comienza a partir de una fecha estándar preprogramada. Para mantener el tiempo y el *RTC* en funcionamiento, es necesario mantener el módulo alimentado conectado a través de un diodo al pin de 3.3V. Para evitar que se pierda la hora el módulo *RTC* contiene una batería de litio el rango de 3V

que sirve como respaldo. En la figura 1.5. se muestra el módulo RTC de Arduino con los pines utilizados para la alimentación y datos.

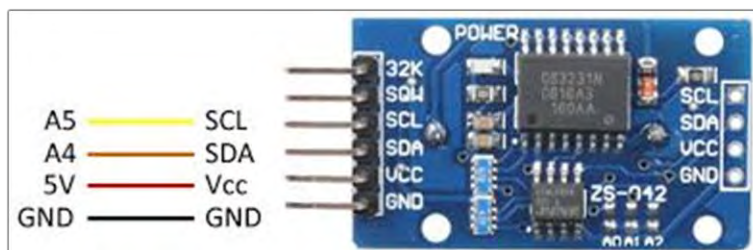


Figura 1. 5. Módulo RTC de Arduino

Fuente: (Maxim Integrated, 2014)

1.4.2. Modulo I²C

El bus de comunicación I²C es muy popular y ampliamente utilizado por muchos dispositivos electrónicos, ya que permite implementar fácilmente muchos diseños electrónicos que requieren comunicación entre un maestro y múltiples dispositivos esclavos o incluso múltiples dispositivos maestros. Cuando se utiliza un direccionamiento de 7 bits, se puede conectar hasta casi 128 (112) dispositivos y cuando se utiliza el direccionamiento de 10 bits hasta 1024 (1008) dispositivos según los datos del fabricante.

En algunas situaciones, puede ser útil configurar dos o más módulos Arduino para compartir información entre ellos. Por ejemplo, el módulo de *Display LCD* y el *RTC* están programadas para comunicarse entre sí mediante una configuración de Receptor principal / Receptor secundario a través del protocolo en serie síncrono I²C. En la figura 1.6. se observa el módulo I²C de Arduino con los pines utilizados para la alimentación y datos.

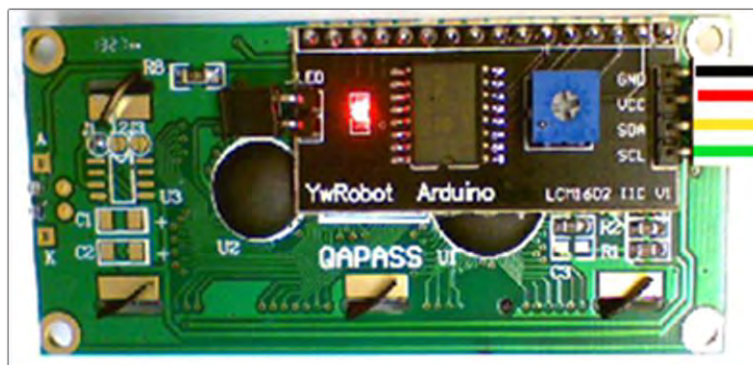


Figura 1. 6. Módulo I²C de Arduino

Fuente: (Opencircuit, 2018, pág. 1)

El protocolo I²C implica el uso de dos buses para enviar y recibir datos: un pin de reloj serial (*SCL*) que el módulo Arduino genera en intervalos regulares y un pin de datos serie (*SDA*) sobre el cual se envían los datos entre los dispositivos. A medida que la línea del reloj cambia de baja a alta (conocida como el flanco ascendente del pulso del reloj), un solo bit de información - que formará en secuencia la dirección de un dispositivo específico y un comando o datos - se transfiere del módulo a el dispositivo I²C sobre la línea *SDA*.

Cuando se envía esta información, poco a poco el dispositivo llamado ejecuta la solicitud y transmite sus datos utilizando la señal de reloj aún generada por el maestro en *SCL* como temporización. Los ocho bits iniciales (es decir, ocho pulsos de reloj) del maestro a los esclavos contienen la dirección del dispositivo maestro que quiere los datos. Los siguientes bits contienen la dirección de memoria en el esclavo de la que el maestro desea leer o escribir datos. Los módulos con sus respectivos pines se detallan en la tabla 1.3.

Tabla 1. 3. Módulos y pines I²C

Modulo Arduino	I ² C / TWI Pines
Uno, Ethernet	A4 (<i>SDA</i>), A5 (<i>SCL</i>)
Mega2560	20 (<i>SDA</i>), 21 (<i>SCL</i>)
Leonardo	2 (<i>SDA</i>), 3 (<i>SCL</i>)
Due	20 (<i>SDA</i>), 21 (<i>SCL</i>), <i>SDA1</i> , <i>SCL1</i>

Fuente: (Naylamp Mechatronics SAC, 2016)

1.5. Dispositivos de salida

Led´s para Horticultura

En la agricultura tradicional durante el día las hojas reciben directamente la luz del sol. Las longitudes de onda del espectro de luz visible desempeñan papeles específicos para el crecimiento de la planta ya que producen pigmentos fotosintéticos dentro de las plantas.

Utilizando el avance de la tecnología para aumentar la tasa de crecimiento de los cultivos se ha creado un tipo de luces LED diseñadas para el crecimiento de las plantas las cuales emiten el espectro de luz de una manera muy específica para lograr resultados superiores en el crecimiento de los cultivos, de manera que mejora tanto el metabolismo primario como el secundario de las plantas el cual influye directamente con el crecimiento y la calidad de la planta.

Las luces LED para crecimiento están diseñadas para la gran mayoría de aplicaciones de cultivo (interior o exterior) donde se necesita iluminación principal y suplementaria. Estas

luces LED emiten luz en la región fotosintética del espectro de luz visible y pueden soportar la mayoría de los entornos existentes en los invernaderos durante toda su vida útil, además el diseño optimizado de las luces LED para crecimiento las hace fáciles de instalar y su avanzado diseño es el catalizador para lograr una óptima fotosíntesis para mejorar la productividad de la planta.

En la Tabla 1.4. Se realiza una comparación entre el espectro de luz visible y el porcentaje del contenido espectral de acuerdo con la longitud de onda.

Tabla 1. 4. Contenido espectral de los LED para crecimiento

CONTENIDO ESPECTRAL (% DEL TOTAL)		
Longitud de Onda	Espectro de luz visible	Contenido Espectral
400-500 nm	33%	2-14 %
500-600 nm	41%	1-5 %
6000-700 nm	26%	81-97%
100 %		

Fuente: (OSRAM Opto Semiconductors, 2018, pág. 4)

En la horticultura utilizan 3 tipos LED's con diferentes rangos de longitud de onda para irradiar a los cultivos:

- *Deep Blue* (azul intenso) con una longitud de onda de 450 [nm].
- *Hyper Red* (rojo hiper) con una longitud de onda de 660 [nm]
- *Far Red* (rojo lejano). con una longitud de onda de 730 [nm].

Estas longitudes de onda están consideradas como las más recomendables para tratar de cubrir el mayor espectro de luz visible según OSRAM Opto Semiconductors. Su dispersión es en tres ángulos de radiación 80°, 120° y 150° lo que ayuda a proporcionar la iluminación perfecta para todo tipo de plantas y permite adaptar la luz exactamente a las necesidades de diversos cultivos.

La distribución espectral de los LED's para crecimiento en la actualidad ha mejorado muchísimo y los ha llevado a un nivel tan alto, que actualmente de la absorción de luz de las

plantas se maneja a niveles tan pequeños como de los fotones. Además, los LED's de crecimiento ofrecen una infinidad de posibilidades especialmente para las técnicas de cultivo modernas, como la iluminación de superficies, agricultura vertical de varias capas o fuentes de iluminación individuales. En la figura 1.7. se observa la distribución espectral de luz de los LED para crecimiento.

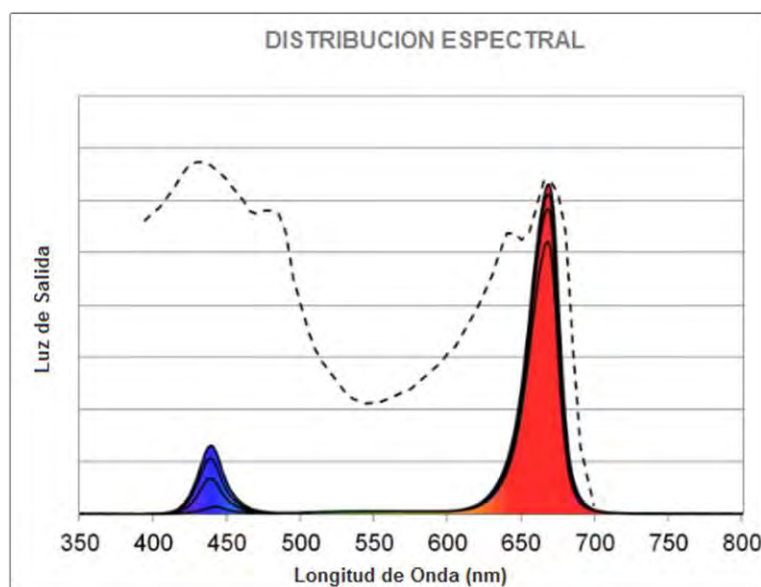


Figura 1. 7. Distribución espectral de luz de los LED para crecimiento

Fuente: (OSRAM Opto Semiconductors, 2018, pág. 12)

Este tipo de iluminación cubre la gama completa de requisitos específicos para la mayoría de los tipos de plantas, como su diseño es muy compacto con un consumo de energía es bajo, permite que su utilización sea fácil, simple y eficiente.

Los LED's de crecimiento utilizados para este diseño para mejorar el rendimiento y la productividad del cultivo tienen las características que se observan en la tabla 1.5.:

Tabla 1. 5. Especificaciones de los LED's de crecimiento

ESPECIFICACIONES	
Alimentación	120 VAC / 50-60 Hz
Consumo de Energía	110 W
Salida de luz	2.2 - 2,4 umol/s per-Watt
Cobertura de luz	Hasta 12(m ²)
Regulador de intensidad	10 – 100 %
Depreciación de la cantidad de luz	L ₈₅ B ₁₀ 50.000 horas
Temperatura de operación	0-104 °F (0-40°C)

Grado de Protección	IP54
----------------------------	------

Fuente: (OSRAM Opto Semiconductors, 2018)

En la figura 1.8 se puede observar la tonalidad violeta generada por la utilización de 3 tipos de LED con diferentes rangos de longitud de onda para irradiar a los cultivos.

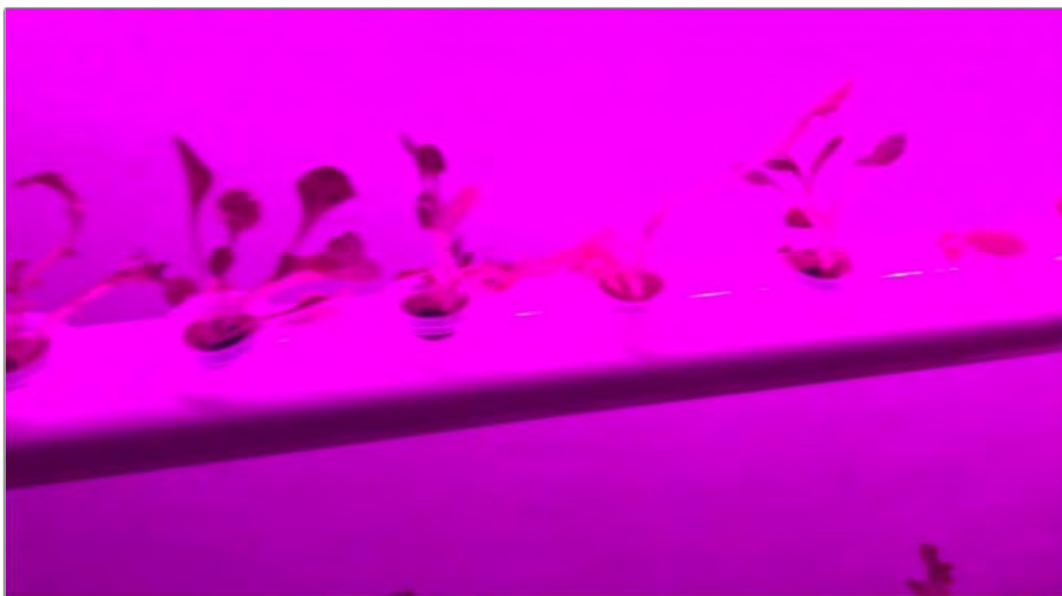


Figura 1. 8. Plantas con Iluminación LED para crecimiento Fuente: Elaborado por el autor

Pantalla LCD (Liquid Crystal Display)

Una pantalla de cristal líquido (LCD) es una pantalla plana que utiliza las propiedades de modulación de la luz de los cristales líquidos. Los cristales líquidos no emiten luz directamente por lo que necesitan una luz de retroalimentación para iluminarse. En la figura 1.9. se observa una pantalla LCD de 20x4.



Figura 1. 9. Pantalla LCD Fuente: Elaborado por el autor

Una pantalla LCD monocromática y alfanumérica de 20x4 significa que se pueden mostrar 20 caracteres en cada una de las 4 filas de la pantalla, por lo que se pueden mostrar un total de 80 caracteres en cualquier instancia de tiempo adicionalmente tiene una interfaz de 16 pines, los cuales se detallan en la tabla 1.6.

Tabla 1. 6. Pines del LCD 20x4

Pin	Símbolo	Función
1	Vss	Tierra (0 V)
2	Vdd	5 V +
3	Vo	Ajuste de Contraste
4	RS	H/L código de instrucción
5	R/W	H/L señal lectura/escritura
6	E	H/L señal de habilitación
7-14	DB0 – DB7	H/L bus de datos para modo de 4- o 8-bits
15	A (LED+)	Ánodo luz posterior
16	K (LED-)	Cátodo luz posterior

Fuente: (Vishay, 2016, pág. 1)

Etapa de potencia

Para la etapa de potencia se utilizó varios módulos de relé de Arduino, los cuales son dispositivos mecánicos que funcionan a partir de un pequeño voltaje de 5VCD aplicado a su bobina y con sus contactos puede manejar grandes cargas de potencia y también le permite aislar la etapa de potencia con la etapa de control.

El relé de 5VCD tiene 2 tipos diferentes de contactos eléctricos en el interior: normalmente abierto (*NO*) y normalmente cerrado (*NC*), su uso dependerá de si desea que la señal de +5V encienda el interruptor o apague el interruptor. El voltaje de funcionamiento es de 120-240 VAC, adicionalmente tiene el terminal común C en ambas configuraciones las cuales se conectan al dispositivo que desea controlar. El relé de 5VCD también tiene tres pines de bajo voltaje (tierra, Vcc y señal) que se conectan directamente al módulo de Arduino. En la figura 1.10. se muestra un Módulo de Relé +5V.

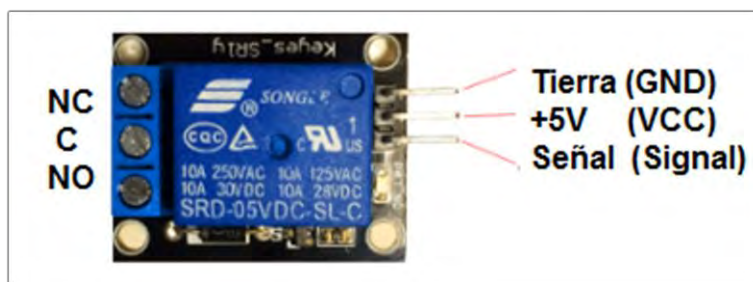


Figura 1. 10. Módulo de relé +5V

Fuente: Elaborado por el autor

Entre sus principales configuraciones están las siguientes:

Normalmente abierto (NO)

En esta configuración, cuando el relé recibe una señal ALTA (+5V), el interruptor se cierra y permite que la corriente fluya desde el terminal C al terminal NO. Una señal BAJA desactiva el relé y detiene el paso la corriente, lo cual se ilustra en la figura 1.11.



Figura 1. 11. Relé normalmente abierto (NO).

Fuente: Elaborado por el autor

Normalmente cerrado (NC)

En esta configuración cuando el relé recibe una señal ALTA (+5V), el interruptor se abre e interrumpe la corriente. Una señal BAJA cierra el interruptor y permite que la corriente fluya desde el terminal C al terminal NC. Los detalles se observan en la figura 1.12.

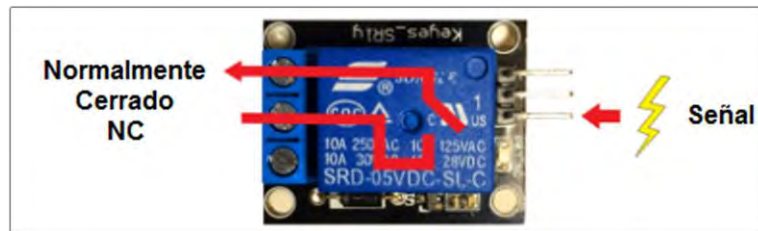


Figura 1. 12. Relé normalmente cerrado (NC). Fuente: Elaborado por el autor

CAPÍTULO 2

DISEÑO DEL PROGRAMA

En este Capítulo se detalla el diagrama de flujo, el funcionamiento de los sistemas retroalimentados, las máquinas de estados finitos, el desarrollo y la elaboración del programa de automatización.

2.1 Diagrama de Flujo.

Para el proyecto se diseñó un sistema de control automático para el manejo del cultivo hidropónico dentro de un contenedor y para poder iniciar con el programa lo primero a realizar es elaborar el diagrama de flujo del programa o algoritmo de programación, ya que de esa manera será mucho más fácil escribir el código de programa en el compilador de Arduino. La ventaja de los diagramas de flujo es que permiten explicar de una manera general el programa a otros y como se trata de un programa complejo el diagrama de flujo permitirá visualizar gráficamente la secuencia lógica del programa. En la figura 2.13. se puede observar el diagrama de flujo para el sistema de control automático con los parámetros de temperatura, humedad, iluminación, tiempo e intervalo de riego.

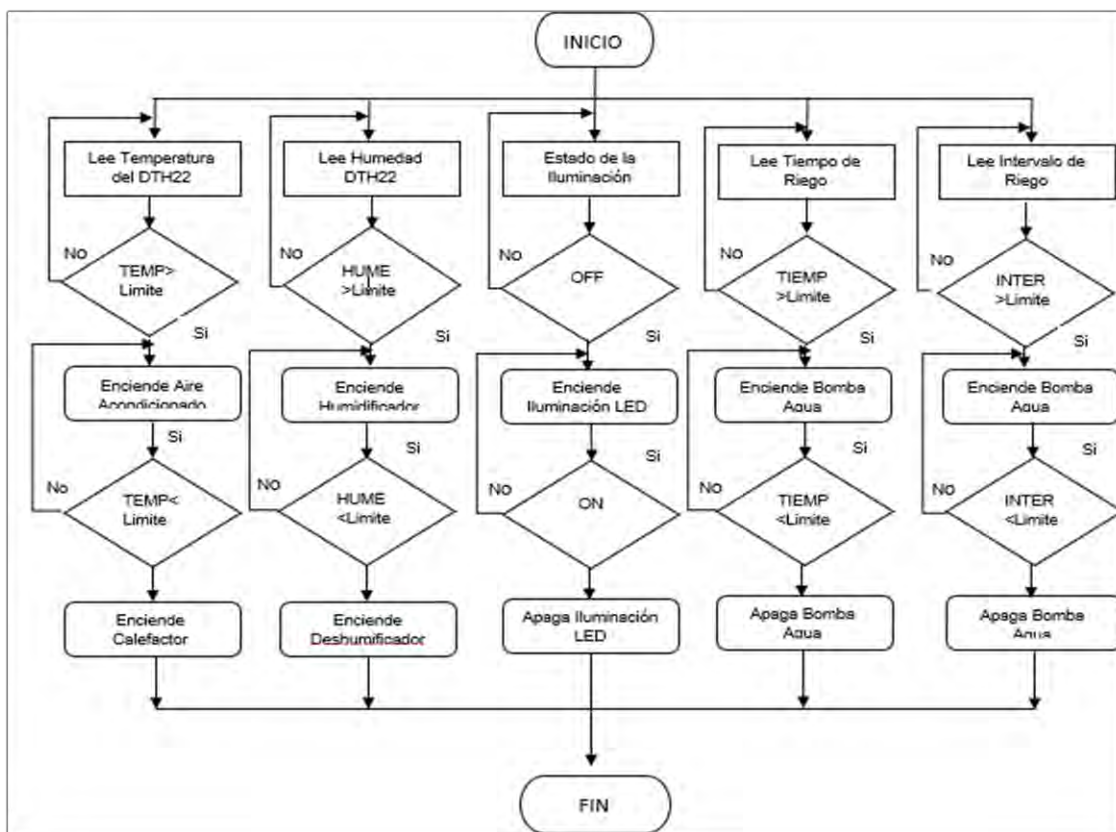


Figura 2. 13. Diagrama de flujo

Fuente: Elaborado por el autor

Luego de realizar el diagrama de flujo se procedió a la elaboración del código de programa creando una máquina de estados finitos que es un sistema de lazo cerrado, como se observa en la figura 2.14. y será el sistema de control principal para monitorear y controlar los parámetros de temperatura, humedad, iluminación, tiempo e intervalo de riego todo esto en una pantalla LCD o mediante un computador o Tablet mediante un explorador web, ingresando la dirección IP publica privada programada en el módulo Arduino Uno Wifi, lo que permite al usuario, la capacidad de controlar de forma automática y manual todo el sistema.

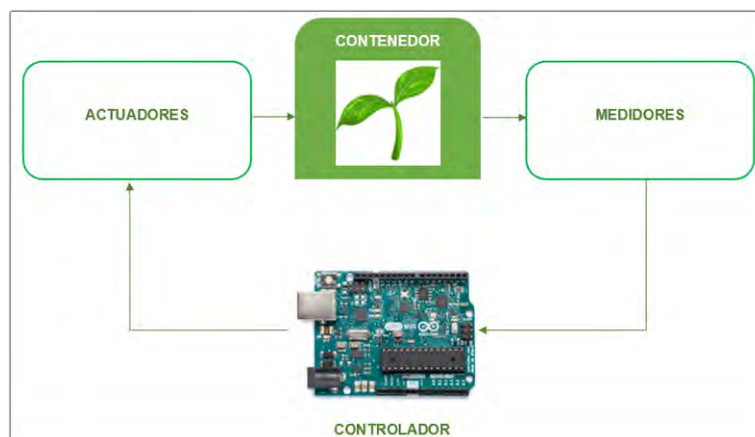


Figura 2. 14. Sistema de control retroalimentado

Fuente: Elaborado por el autor

2.2. Máquina de estados finitos.

Las máquinas de estados son sistemas de entradas y salidas, en donde las salidas dependen de las señales de entradas actuales y también de las anteriores, con lo cual se puede resumir que una máquina de estados es un autómatu o un conjunto de estados que relacionan las entradas y salidas para cada instante, para que la salida dependa únicamente del estado actual de las entradas.

Para la programación del sistema se definió los botones que vamos a necesitar BTN MENU, BTN EXIT, BTN UP y BTN DOWN y los pines de la placa Arduino Wifi a utilizar. Como ilustra la figura 2.15.

```

#define BTN_MENU    0           // Defino los Botones
#define BTN_EXIT    1
#define BTN_UP      2
#define BTN_DOWN    3

uint8_t button[4] = {          // Variable de los botones
  13,                          // PIN 13 Boton Menu
  12,                          // PIN 12 Boton Exit
  11,                          // PIN 11 Boton Up
  10,                          // PIN 10 Boton Down
};

uint8_t button_state[4];      // Variable estado de los botones

```

Figura 2. 15. Definición de botones y pines

Fuente: Elaborado por el autor



Figura 2. 16. Botones para el control Automático.

Fuente: Elaborado por el autor

Luego se definió los estados a utilizarse como se detallan en la figura 2.17. los cuales van a controlar el orden en el cual se van a desplazar dentro de la máquina de estados finitos y el nombre de la variable donde se va a guardar el estado actual.

```
#define S_HOME      0           // Defino los estados
#define S_SET_TEMP  1
#define S_SET_HUME  2
#define S_SET_LIGHT 3
#define S_SET_NUTRI 4
#define S_SET_INTER 5

uint8_t estado = S_HOME;      // Variable estado actual
```

Figura 2. 17. Definición de los estados

Fuente: Elaborado por el autor

Cabe mencionar que se creó un arreglo en el programa para que cada vez que se pulse un botón el sistema detecte un flanco de subida en el tiempo, para lo cual se hizo que la entrada digital sea un *PULL UP*, y con eso, cuando se apriete cualquier botón el voltaje caerá hasta 0V y al soltarlo volverá a estado anterior de 5V. Así, el programa solo detectará cuando exista un cambio en el estado anterior evitando así crear filtro contra ruido o un sistema anti rebotes, estos valores se los guarda en una variable llamada flancoSubida. Esto se explica en la figura 2.18.

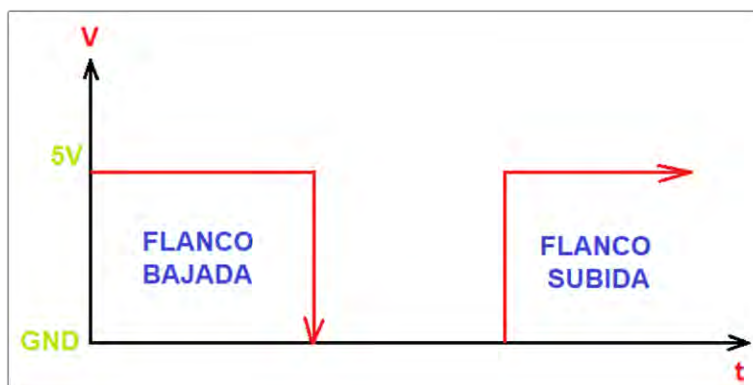


Figura 2. 18. Flanco de bajada y subida.

Fuente: Elaborado por el autor

A continuación, en la figura 2.19 se muestra cómo se escribe el código en el software de Arduino para poder controlar el flanco de subida.

```
uint8_t flancoSubida(int btn) {           // Variable para el flanco de subida
    uint8_t valor_nuevo =
        digitalRead(button[btn]);
    uint8_t result = button_state[btn]!=valor_nuevo && valor_nuevo ==1;
    button_state[btn] = valor_nuevo;
    return result;
}
```

Figura 2. 19. Rutina para controlar el flanco de subida.

Fuente: Elaborado por el autor

Para este circuito se utilizó un autómata que solo puede estar en un solo estado, es decir solo se podrá ejecutar en un solo menú a la vez y que se lo representa con un círculo. A este estado se lo llamó HOME., Cuando estemos en este estado se mostrará el mensaje programado para el MENU y no realizará ninguna tarea, simplemente va a mostrar ese mensaje.

En el mensaje MENU se va a desplegar la información del Título del proyecto, la temperatura actual medida en grados centígrados, la humedad relativa expresada en tanto por ciento, la hora y la fecha actual, como se muestra en la figura 2. 20.



Figura 2. 20. Pantalla principal de MENU

Fuente: Elaborado por el autor

Otro estado que se utiliza es SET TEMP el cual se detalla en la figura 2.21. en el cual se puede cambiar el valor de la temperatura. El cambio desde el MENU hacia el estado SET TEMP se lo realiza mediante una transición al momento en que se presiona el botón MENU.

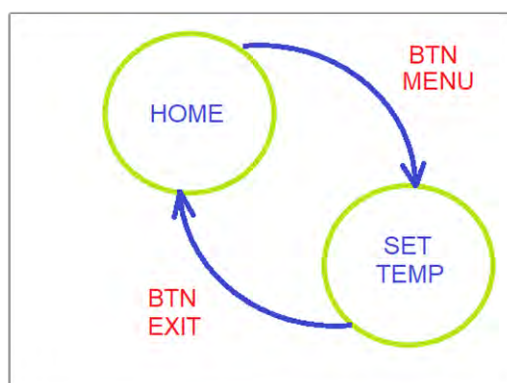


Figura 2. 21. Estado inicial de HOME y SET TEMP.

Fuente: Elaborado por el autor

En el estado SET TEMP se utilizará los botones BTN UP para subir la temperatura y BTN DOWN para bajar la temperatura, el cual se aprecia en la figura 2.22. La temperatura se podrá incrementar y decrementar en un rango programado desde los 10 °C hasta 40 °C (grados centígrados) en intervalos de 1°C activando automáticamente un Calefactor para aumentar la Temperatura y un Aire Acondicionado para disminuir la temperatura. Cuando se desee pasar al siguiente estado se deberá pulsar el botón BTN MENU y cuando se desee salir e ir a la pantalla del estado inicial debe presionar el botón BTN EXIT.

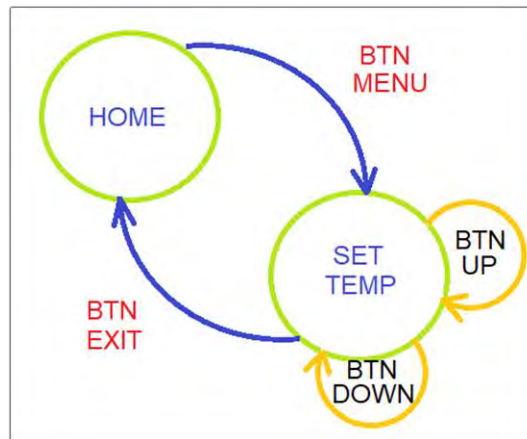


Figura 2. 22. Estado inicial SET TEMP.

Fuente: Elaborado por el autor

En el menú TEMPERATURA se despliega la opción para setear la temperatura de operación y se mostrará cómo se detalla en la figura 2.23.



Figura 2. 23. MENU de la temperatura de operación

Fuente: Elaborado por el autor

Se agregó otro estado con el cual se puede controlar la humedad relativa al cual, el estado se llama SET HUM, el cual se observa en la figura 2.24. donde se utiliza los botones BTN UP para subir la Humedad Relativa y BTN DOWN para bajar la Humedad relativa. Con este control se puede incrementar y decrementar la humedad en un rango programado desde el 40% hasta 100% en intervalos de 10%, activando automáticamente un Humidificador para aumentar la Humedad y un Deshumidificador para disminuir la humedad. Cuando se desee pasar al siguiente estado se deberá pulsar el botón BTN MENU y para salir e ir a la pantalla del estado inicial se deberá presionar el botón BTN EXIT.

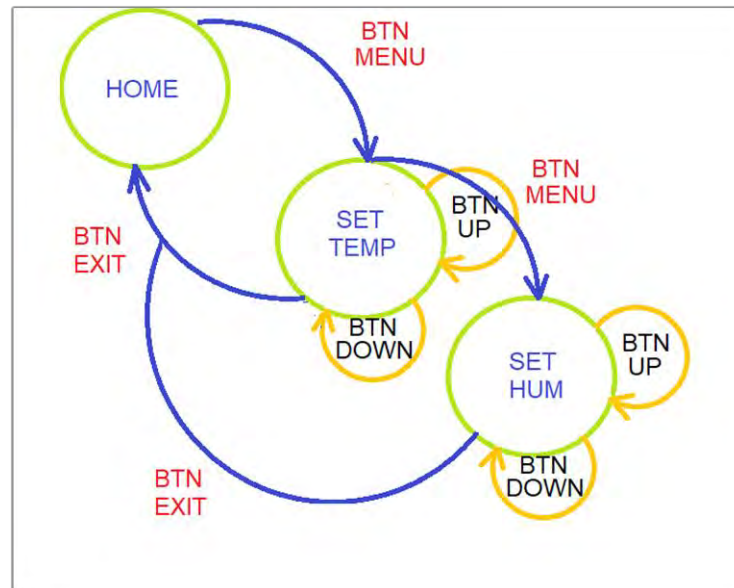


Figura 2. 24. Estado inicial SET HUM

Fuente: Elaborado por el autor

En el menú HUMEDAD se va a desplegar la opción para escoger la humedad relativa de operación y se observa en la figura 2.25.



Figura 2. 25. MENU de la humedad de operación.

Fuente: Elaborado por el autor

Después de controlar la humedad relativa se agregó otro estado para controlar la iluminación dentro del contenedor, al cual lo se lo llamó SET ILUM como se muestra en la figura 2.26. Para el encendido y apagado de los LED se utiliza los botones BTN UP y BTN DOWN respectivamente. Con este estado solo se tiene un control automático de encendido y apagado ON/OFF de las luces led diseñadas para el cultivo hidropónico. Cuando se desee

pasar al siguiente estado se deberá pulsar el botón BTN MENU y para salir e ir a la pantalla del estado inicial se presionará el botón BTN EXIT.

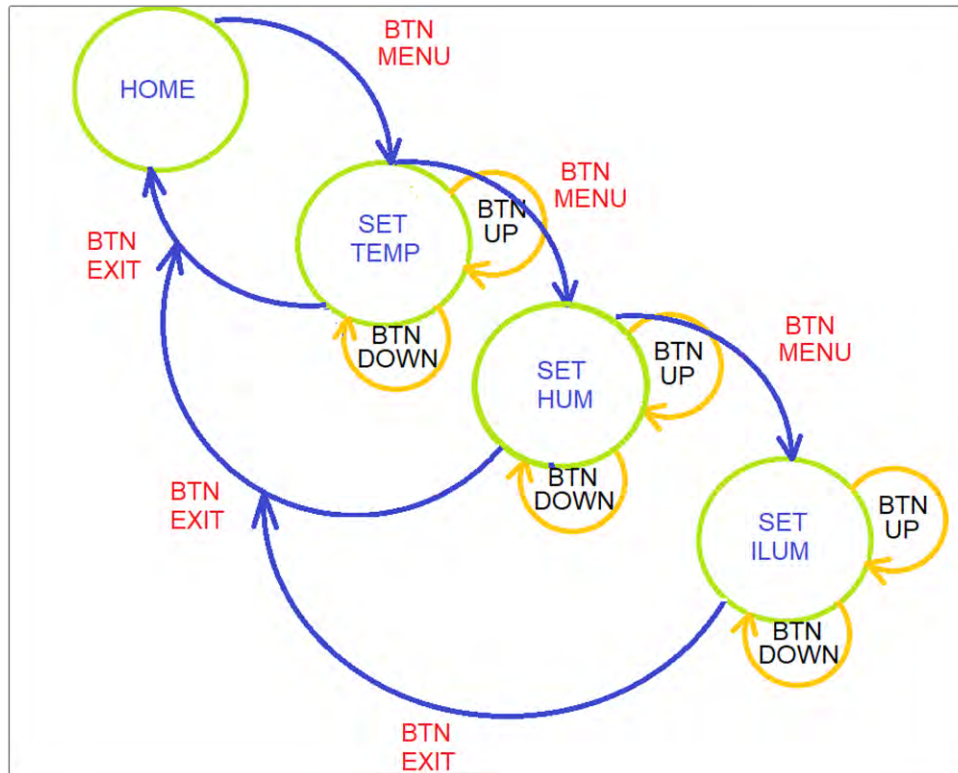


Figura 2. 26. Estado inicial SET ILIM

Fuente: Elaborado por el autor

En el menú ILUMINACION se va a desplegar la opción para encender o apagar la iluminación dentro del contenedor y se mostrara como se muestra en la figura 2.27.

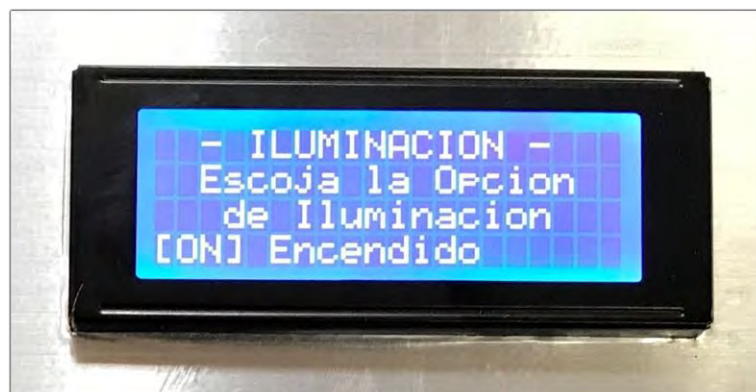


Figura 2. 27. MENU del control de la iluminación. Fuente: Elaborado por el autor

Para controlar el tiempo de riego se agregó otro estado con el nombre SET TIME que se muestra en la figura 2.28. Se utilizan los botones BTN UP para subir el tiempo de regadío y BTN DOWN para disminuir el tiempo de regadío. Con este control automático se puede incrementar y decrementar el tiempo de riego en un rango programado desde 5 minutos hasta 60 minutos en intervalos de ± 5 minutos, activando automáticamente la bomba sumergible para transportar la solución alimenticia a las plantas. Cuando se desee pasar al siguiente estado se debe pulsar el botón BTN MENU y para salir e ir a la pantalla del estado inicial se deberá presionar el botón BTN EXIT.

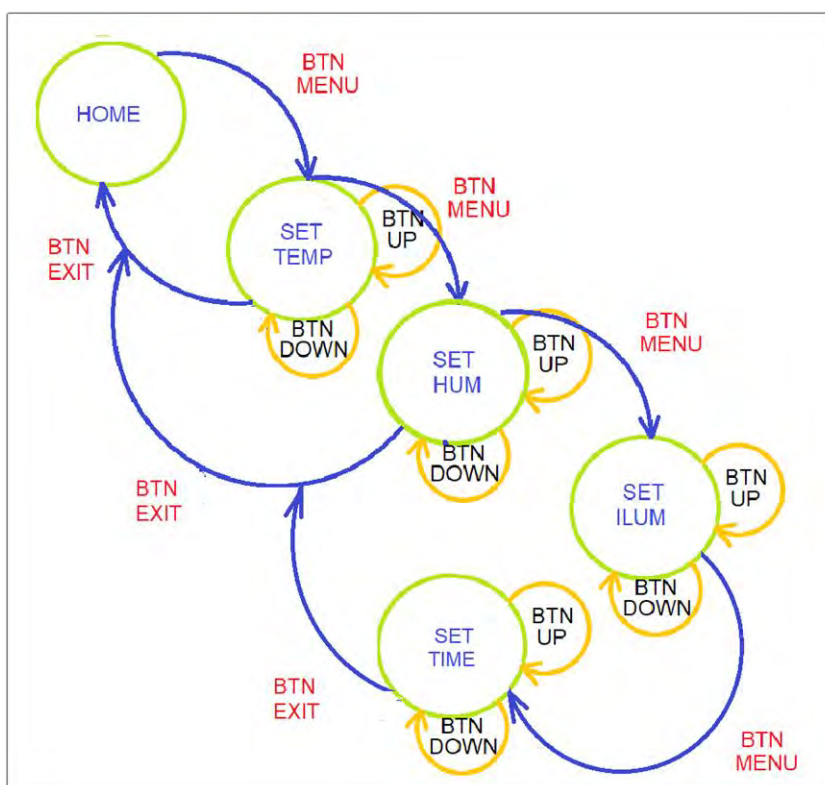


Figura 2. 28. Estado inicial SET TIME

Fuente: Elaborado por el autor

En el menú TIEMPO DE RIEGO se va a desplegar la opción para seleccionar el tiempo que necesario para el riego de los cultivos dentro del contenedor y se detalla en la figura 2.29.



Figura 2. 29. MENU del control del tiempo de riego.

Fuente: Elaborado por el autor

Finalmente, se agregó otro estado para controlar el intervalo de riego, el estado tiene como nombre SET INTER, se utiliza los botones BTN UP para subir el tiempo de regadío y BTN DOWN para disminuir el tiempo de regadío. Con este control automático se puede incrementar y decrementar el tiempo de riego en un rango programado desde 1 hora hasta 24 horas en intervalos de ± 1 hora, activando automáticamente la bomba sumergible para transporta la solución alimenticia a las plantas. En la figura 2.30 se muestra la maquina de estados completa con todos los estados.

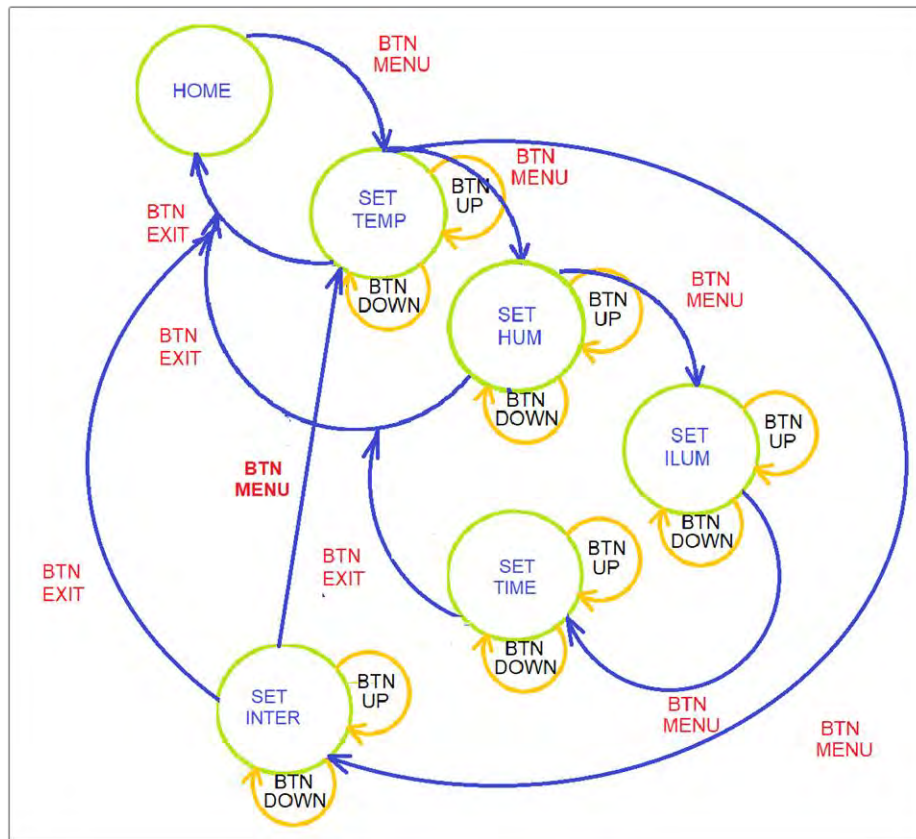


Figura 2. 30. Máquina de estados completa.

Fuente: Elaborado por el autor

En el menú HORA DE RIEGO se va a desplegar la opción para seleccionar el intervalo de tiempo que se necesite para el riego de los cultivos dentro del contenedor y se mostrara como se detalla en la figura 2.31.



Figura 2. 31. MENU del control del intervalo de tiempo para riego.

Fuente: Elaborado por el autor

Como ya no existen más estados cuando se pulse el botón BTN MENU el programa direccionará nuevamente al estado SET TEMP y así seguirá constantemente en un lazo

cerrado completado la máquina de estados finitos. Cuando se desee salir e ir a la pantalla del estado inicial se deberá pulsar el botón BTN EXIT.

Capítulo 3

Diseño del Hardware

En este Capítulo se detalla el diseño del hardware, el desarrollo, la simulación, la construcción de la placa PCB y el ensamblaje del circuito.

Una vez terminada la fase de diseño del software se procedió a la implementación del circuito y lo primero que fue realizado fue el diseño del diagrama esquemático a través del *software* de Proteus. Esta herramienta informática se utilizó para realizar la simulación de un control de temperatura y humedad, el diagrama esquemático y la *PCB*.

La simulación del control de temperatura y humedad se lo presenta en la figura 3.32. de la cual se puede observar la simulación de un sensor DHT22 conectado a un módulo Arduino Uno mostrando los valores de humedad relativa y temperatura.

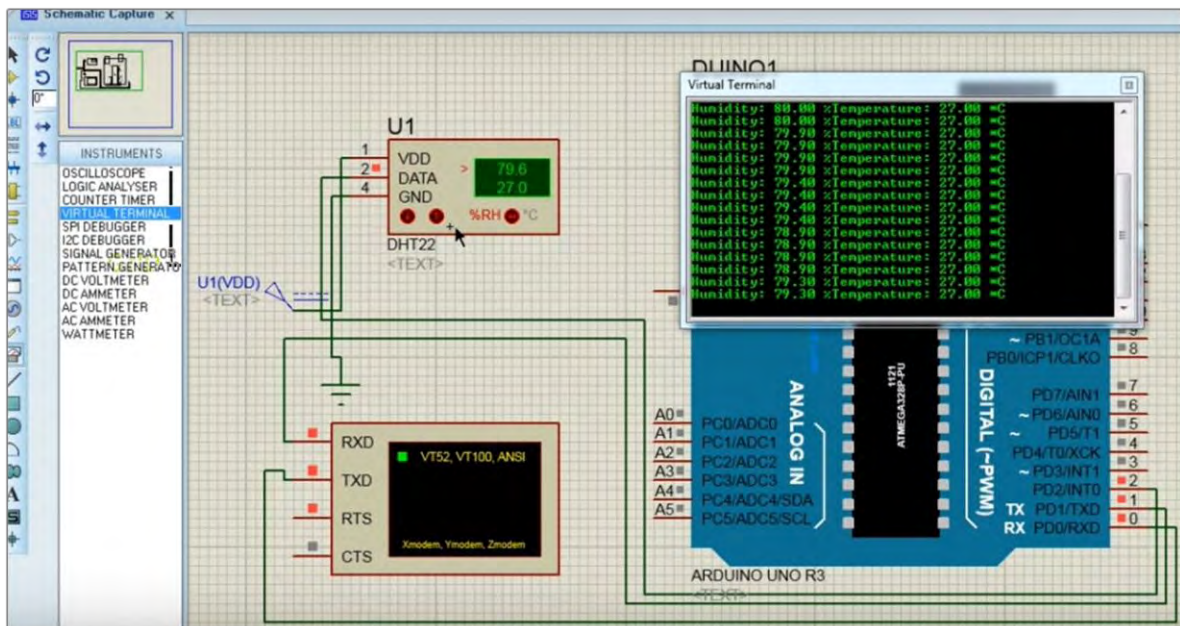


Figura 3. 32. Simulación en Proteus.

Fuente: Elaborado por el autor

Luego de realizar varias simulaciones y verificar que el circuito funciona correctamente se procedió a armar en un *protoboard* el circuito para el control automático con todos los elementos necesarios para empezar a realizar las pruebas y verificar el funcionamiento. En la figura 3.33. se observa el circuito de prueba en protoboard.



Figura 3. 33. Circuito de prueba en protoboard.

Fuente: Elaborado por el autor

3.1. Desarrollo

Una vez que se verificó que el sistema funciona en el *protoboard*, se procedió a crear la placa *PCB* siguiendo las reglas establecidas para la elaboración de circuitos impresos entre las cuales se establece que las pistas deben ser lo más cortas y directas posibles para evitar que la impedancia y otros efectos parásitos pueden limitar el rendimiento del sistema.

Para el ancho de las pistas se utilizó la norma ANSI, que define los requisitos generales para el diseño de PCB y algunos aspectos de ensamble de tarjetas electrónicas, principios y recomendaciones, con lo cual se realizó un cálculo para que estas puedan soportar una corriente de hasta 1 Amperio sin generar daño de las pistas y también que no se genere un incremento de temperatura más allá de 10°C y de acuerdo con los cálculos realizados se optó por un grosor de pista de 0.608 mm. En la figura 3.34 se detalla el cálculo del ancho de una Pista PCB con respecto a la corriente.

ANSI PCB Cálculo del ancho de una Pista PCB con respecto a la corriente “AMPERES”

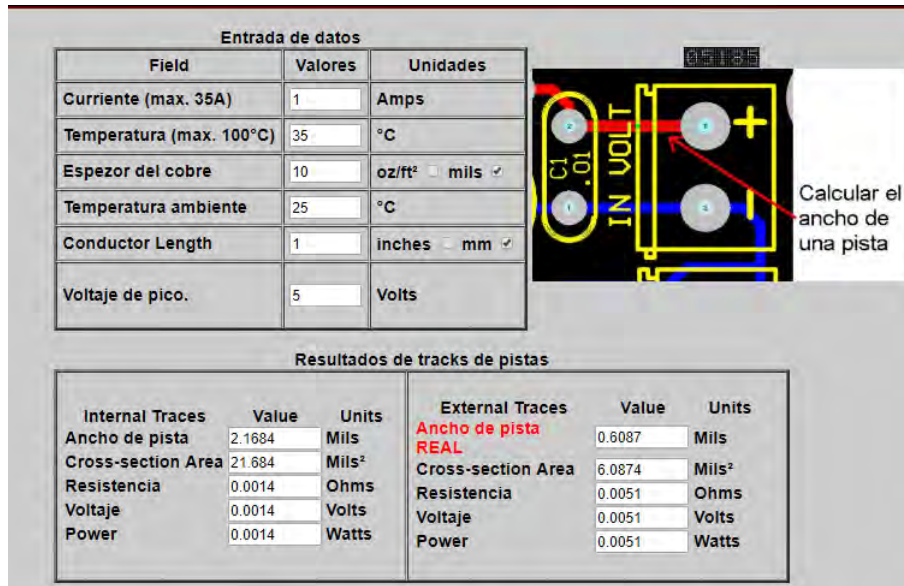


Figura 3. 34. Cálculo del ancho de banda de una pista PCB

Fuente: Elaborado por el autor

Una vez que se realizaron todos los cálculos de diseño se procedió a la elaboración de la placa PCB en el software Proteus, la cual se muestra en la figura 3.35.

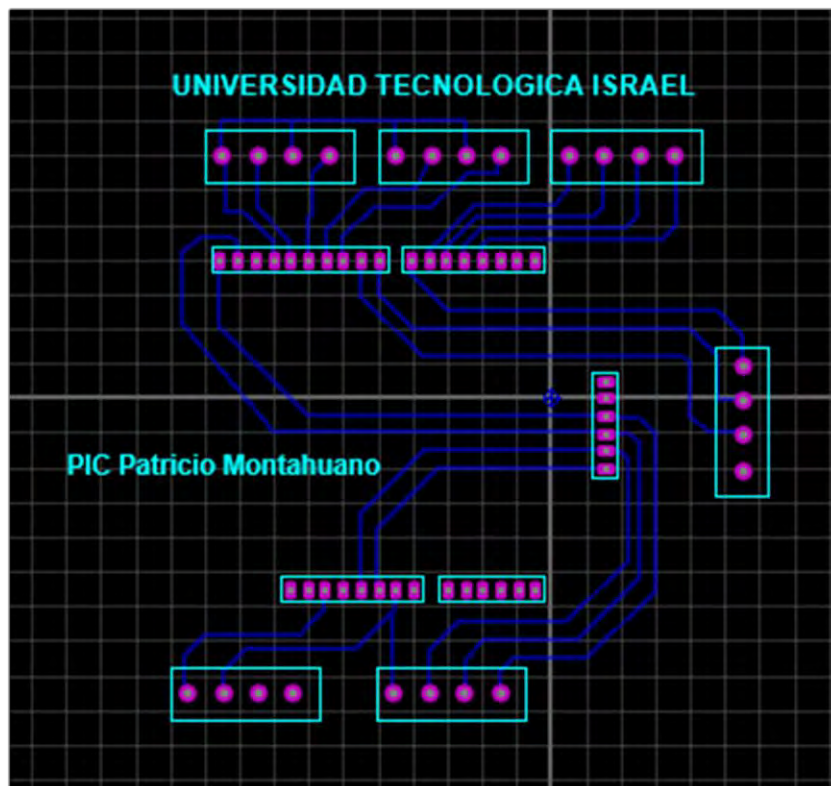


Figura 3. 35. Diseño de la PCB

Fuente: Elaborado por el autor

Una vez que se diseñó la placa *PCB* en el software Proteus, se realizó la impresión del diseño en un material termo-sensible para transferir a través de calor a la placa de cobre donde quemó con un ácido (percloruro férrico) para eliminar el cobre excedente del circuito impreso en la baquelita y las perforaciones fueron realizadas con brocas pequeñas hechas de material de carburo tungsteno.

Para este circuito impreso se utilizaron los estándares IPC-2220 que son aquellos que regulan el diseño, ensamblado y control de calidad de los circuitos impresos *PCB*, como se puede apreciar en la figura 3.36.

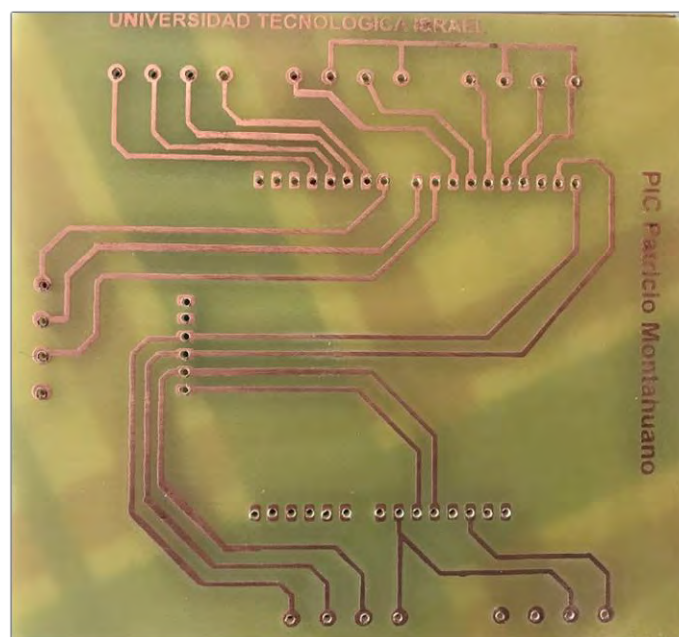


Figura 3. 36. PCB (Placa de circuito impreso).

Fuente: Elaborado por el autor

Para el montaje de los componentes en la *PCB*, se fijaron los elementos y luego se soldaron a los pines con estaño, como se aprecia en las figuras 3.37. y 3.38.

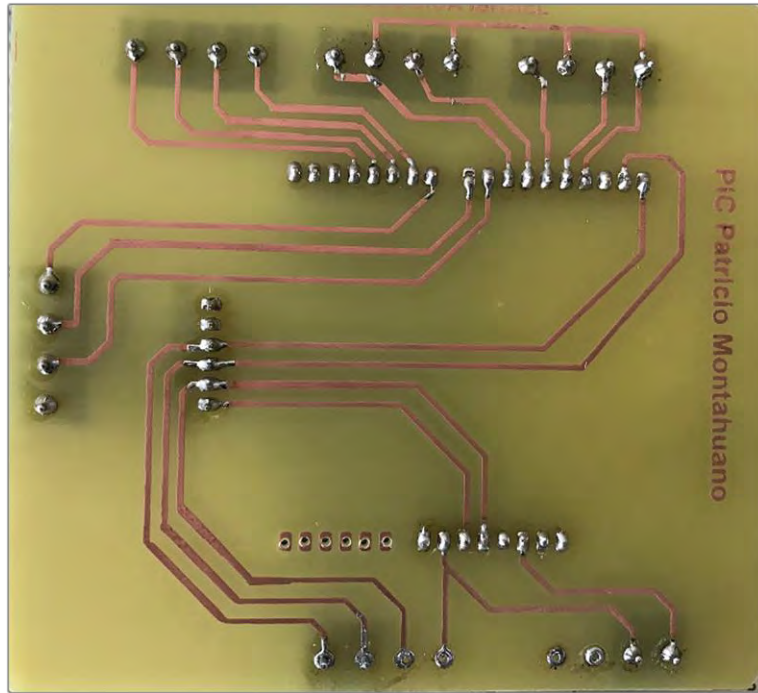


Figura 3. 37. Soldadura de elementos en la PCB (Placa de circuito impreso). Fuente: Elaborado por el autor

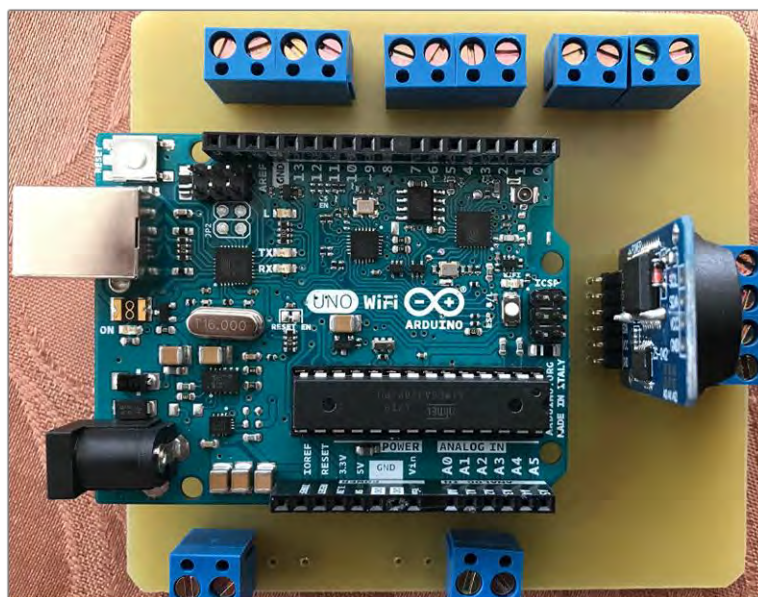


Figura 3. 38. Montaje de elementos en la PCB (Placa de circuito impreso). Fuente: Elaborado por el autor

Una vez que concluimos con el ensamblaje de los componentes internos, procedemos a montar los circuitos en una caja acrílica transparente, la cual servirá para la presentación final del proyecto como se observa en la figura 3.39.

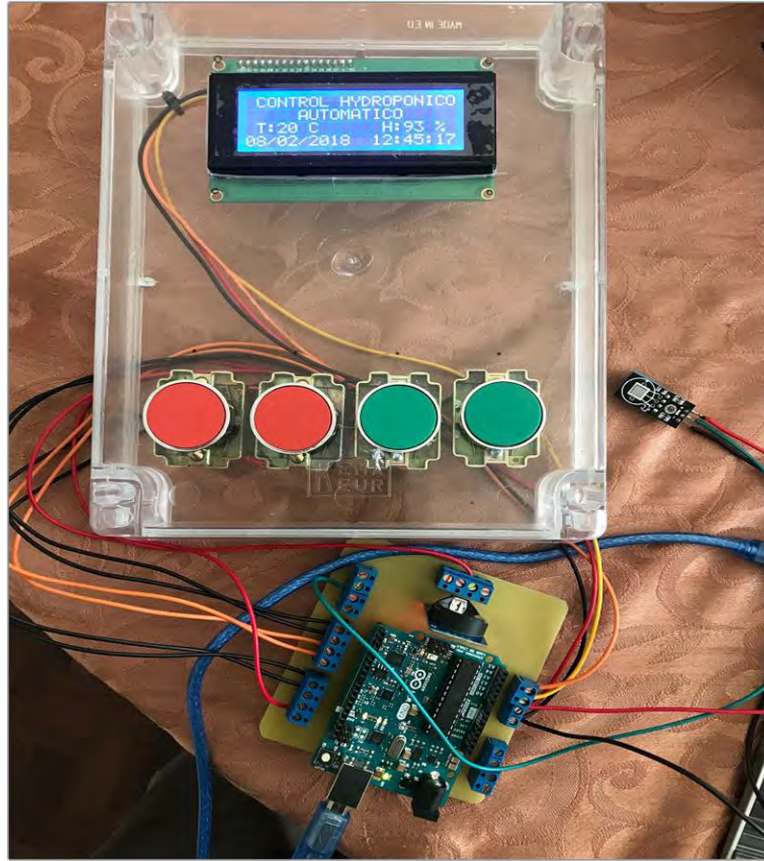


Figura 3. 39. Montaje de los pulsadores en la placa acrílica.

Fuente: Elaborado por el autor

Luego que se concluyó con la parte electrónica del circuito, se procedió al diseño, armado, construcción y montaje del contenedor de carga a escala, donde se colocaron los elementos de control, calefactor, aire acondicionado, humidificador, deshumificador como se muestran en las figuras 3.40. y 3.41.



Figura 3. 40. Armado del contenedor a escala. Fuente: Elaborado por el autor



Figura 3. 41. Proyecto finalizado. Fuente: Elaborado por el autor

CAPÍTULO 4

PRUEBAS Y ANÁLISIS DE RESULTADOS

En este Capítulo se detallan las pruebas realizadas y el análisis de los resultados del circuito de control automático para cultivos hidropónicos.

4.1 Pruebas

La primera parte de las pruebas, se la realizó en el laboratorio cuando todo el circuito de control estaba ensamblado y consistió en confirmar que el módulo de automatización encienda la pantalla *LCD* mostrando el menú principal donde se observe la hora, la fecha, los valores de la temperatura y humedad relativa actual. También se verificó que los botones MENU, SALIDA, ARRIBA y ABAJO funcionen de manera correcta y que mediante su uso se pueda ingresar a los submenús y modificar los parámetros de control. En la figura 4.42. se observa el circuito funcionando una vez ensamblado.



Figura 4. 42. Prueba de funcionamiento del circuito una vez ensamblado.

Fuente: Elaborado por el autor

Luego se comprobaron que los parámetros de temperatura y humedad relativa medidos por el circuito estaban acorde a los valores reales. Como se puede observar en la figura 4.43, la temperatura medida es 12 grados centígrados y la humedad relativa es del 67%.

En la figura 4.43. se puede apreciar los datos medidos con los actuales, los cuales se comparados son aproximadamente los mismos. Ya que al momento de tomar la muestra la temperatura era de 12 grados centígrados y la humedad relativa era del 50%, con lo que se pudo comprobar que las mediciones del circuito eran bastante aproximadas a los valores reales.



Figura 4. 43. Comparación de valores medidos por el circuito.

Fuente: Elaborado por el autor

Una vez concluido el funcionamiento del circuito se puso a funcionar la etapa de potencia conectando las luces LED de crecimiento como se muestra en las figuras 4.44 y 4.45.



Figura 4. 44. Luces LED para crecimiento.

Fuente: Elaborado por el autor



Figura 4. 45. Planta creciendo con luz LED.

Fuente: Elaborado por el autor

También se puso en funcionamiento el circuito para probar la etapa de potencia conectando la bomba de agua como se muestra en la figura 4.46. y también el prototipo del aire acondicionado en la figura 4.47.



Figura 4. 46. Prueba de funcionamiento de la bomba de agua.

Fuente: Elaborado por el autor

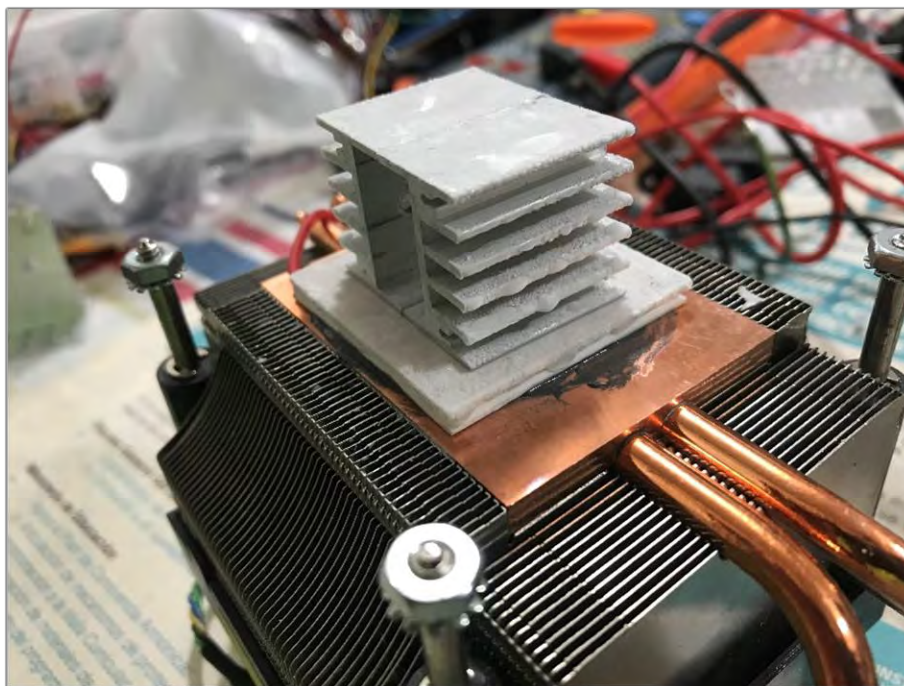


Figura 4. 47. Prueba de funcionamiento del aire acondicionado.

Fuente: Elaborado por el autor

La segunda parte de las pruebas se las realizó en el contenedor de carga a escala donde se sembraron varios tipos de hortalizas y se fue monitoreando por varios meses su crecimiento y el funcionamiento del control automático como se observa en las figuras 4.48. y 4.49.



Figura 4. 48. Crecimiento de hortalizas en ambiente controlado.

Fuente: Elaborado por el autor



Figura 4. 49. Plantas cultivadas con control automático

Fuente: Elaborado por el autor

4.2. Análisis de resultados

Al analizar la evolución de la hidroponía en el Ecuador, se puede observar que existen grandes deficiencias y limitaciones tecnológicas para la implementación de este conocimiento hasta la actualidad, ya que en la actualidad la inversión es mucho más costosa que para un cultivo tradicional. (Ministerio de agricultura y ganadería, 2018)

Luego de verificar las múltiples ventajas que tiene en utilizar en este método de cultivo con control automático para cultivos hidropónicos, analizar la mayoría de los factores que son los esenciales y que se deben tener en cuenta para el crecimiento de la planta, como el cuidado de su temperatura, humedad, la iluminación y la nutrición, se les debe brindar la mayor prioridad a la hora de automatizar ya que gracias a ellos se genera la culminación con éxito de los cultivos y adicionalmente hay que tomar en cuenta también que existen otros factores menos importantes; tales como el control del pH, el control de la temperatura de la solución nutritiva, el nivel de la solución, etc. que también pueden ser mejorados.

Este sistema de control automático tiene gran factibilidad para ser implementado en cualquier invernadero, el cual puede ser adaptado para poder desarrollar el proceso deseado, lo cual agregaría un mayor valor a la producción y permitiría llevar al máximo el aprovechamiento del cultivo utilizando la tecnología de punta para automatizar el sistema.

CONCLUSIONES

- Se definieron los parámetros de control de temperatura, humedad, iluminación nutrición y riego para el sistema de automatización, los cuales fueron controlados de manera satisfactoria a través del sistema de automatización microprocesado cumpliendo con las expectativas planteadas.
- Se diseñó el sistema de automatización de un contenedor de carga para el control de clima y crecimiento de cultivos hidropónicos según los parámetros establecidos en el objetivo general, haciendo que sea muy versátil ya que se puede adaptar a la mayoría de los cultivos por el amplio rango en su funcionamiento.
- Se desarrolló un algoritmo de control y automatización que al ser un sistema autónomo de precisión, permitió manejar todos los parámetros pre-establecidos satisfactoriamente y que los equipos de control funcionen adecuadamente de acuerdo a la programación requerida.
- Se construyó el sistema automatizado y circuito electrónico para controlar temperatura, humedad, nutrición, riego e iluminación tomando en cuenta que son los factores más importantes para el crecimiento para cultivos hidropónicos, lo cual genero el crecimiento adecuado de las plantas.
- Se realizaron las pruebas de funcionamiento y validación de resultados para cultivos hidropónicos y se concluye que el sistema de automatización funciona adecuadamente para los cultivos hidropónicos y podría utilizarse a gran escala ya que los resultados de crecimiento fueron satisfactorios.
- El manejo de los cultivos hidropónicos automatizados en un contenedor permite un control inteligente de las condiciones ambientales, la iluminación y los nutrientes, los cuales resultan inmanejables en los cultivos de tierra, logrando así una mayor producción y mejorando la calidad de los cultivos.

- De datos disponibles de investigaciones de la FAO (<http://www.fao.org/3/a-ah501s.pdf>) se deduce que, si se agrega el sistema de automatización en un contenedor, los costos de inversión inicial son casi cuatro veces mayores que el precio al respecto de un cultivo tradicional en tierra. Pero viendo la realidad y la producción obtenida, se puede predecir que en aproximadamente 3 años se podría recuperar la inversión inicial.

El resultado de este proyecto demuestra que si es posible controlar los parámetros elegidos en la automatización para mejorar la calidad de los cultivos

RECOMENDACIONES

- En este proyecto se utilizaron varios equipos externos como aire acondicionado, calefactor, humidificador y deshumificador para controlar los parámetros programados, pero hay que tener cuidado al momento de calcular las capacidades de acuerdo con el área de cobertura ya que algunas veces es insuficiente y otras sobrepasa los valores requeridos. Mediante las pruebas realizadas en el contenedor se podría escalar a un desarrollo comercial para aplicaciones en el mundo real.
- Otra recomendación es incluir en el diseño un módulo con un reloj en tiempo real *RTC* (*Real Time Clock*) ya que para crear los horarios de los tiempos y los intervalos de riego la programación se complica muchísimo ya que existen muchas variantes de acuerdo con el tipo de cultivo y mediante el *RTC* se simplifica significativamente la programación del microcontrolador.
- Es una buena práctica, analizar y escoger el tipo de microcontrolador que se necesite de acuerdo con los requerimientos del proyecto ya que en el mercado actual existen muchos tipos de dispositivos y elegir el más adecuado simplifica mucho el diseño, por ejemplo: en este proyecto se utilizó un módulo Arduino que tenía el módulo Wifi incorporado, lo que generó que la implementación sea mucho más sencilla y en caso de fallo sería mucho más fácil realizar el cambio de un solo componente.
- Hay que tener en cuenta que cuando se realice la programación del módulo LCD a través del módulo I²C, para que no existan errores cuando se utilice un display de 16x2 en la programación se debe colocar el número hexadecimal 0x3f y para el display de 20x4 se debe colocar el número hexadecimal 0x27.

Bibliografía

- Adafruit.com. (2018). *Datasheet digital relative humidity & temperature sensorAM2302/DHT22*. Retrieved from <https://cdn-shop.adafruit.com/datasheets/Digital+humidity+and+temperature+sensor+AM2302.pdf>
- Arduino. (2016, noviembre 2). *Arduino Uno Wifi*. Retrieved from <https://store.arduino.cc/usa/arduino-uno-wifi>
- Arnon, H. D. (1938). *The water culture method for growing plants without soil. Calif. Agr. Expt. Sta. Cir. 347, 39p.*
- Castilla, W. S. (1995). *Protected cultivation of horticultural crops worldwide. Hort technology 5(1), 6-23.*
- Collins, J. M. (1985). *Hydroponic vegetable production. Hort. Review 7, 483-558.*
- Dalrymple DG. (1973). *A global review of greenhouse food production. USDA.*
- Graves CJ. (1983). *The nutrient film technique. Horticultural Review. 5: 1-44.*
- GW, W. (1979). *Nutrient Film Technique. Grower Bulletin 5. Glasshouse Crops Research Institute. Littlehampton, England.*
- Hawthorne Gardening. (2017). *Flo-n-Gro Grow Systems*. Retrieved from <http://www.flo-n-gro.net/page/five-methods-of-hydroponics/>
- Jensen, M. H. (2017). *Hydroponics Worldwide - A technical Overview, University of Arizona. Tucson, Arizona.*
- Massantini F. (1976). *Floating hydroponics; A method of soilless culture, 91-98.*
- Maxim Integrated. (2014). *Datasheet DS3231 Extremely Accurate I2C-Integrated*. Retrieved from *Datasheet DS3231 Extremely Accurate I2C-Integrated:* <https://datasheets.maximintegrated.com/en/ds/DS3231.pdf>
- Ministerio de agricultura y ganadería. (2018, agosto 16). *Lechuga se produce de forma hidropónica*. Retrieved from <https://www.agricultura.gob.ec/lechuga-se-produce-de-forma-hidroponica/>
- Naylamp Mechatronics SAC. (2016). *Naylamp Mechatronics*. Retrieved from https://www.naylampmechatronics.com/blog/35_Tutorial--LCD-con-I2C-controla-un-LCD-con-so.html
- Opencircuit. (2018). *Datasheet I2C 1602 Serial LCD Module*. Retrieved from <https://opencircuit.nl/ProductInfo/1000061/I2C-LCD-interface.pdf>
- Organizacion de las Naciones Unidas para la alimentación y la agricultura. (2017). *FAO*. Retrieved from <http://www.fao.org/home/es/>
- OSRAM Opto Semiconductors. (2018). *LEDs for horticultural lighting applications*. Regensburg.
- Rduinostar.com. (2015). *Características DHT22*. Retrieved from <http://rduinostar.com/documentacion/datasheets/dht22-caracteristicas-am2302/>

Vishay. (2016). *Datasheet LCD-020N004L LCD- 0 x 4 Character LCD*. Retrieved from <https://www.vishay.com/docs/37314/lcd020n004l.pdf>

Anexo 1. Código Fuente:

ANEXO 1

CODIGO FUENTE DEL ALGORITMO

```
#include <LiquidCrystal_I2C.h> //libreria del I2C para display
#include <DS3231.h> // Libreria del RTC
#include <Wire.h> // Libreria del I2C
#include <RTCLib.h> // Libreria del RTC
#include <DHT.h> // Libreria del Sensor Humedad y Temperatura

RTC_DS1307 RTC;
LiquidCrystal_I2C lcd(0x27,21,4); // Ponemos la direccion LCD 0x3f para display 16,2 y 0x27 para
display 21/4
DS3231 rtc(SDA, SCL); // Escojemos modelo del RTC DS3132

#define DHTPIN 5 // Definimos el PIN8 digital donde se conecta el sensor
#define DHTTYPE DHT11 // Definimos el tipo de sensor DHT22
DHT dht(DHTPIN, DHTTYPE); // Inicializamos el sensor DHT22

#define S_HOME 0 // Defino los estados
#define S_SET_TEMP 1
#define S_SET_HUME 2
#define S_SET_LIGHT 3
#define S_SET_NUTRI 4
#define S_SET_INTER 5

uint8_t estado = S_HOME; // Variable estado actual

#define BTN_MENU 0 // Defino los Botones
#define BTN_EXIT 1
#define BTN_UP 2
#define BTN_DOWN 3

uint8_t button[4] = { // Variable de los botones
  13, // PIN 13 Boton Menu
  12, // PIN 12 Boton Exit
  11, // PIN 11 Boton Up
  10, // PIN 10 Boton Down
};

uint8_t button_state[4]; // Variable estado de los botones

// Guarda el valor actual de la varible para todos los estados //
uint8_t t; // Variable para la Temperatura
uint8_t h; // Variable para la Humedad
uint8_t l = 0; // Variable control Iluminacion
uint8_t hum = 80; // variable para comparar Humedad
uint8_t tem = 20; // variable para comparar Temperatura
uint8_t minuto = 5; // variable para Tiempo Riego (MINUTOS)
uint8_t hora = 1; // variable para Intervalo de Riego (HORAS)

// Realiza la funcion Millis para controlar los tiempos de Riego e Intervalo//
const int ledPin = 7; // El Pin 7 esta designado para la BOMBA
int ledState = LOW; // El estado inicial de la bomba es apagado
long previousMillis = 0; // Guarda el ultimo estado de la bomba que fue actualizado
long intervalOn = 10000; // Tiempo del Intervalo ON Tiempo Riego (MINUTOS)[multiplicar *
60] 60000
long intervalOff = 10000; // Tiempo del Intervalo OFF Intervalo de Riego (HORAS) [multiplicar
*3600] 3600000
```

```

uint8_t flancoSubida(int btn) { // Variable para el flanco de subida
  uint8_t valor_nuevo =
    digitalRead(button[btn]);
  uint8_t result = button_state[btn]!=valor_nuevo && valor_nuevo ==1;
  button_state[btn] = valor_nuevo;
  return result;
}
void setup() {

Serial.begin(9600);

  pinMode(button[BTN_MENU], INPUT_PULLUP); // Indicamos que las entrada boton MENU es
  PULLUP
  pinMode(button[BTN_EXIT], INPUT_PULLUP); // Indicamos que las entrada boton EXIT es
  PULLUP
  pinMode(button[BTN_UP], INPUT_PULLUP); // Indicamos que las entrada boton UP es PULLUP
  pinMode(button[BTN_DOWN], INPUT_PULLUP); // Indicamos que las entrada boton DOWN es
  PULLUP
  pinMode(3,OUTPUT); // Indicamos que el PIN 3 es salida Controla CALEFACTOR
  pinMode(4,OUTPUT); // Indicamos que el PIN 4 es salida Controla AIRE
  ACONDICIONADO
  pinMode(6,OUTPUT); // Indicamos que el PIN 6 es salida Controla HUMIFICADOR
  pinMode(9,OUTPUT); // Indicamos que el PIN 9 es salida Controla
  DESHUMIFICADOR
  pinMode(8,OUTPUT); // Indicamos que el PIN 8 es salida Controla Iluminacion
  digitalWrite (8,HIGH); // Indicamos que por default la luz va a estar encendida
  pinMode(ledPin, OUTPUT); // Indicamos que el PIN 7 es salida Controla de la Bomba

  button_state[0] = HIGH;
  button_state[1] = HIGH;
  button_state[2] = HIGH;
  button_state[3] = HIGH;

  lcd.init(); // Inicializamos el LCD
  lcd.backlight(); // Inicializamos el backlight LCD
  rtc.begin(); // Inicializamos el RTC
  dht.begin(); // Inicializamos el sensor DHT
  printHome();

  // Ajuste de la hora y fecha de la Computadora//
  //RTC.adjust(DateTime(__DATE__, __TIME__)); // Lee la fecha y hora del PC (Solo en la primera
  carga)
  // el anterior se usa solo en la configuracion inicial luego se pone como comentario
  // y se vuelve a cargar el programa sin esa linea.
}

/** Inicio de funciones dibujada**/

void printHome() { // Pantalla Principal LCD
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" CONTROL HYDROPONICO"); // Texto Primera Linea
  lcd.setCursor(0,1);
  lcd.print (" AUTOMATICO"); // Texto Tercera Linea
  lcd.setCursor(0,2);
}
void printTemp(){ // Pantala temperatura LCD
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" - TEMPERATURA -");
}

```

```

    lcd.setCursor(0,1);
    lcd.print(" Escoja Temperatura");
    lcd.setCursor(0,2);
    lcd.print(" de Operacion");
}
void printHume(){                                // Pantala Humedad LCD
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" - HUMEDAD -");
    lcd.setCursor(0,1);
    lcd.print("Escoja el Porcentaje");
    lcd.setCursor(0,2);
    lcd.print(" de Humedad");
}
void printLight(){                               // Pantala iluminacion LCD
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" - ILUMINACION -");
    lcd.setCursor(0,1);
    lcd.print(" Escoja la Opcion");
    lcd.setCursor(0,2);
    lcd.print(" de Iluminacion");
}
void printNutri(){                              // Pantala Tiempo de Riego LCD
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" - TIEMPO RIEGO -");
    lcd.setCursor(0,1);
    lcd.print(" Escoja la Duracion");
    lcd.setCursor(0,2);
    lcd.print(" del Riego");
}
void printInter(){                              // Pantala Intervalo de Riego LCD
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" - HORA DE RIEGO -");
    lcd.setCursor(0,1);
    lcd.print(" Escoja el Intervalo");
    lcd.setCursor(0,2);
    lcd.print(" de Riego");
}
void printInfoT(){                              // Pantala secundaria LCD Temperatura
    lcd.setCursor(0,3);
    lcd.print(" TEMP: ");
    lcd.print(tem);
    lcd.print(" c");
}
void printInfoH(){                              // Pantala secundaria LCD Humedad
    lcd.setCursor(0,3);
    lcd.print(" HUME: ");
    lcd.print(hum);
    lcd.print(" %");
}
void printInfoLOn(){                            // Pantala secundaria LCD Iluminacion ON
    lcd.setCursor(0,3);
    lcd.print("[ON] Encendido ");
}
void printInfoLOff(){                           // Pantala secundaria LCD Iluminacion OFF
    lcd.setCursor(0,3);
    lcd.print(" Apagado [OFF]");
}

```

```

}
void printInfoN(){           // Pantala secundaria LCD Tiempo Riego
  lcd.setCursor(0,3);
  lcd.print("  ");
  lcd.print(minuto);
  lcd.print(" MINUTOS ");
}
void printInfoI(){          // Pantala secundaria LCD Intervalo Riego
  lcd.setCursor(0,3);
  lcd.print("  ");
  lcd.print(hora);
  lcd.print(" HORA(S)");
}

}
void printClock(){         // Pantala del Reloj
  lcd.setCursor(0,2);
  lcd.print(" T:");
  lcd.print(t);
  lcd.print(" C  ");
  lcd.print(" H:");
  lcd.print(h);
  lcd.print("%");
  lcd.setCursor(0,3);
  lcd.print(rtc.getDateStr(0,1,'/'));
  lcd.print(" ");
  lcd.print(rtc.getTimeStr());
}

void loop() {

  Serial.println(" Grados C \t");

  Serial.println(t);
  Serial.println(h);
  Serial.println();
  Serial.println(rtc.getTimeStr());

  // Control de los tiempos del Sistema//
  unsigned long currentMillis = millis();
  if (ledState == LOW) {
    if(currentMillis - previousMillis > intervalOff*hora) {
      previousMillis = currentMillis;
      ledState = HIGH;
    }
  } else {
    if(currentMillis - previousMillis > intervalOn*minuto) {
      previousMillis = currentMillis;
      ledState = LOW;
    }
  }
}
digitalWrite(ledPin, ledState);

  t = dht.readTemperature();           // Leemos la temperatura en grados centígrados (por
defecto)
  h = dht.readHumidity();              // Leemos el Porcentaje de Humedad (por defecto %)

  // Control de Variable de la Temperatura//
  if(t+4<tem){

```

```

    digitalWrite(3,LOW);                // Activacion del Pin 3 si la temp es mayor
(CALEFACTOR)
    }else{
    digitalWrite(3,HIGH);                // Desactivacion del Pin 3 si la temp es menor
(CALEFACTOR)
    }
    if (t>tem){                          // EL 5 ES EL INTERVALO DE TIEMPO DE ACTICACION
AC/CALEF
    digitalWrite(4,LOW);                // Desactivacion del Pin 4 si la temp es mayor (AIRE
ACONDICIONADR)
    } else {
    digitalWrite(4,HIGH);                // Activacion del Pin 4 si la temp es mayor (AIRE
ACONDICIONADO)
    }

// Control de Variable de la Humedad//

    if (h+10<hum){
    digitalWrite(6,LOW);                // Activacion del Pin 6 si la Humedad es menor
(HUMIDIFICADOR)
    } else {
    digitalWrite(6,HIGH);                // Desactivacion del Pin 6 si la temp es mayor
(HUMIDIFICADOR)
    }
    if (h>hum){                          // EL 10 ES EL INTERVALO DE TIEMPO DE ACTICACION
(HUMI/DESHUMI)
    digitalWrite(9,LOW);                // Desactivacion del Pin 9 si la temp es mayor
(DESHUMIDIFICADOR)

    } else {
    digitalWrite(9,HIGH);                // Activacion del Pin 9 si la temp es mayor
(DESHUMIDIFICADOR)
    }

switch (estado){                        // Solo ejecuta la variable con cambio de estado
case S_HOME:                            /** Inicio Home**/
    if(flancoSubida(BTN_MENU)){ estado = S_SET_TEMP;
    printTemp();
    printInfoT();
    break;
    }
    printClock();
    break;                               /*Fin Home*/

case S_SET_TEMP: /**Inicio Set TEMP**/

    if(flancoSubida(BTN_MENU)){
    estado = S_SET_HUME;
    printHume();
    printInfoH();
    break;
    }

    if(flancoSubida(BTN_EXIT)){
    estado = S_HOME;
    printHome();
    break;
    }

```

```

if(flancoSubida(BTN_UP)){

if(tem<40){
    tem++;
    // Intervalo de la temperatura ( 5 grados)//

} else {
    tem = 0;
}
printTemp();
printInfoT();
break;
}
if(flancoSubida(BTN_DOWN)){
    if(tem>0){
        tem--;
        // Intervalo de la temperatura ( 5 grados)//
    } else {
        tem = 40;
    }
    printTemp();
    printInfoT();
    break;
}
//printClock();
break; /** Fin set TEMP**/

case S_SET_HUME: /**Inicio Set HUME**/

    if(flancoSubida(BTN_MENU)){
        estado = S_SET_LIGHT;
        printLight();
        printInfoLOn();
        break;
    }

    if(flancoSubida(BTN_EXIT)){
        estado = S_HOME;
        printHome();
        break;
    }
    if(flancoSubida(BTN_UP)){
        if(hum<100){
            hum+=10;
            // Intervalo de la Humedad ( 10%)//
        } else {
            hum = 0;
        }
        printHume();
        printInfoH();
        break;
    }

    if(flancoSubida(BTN_DOWN)){
        if(hum>0){
            hum-=10;
            // Intervalo de la Humedad ( 10%)//
        } else {
            hum = 100;
        }
        printHume();
        printInfoH();
        break;
    }
}

```

```

break; /** Fin set HUME**/

case S_SET_LIGHT: /**Inicio Set LIGHT**/

    if(flancoSubida(BTN_MENU)){
        estado = S_SET_NUTRI;
        printNutri();
        printInfoN();
        break;
    }
    if(flancoSubida(BTN_EXIT)){
        estado = S_HOME;
        printHome();
        break;
    }
    if(flancoSubida(BTN_UP)){
        if(l<=1){
            l++;
        } else {
            l = 0;
        }
        printLight();           //EJECUTA ON DE LA ILUMINACION//
        printInfoLOn();
        digitalWrite(8,HIGH);   // ON pin 8 Iluminacion
        break;
    }
    if(flancoSubida(BTN_DOWN)){
        if(l>0){
            l--;
        } else {
            l = 100;
        }
        printLight();           //EJECUTA OFF DE LA ILUMINACION//
        printInfoLOff();
        digitalWrite(8,LOW);    // OFF pin 8 Iluminacion
        break;
    }
}
break; /** Fin set LIGHT**/

```

```

case S_SET_NUTRI: /**Inicio Set NUTRI**/

```

```

    if(flancoSubida(BTN_MENU)){
        estado = S_SET_INTER;
        printInter();
        printInfoI();
        break;
    }

    if(flancoSubida(BTN_EXIT)){
        estado = S_HOME;
        printHome();
        break;
    }

    if(flancoSubida(BTN_UP)){
        if( minuto<30){
            minuto+=5;
        } else {
            minuto =5;
        }
    }

```

```

    printNutri();          //DEBO EJECUTAR LA ACCION DEL TIEMPO DE RIEGO//
    printInfoN ();
    break;
}
if(flancoSubida(BTN_DOWN)){
    if(minuto>5){
        minuto-=5;
    } else {
        minuto = 30;}
    printNutri();          //DEBO EJECUTAR LA ACCION DEDEL TIEMPO DE RIEGO//
    printInfoN();

    break;
}
break; /** Fin set NUTRI**

case S_SET_INTER: /**Inicio Set Intervalo de Tiempo**/

    if(flancoSubida(BTN_MENU)){
        estado = S_SET_TEMP;
        printTemp();
        printInfoT();
        break;
    }

    if(flancoSubida(BTN_EXIT)){
        estado = S_HOME;
        printHome();
        break;
    }

    if(flancoSubida(BTN_UP)){
        if(hora<24){
            hora++;

        } else {
            hora = 1;
        }
        printInter();      //DEBO EJECUTAR LA ACCION DEL INTERVALO DE RIEGO//
        printInfoI();
        break;
    }
    if(flancoSubida(BTN_DOWN)){
        if(hora>1){
            hora--;
        } else {
            hora = 24;
        }
        printInter();      //DEBO EJECUTAR LA ACCION DEL INTERVALO DE RIEGO//
        printInfoI();
        break;
    }
break; /** Fin set INTER**/

};

}

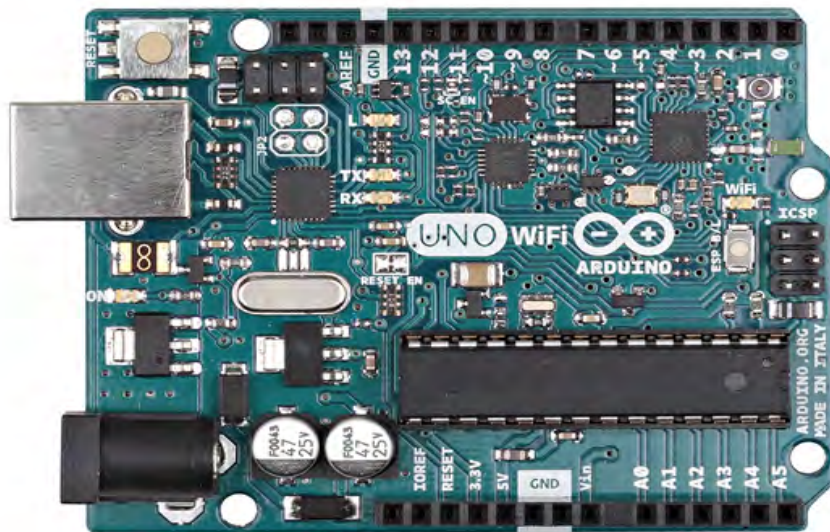
```


Anexo 2. Datasheet ArduinoUno Wifi:



Arduino UNO WiFi

The UNO WiFi board is the Arduino UNO with integrated WiFi. For beginner and advanced users alike.



The Arduino UNO WiFi board is based on the ATmega328 and it has an integrated ESP8266 WiFi Module. If you are starting out with Arduino, or starting with IoT, this is the board to get.

Power

The Arduino Uno Wi-Fi can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

5V. This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.

3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 1 A (according to the power input source).

GND. Ground pins.

IOREF. This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V supplies.

Memory

The ATmega328 has 32 KB Flash (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using `pinMode()`, `digitalWrite()` and `digitalRead()` functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialised functions:

Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.

External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the `attachInterrupt()` function for details.

PWM: 3, 5, 6, 9, 10 and 11. Provide 8-bit PWM output with the `analogWrite()` function.

SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off. The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the `analogReference()` function. Additionally, some pins have specialized functionality:

TWI: A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library. NOTE: the A4 or SDA pin and A5 or SCL pin are in use by the IO Expander SC16IS750IBS. There are a couple of other pins on the board:

AREF. Reference voltage for the analog inputs. Used with `analogReference()`.

RESET. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

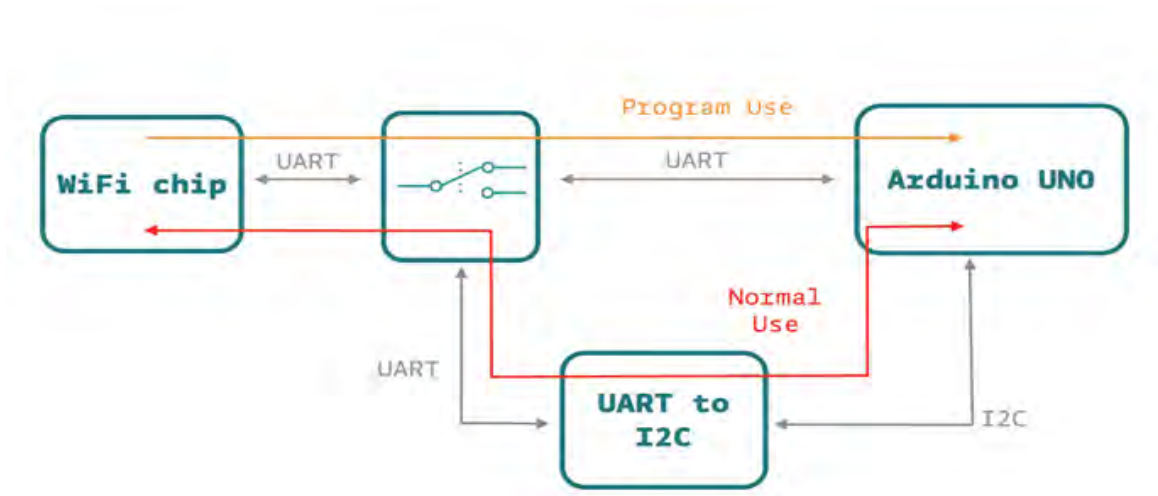
Communication

The Arduino Uno Wi-Fi has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial library allows for serial communication on any of the Uno's digital pins.

The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; For SPI communication, use the SPI library.

On the Arduino Uno Wi-Fi there's a SC16IS750IBS IO Expander Single UART with I2C-bus/SPI interface that allows the communication between AtMega 16u2, AtMega 328p and the ESP8266EX.



The Arduino Uno Wi-Fi allow you to communicate via Wi-Fi with your sensors or actuators mounted on your board to create easily and quickly your IoT System. You can use your Arduino Uno Wi-Fi as a client of your Wi-Fi network, as a server to connect other client devices or you can create an ad'hoc Wi-Fi connection.

The perfect way to communicate to internet via your Arduino Uno Wi-Fi is the Ciao Library and using, for example, the REST connector.

On the Arduino Uno Wi-Fi is pre-uploaded the RestServer sketch that allows you to command immediately your board via browser:

Connect to the Arduino Uno Wi-Fi SSID and go to the link <http://192.168.240.1/arduino/digital/13/1> to turn ON the LED L

Here other possible commands:

- * `"/arduino/digital/13"` -> `digitalRead(13)`
- * `"/arduino/digital/13/1"` -> `digitalWrite(13, HIGH)`
- * `"/arduino/analog/2/123"` -> `analogWrite(2, 123)`
- * `"/arduino/analog/2"` -> `analogRead(2)`
- * `"/arduino/mode/13/input"` -> `pinMode(13, INPUT)`
- * `"/arduino/mode/13/output"` -> `pinMode(13, OUTPUT)`

Programming

The Arduino Uno Wi-Fi can be programmed as a classic Arduino Uno with the Arduino software (download). Select "Arduino Uno Wi-Fi" from the Tools > Board menu (according to the microcontroller on your board).

The new features of the Arduino Uno Wi-Fi allow you to upload your sketch via Wi-Fi (in the same way of Arduino Yun), without any wired connection to your PC and your board.

The ATmega328 on the Arduino Uno Wi-Fi comes pre-burned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol.

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using Arduino ISP or similar.

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno Wi-Fi is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno WiFi is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno WiFi contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line.

USB Overcurrent Protection

The Arduino Uno Wi-Fi has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of the Uno Wi-Fi PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

Overview

The Arduino Uno Wi-Fi is the new Arduino Uno with an integrated Wi-Fi module! The board is based on the ATmega328P (datasheet (http://www.atmel.com/images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Complete.pdf)) with an ESP8266 Wi-Fi Module integrated (datasheet (<http://download.arduino.org/products/UNOWIFI/OA-ESP8266-Datasheet-EN-v4.3.pdf>)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

The ESP8266 Wi-Fi Module is a self contained SoC with integrated TCP/IP protocol stack that can give access to your Wi-Fi network. (Or the device can act as an access point.) One useful feature of Uno Wi-Fi is support for OTA (over-the-air) programming, either for transfer of Arduino sketches or Wi-Fi firmware.

ARDUINO MICROPROCESSOR

Processor	ESP8266
Architecture	Tensilica Xtensa LX106
Operating Voltage	3.3 V
Flash Memory	4 MB
RAM	8 MB instruction, 12 MB data
Clock Speed	80 MHz
WiFi	802.11 b/g/n 2.4 GHz
Wake up time	< 2 ms

ARDUINO MICROCONTROLLER

Microcontroller	ATmega328
Architecture	Atmel AVR 8-bit
Operating Voltage	5 V
Flash memory	32 KB
SRAM	2 KB
Clock Speed	16 MHz
Analog I/O Pins	6
EEPROM	1 KB
DC Current per I/O Pins	40 mA

GENERAL

Digital I/O Pins	20
PWM Output	6
Power Consumption	93 mA
PCB Size	53 x 68.5 mm
Weight	0.025 Kg
Product Code	A000133

Anexo 3. Datasheet DTH22 Digital Humidity-Temperature Sensor:



Standard AM2302/DHT22



AM2302/DHT22 with big case and wires

Digital relative humidity & temperature sensor AM2302/DHT22

1. Feature & Application:

- *High precision
- *Capacitive type
- *Full range temperature compensated
- *Relative humidity and temperature measurement
- *Calibrated digital signal
- *Outstanding long-term stability
- *Extra components not needed
- *Long transmission distance, up to 100 meters
- *Low power consumption
- *4 pins packaged and fully interchangeable

2. Description:

AM2302 output calibrated digital signal. It applies exclusive digital-signal-collecting-technique and humidity sensing technology, assuring its reliability and stability. Its sensing elements is connected with 8-bit single-chip computer.

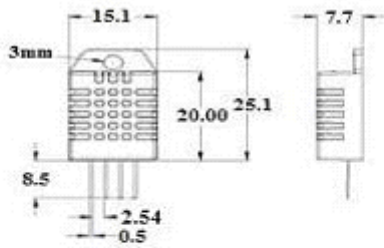
Every sensor of this model is temperature compensated and calibrated in accurate calibration chamber and the calibration-coefficient is saved in type of programme in OTP memory, when the sensor is detecting, it will cite coefficient from memory.

Small size & low consumption & long transmission distance(100m) enable AM2302 to be suited in all kinds of harsh application occasions. Single-row packaged with four pins, making the connection very convenient.

3. Technical Specification:

Model	AM2302	
Power supply	3.3-5.5V DC	
Output signal	digital signal via 1-wire bus	
Sensing element	Polymer humidity capacitor	
Operating range	humidity 0-100%RH;	temperature -40~80Celsius
Accuracy	humidity +-2%RH (Max +-5%RH);	temperature +-0.5Celsius
Resolution or sensitivity	humidity 0.1%RH;	temperature 0.1Celsius
Repeatability	humidity +-1%RH;	temperature +-0.2Celsius
Humidity hysteresis	+-0.3%RH	
Long-term Stability	+-0.5%RH/year	
Interchangeability	fully interchangeable	

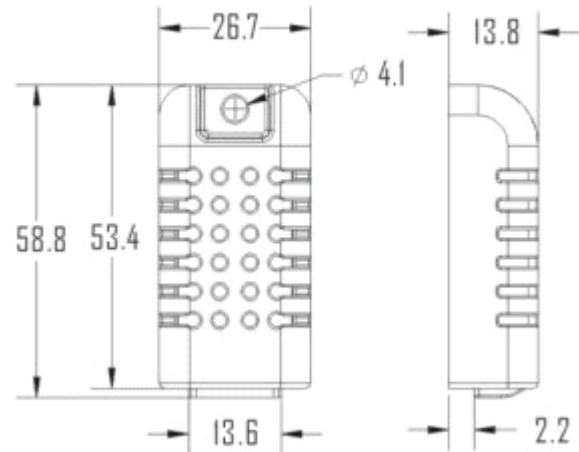
4. Dimensions: (unit---mm)



Pin sequence number: 1 2 3 4 (from left to right direction).

Pin	Function
1	VDD—power supply
2	DATA—signal
3	GND
4	GND

Standard AM2302's dimensions as above

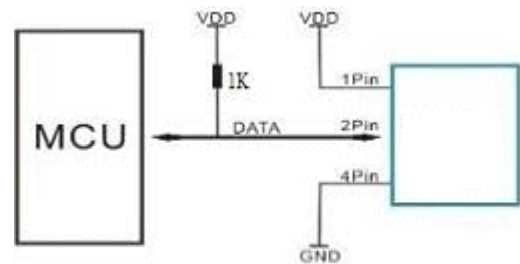


Big case's dimensions as above

Red wire--power supply, Black wire--GND

Yellow wire--Data output

5. Electrical connection diagram:



6. Operating specifications:

(1) Power and Pins

Power's voltage should be 3.3-5.5V DC. When power is supplied to sensor, don't send any instruction to the sensor within one second to pass unstable status. One capacitor valued 100nF can be added between VDD and GND for wave filtering.

(2) Communication and signal

1-wire bus is used for communication between MCU and AM2302. (Our 1-wire bus is specially designed, it's different from Maxim/Dallas 1-wire bus, so it's incompatible with Dallas 1-wire bus.)

Illustration of our 1-wire bus:

DATA=16 bits RH data+16 bits Temperature data+8 bits check-sum

Example: MCU has received 40 bits data from AM2302 as

0000 0010 1000 1100 0000 0001 0101 1111 1110 1110
 16 bits RH data 16 bits T data check sum

Here we convert 16 bits RH data from binary system to decimal system,

0000 0010 1000 1100 → 652
 Binary system Decimal system

RH=652/10=65.2%RH

Here we convert 16 bits T data from binary system to decimal system,

0000 0001 0101 1111 → 351
 Binary system Decimal system

T=351/10=35.1°C

When highest bit of temperature is 1, it means the temperature is below 0 degree Celsius.

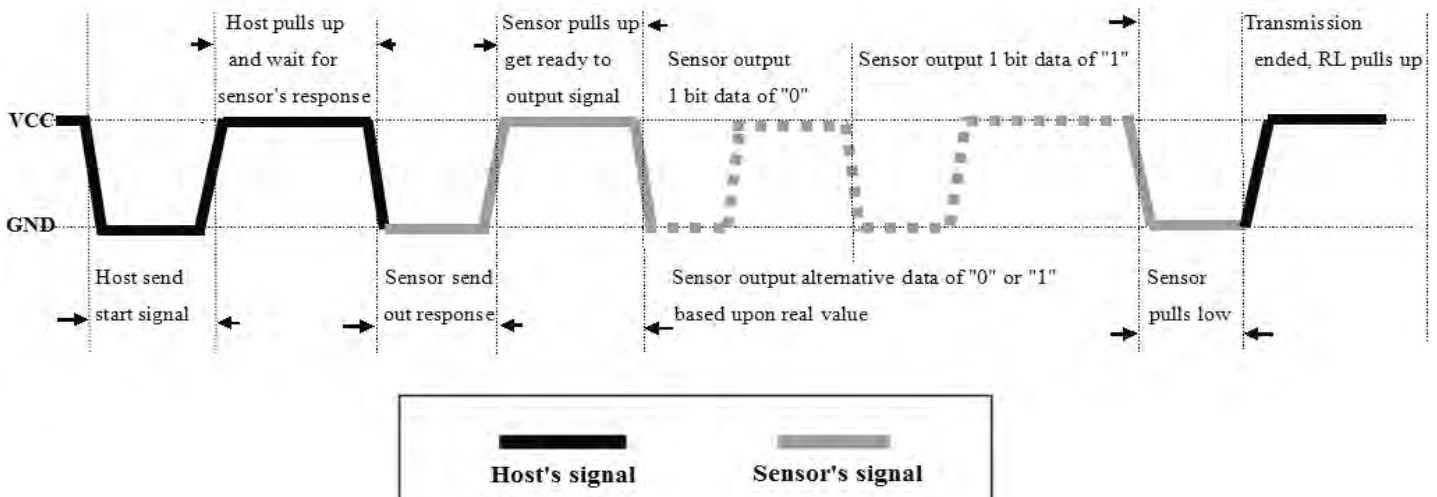
Example: 1000 0000 0110 0101, T= minus 10.1°C
 16 bits T data

Sum=0000 0010+1000 1100+0000 0001+0101 1111=1110 1110

Check-sum=the last 8 bits of Sum=1110 1110

When MCU send start signal, AM2302 change from standby-status to running-status. When MCU finishes sending the start signal, AM2302 will send response signal of 40-bit data that reflect the relative humidity and temperature to MCU. Without start signal from MCU, AM2302 will not give response signal to MCU. One start signal for one response data from AM2302 that reflect the relative humidity and temperature. AM2302 will change to standby status when data collecting finished if it don't receive start signal from MCU again.

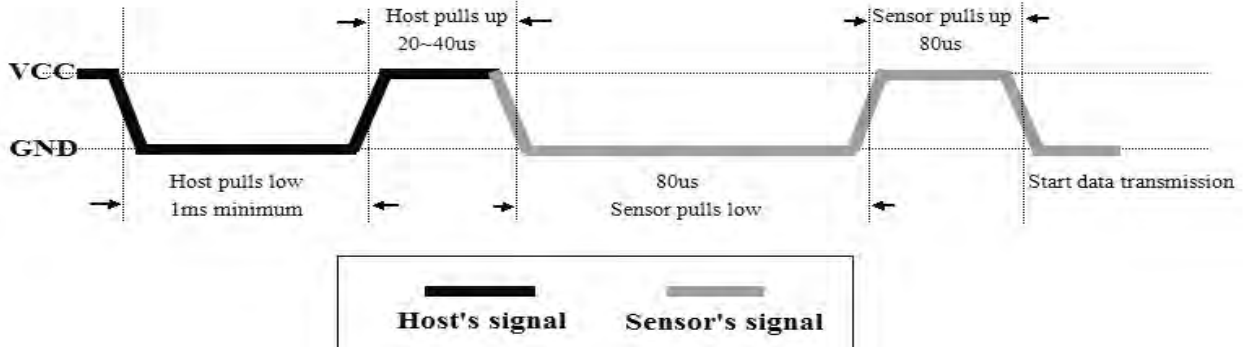
See below figure for overall communication process, **the interval of whole process must beyond 2 seconds.**



1) Step 1: MCU send out start signal to AM2302 and AM2302 send response signal to MCU

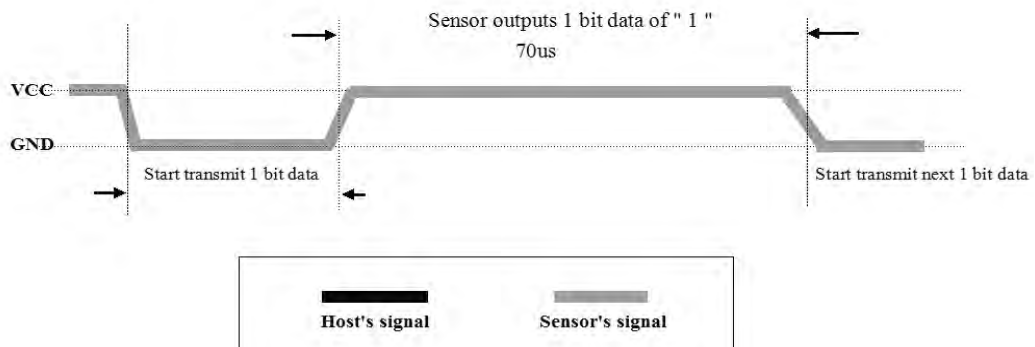
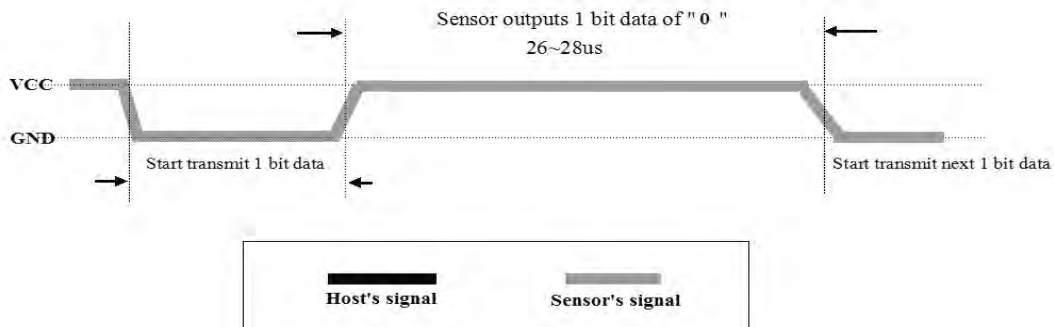
Data-bus's free status is high voltage level. When communication between MCU and AM2302 begins, MCU will pull low data-bus and this process must beyond at least 1~10ms to ensure AM2302 could detect MCU's signal, then MCU will pull up and wait 20-40us for AM2302's response.

When AM2302 detect the start signal, AM2302 will pull low the bus 80us as response signal, then AM2302 pulls up 80us for preparation to send data. See below figure:



2). Step 2: AM2302 send data to MCU

When AM2302 is sending data to MCU, every bit's transmission begin with low-voltage-level that last 50us, the following high-voltage-level signal's length decide the bit is "1" or "0". See below figures:



Attention:

If signal from AM2302 is always high-voltage-level, it means AM2302 is not working properly, please check the electrical connection status.

7. Electrical Characteristics:

Items	Condition	Min	Typical	Max	Unit
Power supply	DC	3.3	5	6	V
Current supply	Measuring	1		1.5	mA
	Stand-by	40	Null	50	uA
Collecting period	Second		2		Second

8. Attentions of application:

(1) Operating and storage conditions

We don't recommend the applying RH-range beyond the range stated in this specification. The AM2302 sensor can recover after working in abnormal operating condition to calibrated status, but will accelerate sensors' aging.

(2) Attentions to chemical materials

Vapor from chemical materials may interfere AM2302's sensitive-elements and debase AM2302's sensitivity.

(3) Disposal when (1) & (2) happens

Step one: Keep the AM2302 sensor at condition of Temperature 50~60Celsius, humidity <10%RH for 2 hours;

Step two: After step one, keep the AM2302 sensor at condition of Temperature 20~30Celsius, humidity >70%RH for 5 hours.

(4) Attention to temperature's affection

Relative humidity strongly depend on temperature, that is why we use temperature compensation technology to ensure accurate measurement of RH. But it's still be much better to keep the sensor at same temperature when sensing.

AM2302 should be mounted at the place as far as possible from parts that may cause change to temperature.

(5) Attentions to light

Long time exposure to strong light and ultraviolet may debase AM2302's performance.

(6) Attentions to connection wires

The connection wires' quality will effect communication's quality and distance, high quality shielding-wire is recommended.

(7) Other attentions

* Welding temperature should be bellow 260Celsius.

* Avoid using the sensor under dew condition.

* Don't use this product in safety or emergency stop devices or any other occasion that failure of AM2302 may cause personal injury.

Anexo 4. Datasheet Real Time Clock Rtc-Ds3231

DS3231

Extremely Accurate I2C-Integrated RTC/TCXO/Crystal

General Description

The DS3231 is a low-cost, extremely accurate I²C real-time clock (RTC) with an integrated temperature-compensated crystal oscillator (TCXO) and crystal. The device incorporates a battery input, and maintains accurate timekeeping when main power to the device is interrupted. The integration of the crystal resonator enhances the long-term accuracy of the device as well as reduces the piece-part count in a manufacturing line. The DS3231 is available in commercial and industrial temperature ranges, and is offered in a 16-pin, 300-mil SO package.

The RTC maintains seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with an AM/PM indicator. Two programmable time-of-day alarms and a programmable square-wave output are provided. Address and data are transferred serially through an I²C bidirectional bus.

A precision temperature-compensated voltage reference and comparator circuit monitors the status of V_{CC} to detect power failures, to provide a reset output, and to automatically switch to the backup supply when necessary. Additionally, the RST pin is monitored as a pushbutton input for generating a µP reset.

Benefits and Features

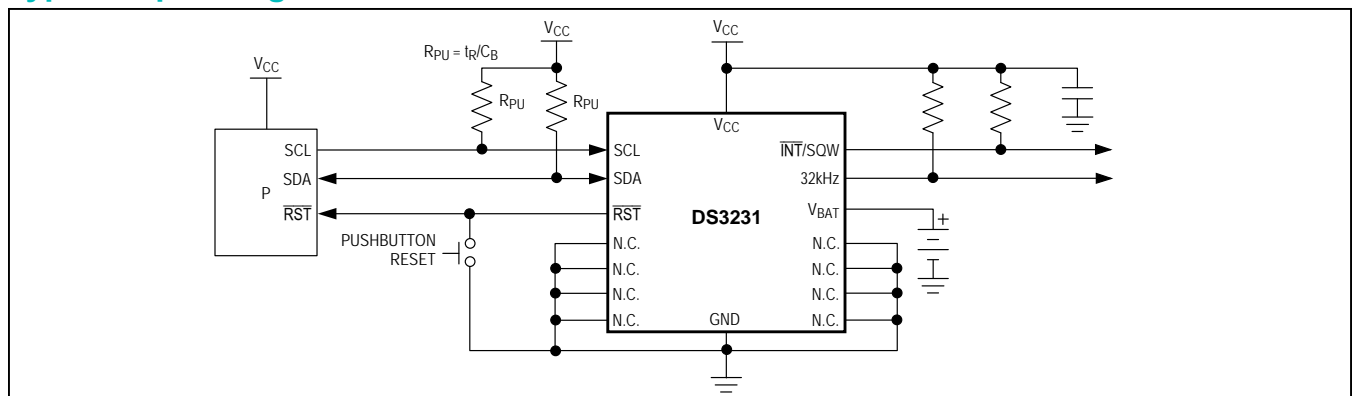
- Highly Accurate RTC Completely Manages All Timekeeping Functions
 - Real-Time Clock Counts Seconds, Minutes, Hours, Date of the Month, Month, Day of the Week, and Year, with Leap-Year Compensation Valid Up to 2100
 - Accuracy ±2ppm from 0°C to +40°C
 - Accuracy ±3.5ppm from -40°C to +85°C
 - Digital Temp Sensor Output: ±3°C Accuracy
 - Register for Aging Trim
 - RST Output/Pushbutton Reset Debounce Input
 - Two Time-of-Day Alarms
 - Programmable Square-Wave Output Signal
- Simple Serial Interface Connects to Most Microcontrollers
 - Fast (400kHz) I²C Interface
- Battery-Backup Input for Continuous Timekeeping
 - Low Power Operation Extends Battery-Backup Run Time
 - 3.3V Operation
- Operating Temperature Ranges: Commercial (0°C to +70°C) and Industrial (-40°C to +85°C)
- Underwriters Laboratories® (UL) Recognized

Applications

- Servers
- Telematics
- Utility Power Meters
- GPS

Ordering Information and Pin Configuration appear at end of data sheet.

Typical Operating Circuit



Underwriters Laboratories is a registered certification mark of Underwriters Laboratories Inc.

Absolute Maximum Ratings

Voltage Range on Any Pin Relative to Ground -0.3V to +6.0V
 Junction-to-Ambient Thermal Resistance (θ_{JA}) (Note 1) 73°C/W
 Junction-to-Case Thermal Resistance (θ_{JC}) (Note 1) 23°C/W
 Operating Temperature Range
 DS3231S 0°C to +70°C
 DS3231SN -40°C to +85°C

Junction Temperature +125°C
 Storage Temperature Range -40°C to +85°C
 Lead Temperature (soldering, 10s) +260°C
 Soldering Temperature (reflow, 2 times max) +260°C
 (see the *Handling, PCB Layout, and Assembly* section)

Note 1: Package thermal resistances were obtained using the method described in JEDEC specification JESD51-7, using a four-layer board. For detailed information on package thermal considerations, refer to www.maximintegrated.com/thermal-tutorial.

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Recommended Operating Conditions

($T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.) (Notes 2, 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Supply Voltage	V_{CC}		2.3	3.3	5.5	V
	V_{BAT}		2.3	3.0	5.5	V
Logic 1 Input SDA, SCL	V_{IH}		0.7 x V_{CC}		$V_{CC} + 0.3$	V
Logic 0 Input SDA, SCL	V_{IL}		-0.3		0.3 x V_{CC}	V

Electrical Characteristics

($V_{CC} = 2.3V$ to $5.5V$, $V_{CC} =$ Active Supply (see Table 1), $T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.) (Typical values are at $V_{CC} = 3.3V$, $V_{BAT} = 3.0V$, and $T_A = +25^\circ C$, unless otherwise noted.) (Notes 2, 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Active Supply Current	I_{CCA}	(Notes 4, 5)	$V_{CC} = 3.63V$		200	μA
			$V_{CC} = 5.5V$		300	
Standby Supply Current	I_{CCS}	I ² C bus inactive, 32kHz output on, SQW output off (Note 5)	$V_{CC} = 3.63V$		110	μA
			$V_{CC} = 5.5V$		170	
Temperature Conversion Current	$I_{CCSCONV}$	I ² C bus inactive, 32kHz output on, SQW output off	$V_{CC} = 3.63V$		575	μA
			$V_{CC} = 5.5V$		650	
Power-Fail Voltage	V_{PF}		2.45	2.575	2.70	V
Logic 0 Output, 32kHz, \overline{INT}/SQW , SDA	V_{OL}	$I_{OL} = 3mA$			0.4	V
Logic 0 Output, \overline{RST}	V_{OL}	$I_{OL} = 1mA$			0.4	V
Output Leakage Current 32kHz, \overline{INT}/SQW , SDA	I_{LO}	Output high impedance	-1	0	+1	μA
Input Leakage SCL	I_{LI}		-1		+1	μA
\overline{RST} Pin I/O Leakage	I_{OL}	\overline{RST} high impedance (Note 6)	-200		+10	μA
V_{BAT} Leakage Current (V_{CC} Active)	I_{BATLKG}			25	100	nA

Electrical Characteristics (continued)

($V_{CC} = 2.3V$ to $5.5V$, V_{CC} = Active Supply (see Table 1), $T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.) (Typical values are at $V_{CC} = 3.3V$, $V_{BAT} = 3.0V$, and $T_A = +25^\circ C$, unless otherwise noted.) (Notes 2, 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Output Frequency	f_{OUT}	$V_{CC} = 3.3V$ or $V_{BAT} = 3.3V$		32.768		kHz
Frequency Stability vs. Temperature (Commercial)	$\Delta f/f_{OUT}$	$V_{CC} = 3.3V$ or $V_{BAT} = 3.3V$, aging offset = 00h	0°C to +40°C		±2	ppm
			>40°C to +70°C		±3.5	
Frequency Stability vs. Temperature (Industrial)	$\Delta f/f_{OUT}$	$V_{CC} = 3.3V$ or $V_{BAT} = 3.3V$, aging offset = 00h	-40°C to <0°C		±3.5	ppm
			0°C to +40°C		±2	
			>40°C to +85°C		±3.5	
Frequency Stability vs. Voltage	$\Delta f/V$			1		ppm/V
Trim Register Frequency Sensitivity per LSB	$\Delta f/LSB$	Specified at:	-40°C		0.7	ppm
			+25°C		0.1	
			+70°C		0.4	
			+85°C		0.8	
Temperature Accuracy	Temp	$V_{CC} = 3.3V$ or $V_{BAT} = 3.3V$	-3		+3	°C
Crystal Aging	$\Delta f/f_O$	After reflow, not production tested	First year		±1.0	ppm
			0–10 years		±5.0	

Electrical Characteristics

($V_{CC} = 0V$, $V_{BAT} = 2.3V$ to $5.5V$, $T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.) (Note 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Active Battery Current	I_{BATA}	$\overline{EOSC} = 0$, BBSQW = 0, SCL = 400kHz (Note 5)	$V_{BAT} = 3.63V$		70	µA
			$V_{BAT} = 5.5V$		150	
Timekeeping Battery Current	I_{BATT}	$\overline{EOSC} = 0$, BBSQW = 0, EN32kHz = 1, SCL = SDA = 0V or SCL = SDA = V_{BAT} (Note 5)	$V_{BAT} = 3.63V$	0.84	3.0	µA
			$V_{BAT} = 5.5V$	1.0	3.5	
Temperature Conversion Current	I_{BATTCC}	$\overline{EOSC} = 0$, BBSQW = 0, SCL = SDA = 0V or SCL = SDA = V_{BAT}	$V_{BAT} = 3.63V$		575	µA
			$V_{BAT} = 5.5V$		650	
Data-Retention Current	I_{BATTDR}	$\overline{EOSC} = 1$, SCL = SDA = 0V, +25°C			100	nA

AC Electrical Characteristics

($V_{CC} = V_{CC(MIN)}$ to $V_{CC(MAX)}$ or $V_{BAT} = V_{BAT(MIN)}$ to $V_{BAT(MAX)}$, $V_{BAT} > V_{CC}$, $T_A = T_{MIN}$ to T_{MAX} , unless otherwise noted.) (Note 2)

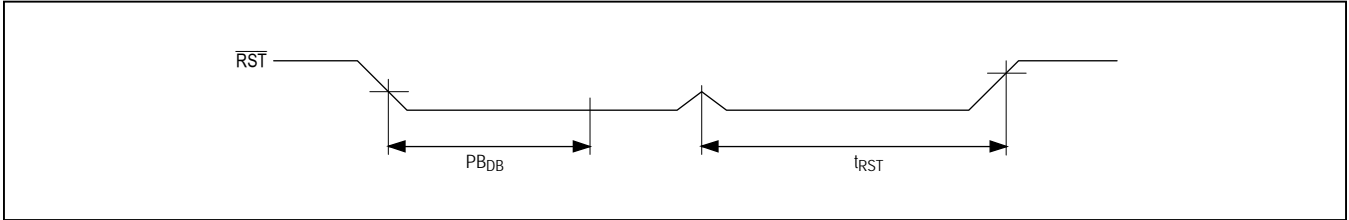
PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
SCL Clock Frequency	f_{SCL}	Fast mode	100		400	kHz
		Standard mode	0		100	
Bus Free Time Between STOP and START Conditions	t_{BUF}	Fast mode	1.3			μ s
		Standard mode	4.7			
Hold Time (Repeated) START Condition (Note 7)	$t_{HD:STA}$	Fast mode	0.6			μ s
		Standard mode	4.0			
Low Period of SCL Clock	t_{LOW}	Fast mode	1.3			μ s
		Standard mode	4.7			
High Period of SCL Clock	t_{HIGH}	Fast mode	0.6			μ s
		Standard mode	4.0			
Data Hold Time (Notes 8, 9)	$t_{HD:DAT}$	Fast mode	0		0.9	μ s
		Standard mode	0		0.9	
Data Setup Time (Note 10)	$t_{SU:DAT}$	Fast mode	100			ns
		Standard mode	250			
START Setup Time	$t_{SU:STA}$	Fast mode	0.6			μ s
		Standard mode	4.7			
Rise Time of Both SDA and SCL Signals (Note 11)	t_R	Fast mode	20 +		300	ns
		Standard mode	0.1 C_B		1000	
Fall Time of Both SDA and SCL Signals (Note 11)	t_F	Fast mode	20 +		300	ns
		Standard mode	0.1 C_B		300	
Setup Time for STOP Condition	$t_{SU:STO}$	Fast mode	0.6			μ s
		Standard mode	4.7			
Capacitive Load for Each Bus Line	C_B	(Note 11)			400	pF
Capacitance for SDA, SCL	$C_{I/O}$			10		pF
Pulse Width of Spikes That Must Be Suppressed by the Input Filter	t_{SP}			30		ns
Pushbutton Debounce	PB_{DB}			250		ms
Reset Active Time	t_{RST}			250		ms
Oscillator Stop Flag (OSF) Delay	t_{OSF}	(Note 12)		100		ms
Temperature Conversion Time	t_{CONV}			125	200	ms

Power-Switch Characteristics

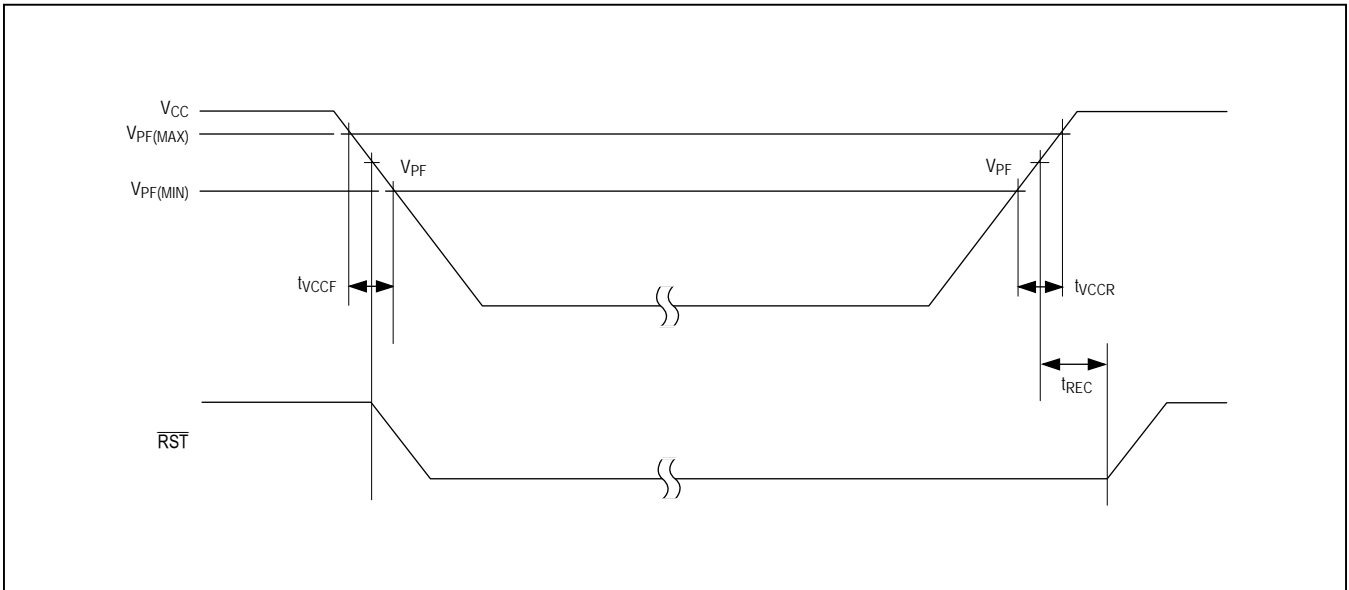
($T_A = T_{MIN}$ to T_{MAX})

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
V_{CC} Fall Time; $V_{PF(MAX)}$ to $V_{PF(MIN)}$	t_{VCCF}		300			μ s
V_{CC} Rise Time; $V_{PF(MIN)}$ to $V_{PF(MAX)}$	t_{VCCR}		0			μ s
Recovery at Power-Up	t_{REC}	(Note 13)		250	300	ms

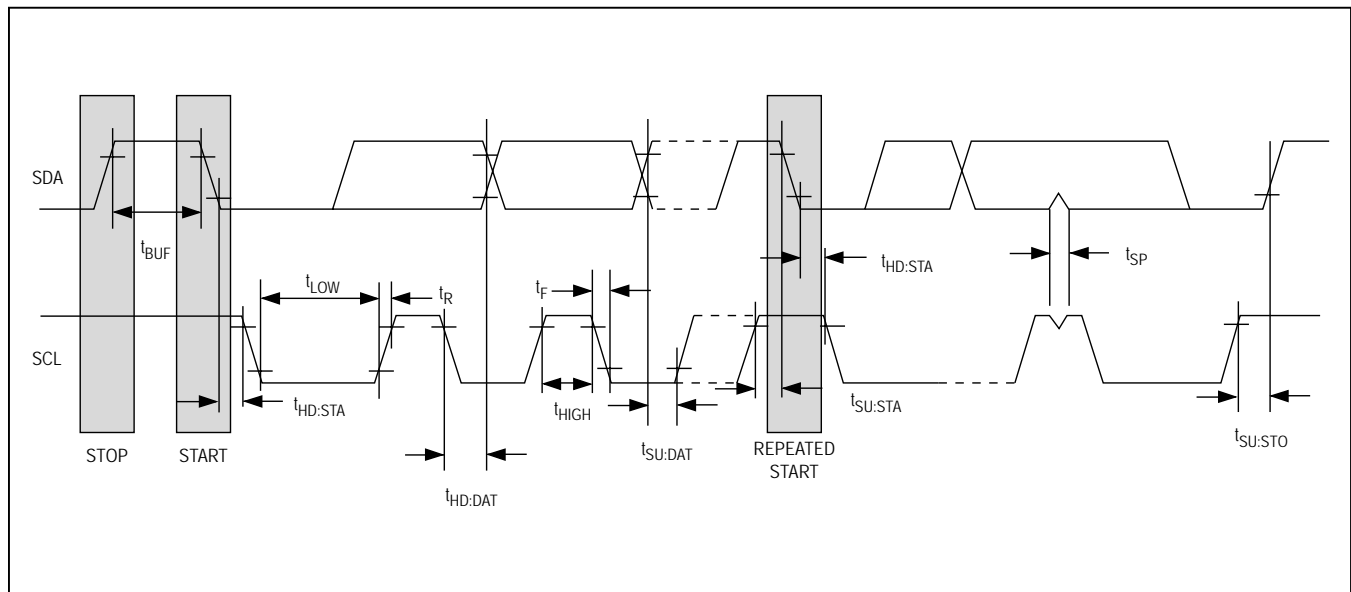
Pushbutton Reset Timing



Power-Switch Timing



Data Transfer on I2C Serial Bus



WARNING: Negative undershoots below -0.3V while the part is in battery-backed mode may cause loss of data.

Note 2: Limits at -40°C are guaranteed by design and not production tested.

Note 3: All voltages are referenced to ground.

Note 4: I_{CCA} —SCL clocking at max frequency = 400kHz.

Note 5: Current is the averaged input current, which includes the temperature conversion current.

Note 6: The \overline{RST} pin has an internal 50k Ω (nominal) pullup resistor to V_{CC} .

Note 7: After this period, the first clock pulse is generated.

Note 8: A device must internally provide a hold time of at least 300ns for the SDA signal (referred to the $V_{IH(MIN)}$ of the SCL signal) to bridge the undefined region of the falling edge of SCL.

Note 9: The maximum $t_{HD:DAT}$ needs only to be met if the device does not stretch the low period (t_{LOW}) of the SCL signal.

Note 10: A fast-mode device can be used in a standard-mode system, but the requirement $t_{SU:DAT} \geq 250$ ns must then be met. This is automatically the case if the device does not stretch the low period of the SCL signal. If such a device does stretch the low period of the SCL signal, it must output the next data bit to the SDA line $t_{R(MAX)} + t_{SU:DAT} = 1000 + 250 = 1250$ ns before the SCL line is released.

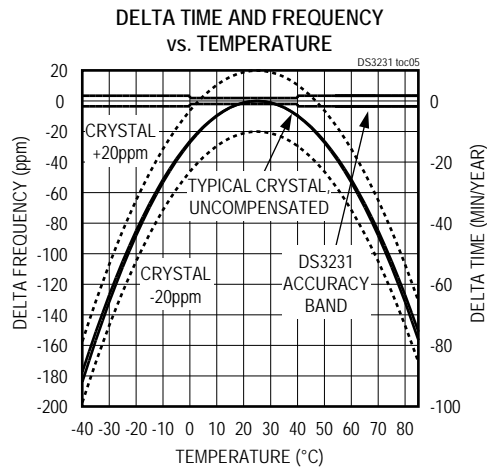
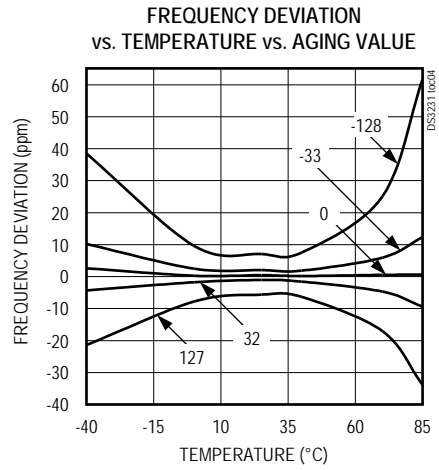
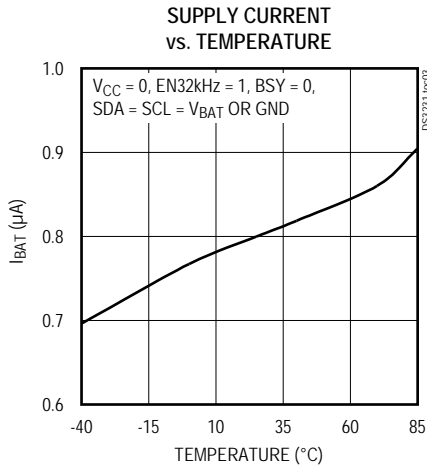
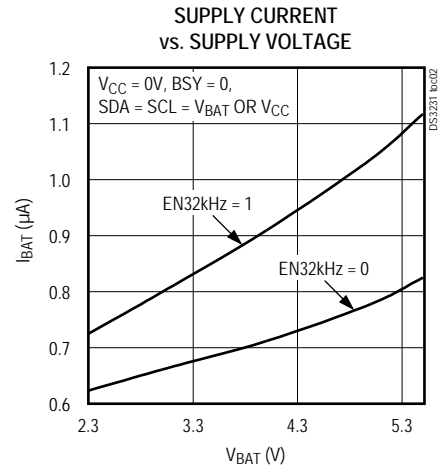
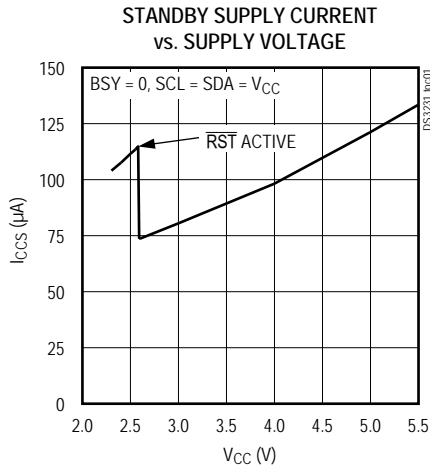
Note 11: C_B —total capacitance of one bus line in pF.

Note 12: The parameter t_{OSF} is the period of time the oscillator must be stopped for the OSF flag to be set over the voltage range of $0.0V \leq V_{CC} \leq V_{CC(MAX)}$ and $2.3V \leq V_{BAT} \leq 3.4V$.

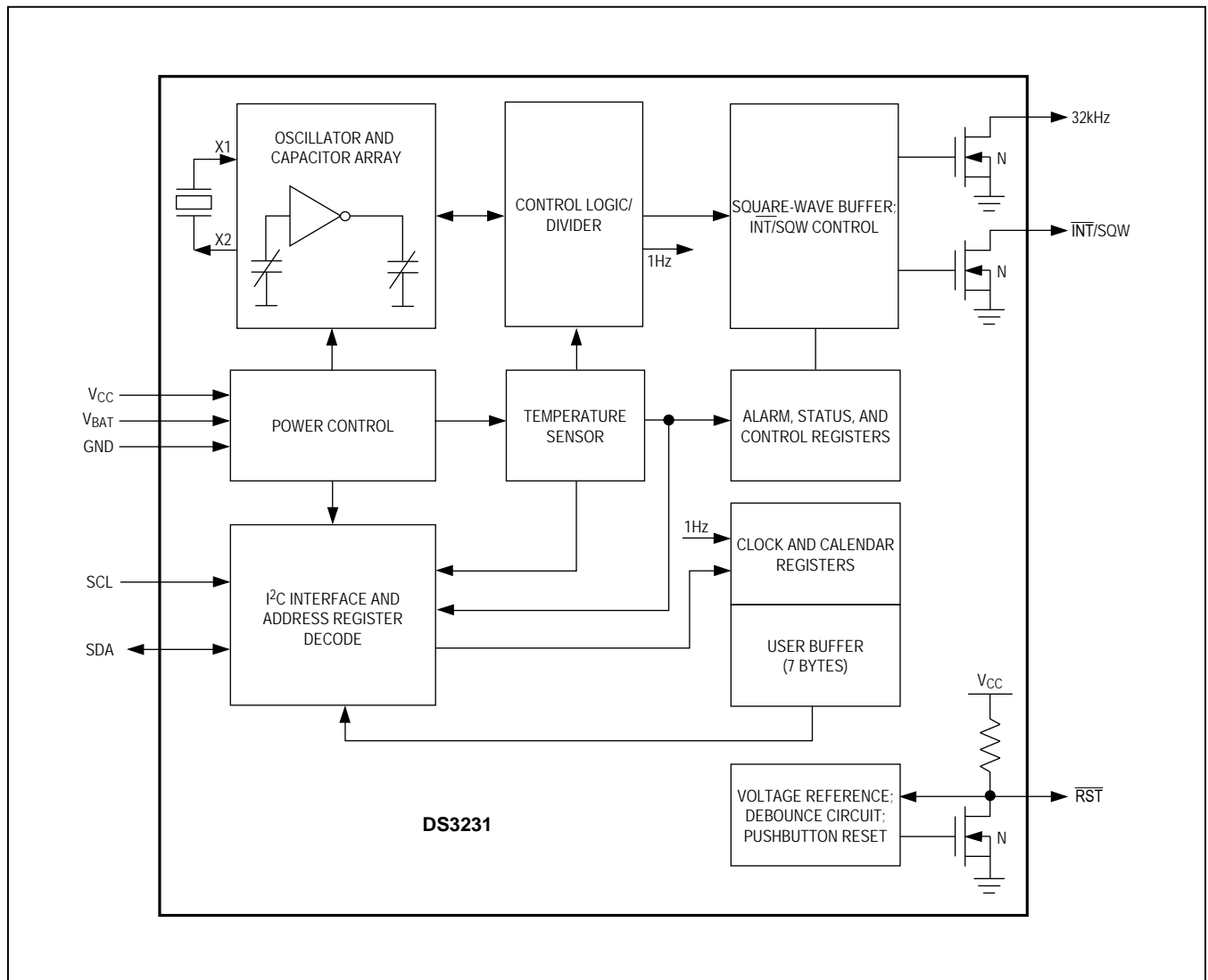
Note 13: This delay applies only if the oscillator is enabled and running. If the \overline{EOSC} bit is a 1, t_{REC} is bypassed and \overline{RST} immediately goes high. The state of \overline{RST} does not affect the I2C interface, RTC, or TCXO.

Typical Operating Characteristics

($V_{CC} = +3.3V$, $T_A = +25^\circ C$, unless otherwise noted.)



Block Diagram



Pin Description

PIN	NAME	FUNCTION
1	32kHz	32kHz Output. This open-drain pin requires an external pullup resistor. When enabled, the output operates on either power supply. It may be left open if not used.
2	V _{CC}	DC Power Pin for Primary Power Supply. This pin should be decoupled using a 0.1μF to 1.0μF capacitor. If not used, connect to ground.
3	$\overline{\text{INT}}/\text{SQW}$	Active-Low Interrupt or Square-Wave Output. This open-drain pin requires an external pullup resistor connected to a supply at 5.5V or less. This multifunction pin is determined by the state of the INTCN bit in the Control Register (0Eh). When INTCN is set to logic 0, this pin outputs a square wave and its frequency is determined by RS2 and RS1 bits. When INTCN is set to logic 1, then a match between the timekeeping registers and either of the alarm registers activates the $\overline{\text{INT}}/\text{SQW}$ pin (if the alarm is enabled). Because the INTCN bit is set to logic 1 when power is first applied, the pin defaults to an interrupt output with alarms disabled. The pullup voltage can be up to 5.5V, regardless of the voltage on V _{CC} . If not used, this pin can be left unconnected.
4	$\overline{\text{RST}}$	Active-Low Reset. This pin is an open-drain input/output. It indicates the status of V _{CC} relative to the V _{PF} specification. As V _{CC} falls below V _{PF} , the $\overline{\text{RST}}$ pin is driven low. When V _{CC} exceeds V _{PF} , for t _{RST} , the $\overline{\text{RST}}$ pin is pulled high by the internal pullup resistor. The active-low, open-drain output is combined with a debounced pushbutton input function. This pin can be activated by a pushbutton reset request. It has an internal 50kΩ nominal value pullup resistor to V _{CC} . No external pullup resistors should be connected. If the oscillator is disabled, t _{REC} is bypassed and $\overline{\text{RST}}$ immediately goes high.
5–12	N.C.	No Connection. Must be connected to ground.
13	GND	Ground
14	V _{BAT}	Backup Power-Supply Input. When using the device with the V _{BAT} input as the primary power source, this pin should be decoupled using a 0.1μF to 1.0μF low-leakage capacitor. When using the device with the V _{BAT} input as the backup power source, the capacitor is not required. If V _{BAT} is not used, connect to ground. The device is UL recognized to ensure against reverse charging when used with a primary lithium battery. Go to www.maximintegrated.com/qa/info/ul .
15	SDA	Serial Data Input/Output. This pin is the data input/output for the I ² C serial interface. This open-drain pin requires an external pullup resistor. The pullup voltage can be up to 5.5V, regardless of the voltage on V _{CC} .
16	SCL	Serial Clock Input. This pin is the clock input for the I ² C serial interface and is used to synchronize data movement on the serial interface. Up to 5.5V can be used for this pin, regardless of the voltage on V _{CC} .

Detailed Description

The DS3231 is a serial RTC driven by a temperature-compensated 32kHz crystal oscillator. The TCXO provides a stable and accurate reference clock, and maintains the RTC to within ±2 minutes per year accuracy from -40°C to +85°C. The TCXO frequency output is available at the 32kHz pin. The RTC is a low-power clock/calendar with two programmable time-of-day alarms and a programmable square-wave output. The $\overline{\text{INT}}/\text{SQW}$ provides either an interrupt signal due to alarm conditions or a square-wave output. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap

year. The clock operates in either the 24-hour or 12-hour format with an AM/PM indicator. The internal registers are accessible through an I²C bus interface.

A temperature-compensated voltage reference and comparator circuit monitors the level of V_{CC} to detect power failures and to automatically switch to the backup supply when necessary. The $\overline{\text{RST}}$ pin provides an external pushbutton function and acts as an indicator of a power-fail event.

Operation

The block diagram shows the main elements of the DS3231. The eight blocks can be grouped into four functional groups: TCXO, power control, pushbutton function, and RTC. Their operations are described separately in the following sections.

32kHz TCXO

The temperature sensor, oscillator, and control logic form the TCXO. The controller reads the output of the on-chip temperature sensor and uses a lookup table to determine the capacitance required, adds the aging correction in AGE register, and then sets the capacitance selection registers. New values, including changes to the AGE register, are loaded only when a change in the temperature value occurs, or when a user-initiated temperature conversion is completed. Temperature conversion occurs on initial application of V_{CC} and once every 64 seconds afterwards.

Power Control

This function is provided by a temperature-compensated voltage reference and a comparator circuit that monitors the V_{CC} level. When V_{CC} is greater than V_{PF} , the part is powered by V_{CC} . When V_{CC} is less than V_{PF} but greater than V_{BAT} , the DS3231 is powered by V_{CC} . If V_{CC} is less than V_{PF} and is less than V_{BAT} , the device is powered by V_{BAT} . See Table 1.

Table 1. Power Control

SUPPLY CONDITION	ACTIVE SUPPLY
$V_{CC} < V_{PF}, V_{CC} < V_{BAT}$	V_{BAT}
$V_{CC} < V_{PF}, V_{CC} > V_{BAT}$	V_{CC}
$V_{CC} > V_{PF}, V_{CC} < V_{BAT}$	V_{CC}
$V_{CC} > V_{PF}, V_{CC} > V_{BAT}$	V_{CC}

To preserve the battery, the first time V_{BAT} is applied to the device, the oscillator will not start up until V_{CC} exceeds V_{PF} , or until a valid I²C address is written to the part. Typical oscillator startup time is less than one second. Approximately 2 seconds after V_{CC} is applied, or a valid I²C address is written, the device makes a temperature measurement and applies the calculated correction to the oscillator. Once the oscillator is running, it continues to run as long as a valid power source is available (V_{CC} or V_{BAT}), and the device continues to measure the temperature and correct the oscillator frequency every 64 seconds.

On the first application of power (V_{CC}) or when a valid I²C address is written to the part (V_{BAT}), the time and date registers are reset to 01/01/00 01 00:00:00 (DD/MM/YY DOW HH:MM:SS).

V_{BAT} Operation

There are several modes of operation that affect the amount of V_{BAT} current that is drawn. While the device

is powered by V_{BAT} and the serial interface is active, active battery current, I_{BATA} , is drawn. When the serial interface is inactive, timekeeping current (I_{BATT}), which includes the averaged temperature conversion current, I_{BATTTC} , is used (refer to Application Note 3644: *Power Considerations for Accurate Real-Time Clocks* for details). Temperature conversion current, I_{BATTTC} , is specified since the system must be able to support the periodic higher current pulse and still maintain a valid voltage level. Data retention current, I_{BATTDR} , is the current drawn by the part when the oscillator is stopped ($\overline{EOSC} = 1$). This mode can be used to minimize battery requirements for times when maintaining time and date information is not necessary, e.g., while the end system is waiting to be shipped to a customer.

Pushbutton Reset Function

The DS3231 provides for a pushbutton switch to be connected to the \overline{RST} output pin. When the DS3231 is not in a reset cycle, it continuously monitors the \overline{RST} signal for a low going edge. If an edge transition is detected, the DS3231 debounces the switch by pulling the \overline{RST} low. After the internal timer has expired (PB_{DB}), the DS3231 continues to monitor the \overline{RST} line. If the line is still low, the DS3231 continuously monitors the line looking for a rising edge. Upon detecting release, the DS3231 forces the \overline{RST} pin low and holds it low for t_{RST} .

\overline{RST} is also used to indicate a power-fail condition. When V_{CC} is lower than V_{PF} , an internal power-fail signal is generated, which forces the \overline{RST} pin low. When V_{CC} returns to a level above V_{PF} , the \overline{RST} pin is held low for approximately 250ms (t_{REC}) to allow the power supply to stabilize. If the oscillator is not running (see the *Power Control* section) when V_{CC} is applied, t_{REC} is bypassed and \overline{RST} immediately goes high. Assertion of the \overline{RST} output, whether by pushbutton or power-fail detection, does not affect the internal operation of the DS3231.

Real-Time Clock

With the clock source from the TCXO, the RTC provides seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with an $\overline{AM/PM}$ indicator.

The clock provides two programmable time-of-day alarms and a programmable square-wave output. The INT/SQW pin either generates an interrupt due to alarm condition or outputs a square-wave signal and the selection is controlled by the bit INTCN.

ADDRESS	BIT 7 MSB	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0 LSB	FUNCTION	RANGE
00h	0	10 Seconds			Seconds				Seconds	00–59
01h	0	10 Minutes			Minutes				Minutes	00–59
02h	0	12/24	AM/PM	10 Hour	Hour				Hours	1–12 + AM/PM 00–23
			20 Hour							
03h	0	0	0	0	0	Day			Day	1–7
04h	0	0	10 Date		Date				Date	01–31
05h	Century	0	0	10 Month	Month				Month/ Century	01–12 + Century
06h	10 Year				Year				Year	00–99
07h	A1M1	10 Seconds			Seconds				Alarm 1 Seconds	00–59
08h	A1M2	10 Minutes			Minutes				Alarm 1 Minutes	00–59
09h	A1M3	12/24	AM/PM	10 Hour	Hour				Alarm 1 Hours	1–12 + AM/PM 00–23
			20 Hour							
0Ah	A1M4	DY/DT	10 Date		Day				Alarm 1 Day	1–7
					Date				Alarm 1 Date	1–31
0Bh	A2M2	10 Minutes			Minutes				Alarm 2 Minutes	00–59
0Ch	A2M3	12/24	AM/PM	10 Hour	Hour				Alarm 2 Hours	1–12 + AM/PM 00–23
			20 Hour							
0Dh	A2M4	DY/DT	10 Date		Day				Alarm 2 Day	1–7
					Date				Alarm 2 Date	1–31
0Eh	EOSC	BBSQW	CONV	RS2	RS1	INTCN	A2IE	A1IE	Control	—
0Fh	OSF	0	0	0	EN32kHz	BSY	A2F	A1F	Control/Status	—
10h	SIGN	DATA	DATA	DATA	DATA	DATA	DATA	DATA	Aging Offset	—
11h	SIGN	DATA	DATA	DATA	DATA	DATA	DATA	DATA	MSB of Temp	—
12h	DATA	DATA	0	0	0	0	0	0	LSB of Temp	—

Figure 1. Timekeeping Registers

Note: Unless otherwise specified, the registers' state is not defined when power is first applied.

Address Map

Figure 1 shows the address map for the DS3231 timekeeping registers. During a multibyte access, when the address pointer reaches the end of the register space (12h), it wraps around to location 00h. On an I2C START or address pointer incrementing to location 00h, the current time is transferred to a second set of registers. The time information is read from these secondary registers, while the clock may continue to run. This eliminates the need to reread the registers in case the main registers update during a read.

I2C Interface

The I2C interface is accessible whenever either V_{CC} or V_{BAT} is at a valid level. If a microcontroller connected

to the DS3231 resets because of a loss of V_{CC} or other event, it is possible that the microcontroller and DS3231 I2C communications could become unsynchronized, e.g., the microcontroller resets while reading data from the DS3231. When the microcontroller resets, the DS3231 I2C interface may be placed into a known state by toggling SCL until SDA is observed to be at a high level. At that point the microcontroller should pull SDA low while SCL is high, generating a START condition.

Clock and Calendar

The time and calendar information is obtained by reading the appropriate register bytes. Figure 1 illustrates the RTC registers. The time and calendar data are set or initialized by writing the appropriate register bytes. The contents of the time and calendar registers are in the binary-coded

decimal (BCD) format. The DS3231 can be run in either 12-hour or 24-hour mode. Bit 6 of the hours register is defined as the 12- or 24-hour mode select bit. When high, the 12-hour mode is selected. In the 12-hour mode, bit 5 is the AM/PM bit with logic-high being PM. In the 24-hour mode, bit 5 is the 20-hour bit (20–23 hours). The century bit (bit 7 of the month register) is toggled when the years register overflows from 99 to 00.

The day-of-week register increments at midnight. Values that correspond to the day of week are user-defined but must be sequential (i.e., if 1 equals Sunday, then 2 equals Monday, and so on). Illogical time and date entries result in undefined operation.

When reading or writing the time and date registers, secondary (user) buffers are used to prevent errors when the internal registers update. When reading the time and date registers, the user buffers are synchronized to the internal registers on any START and when the register pointer rolls over to zero. The time information is read from these secondary registers, while the clock continues to run. This eliminates the need to reread the registers in case the main registers update during a read.

The countdown chain is reset whenever the seconds register is written. Write transfers occur on the acknowledge from the DS3231. Once the countdown chain is reset, to avoid rollover issues the remaining time and date registers must be written within 1 second. The 1Hz square-wave output, if enabled, transitions high 500ms after the seconds data transfer, provided the oscillator is already running.

Alarms

The DS3231 contains two time-of-day/date alarms. Alarm 1 can be set by writing to registers 07h to 0Ah. Alarm 2 can be set by writing to registers 0Bh to 0Dh. The alarms can be programmed (by the alarm enable and INTCN bits of the control register) to activate the $\overline{\text{INT}}/\text{SQW}$ output on an alarm match condition. Bit 7 of each of the time-of-day/date alarm registers are mask bits (Table 2). When all the mask bits for each alarm are logic 0, an alarm only occurs when the values in the timekeeping registers match the corresponding values stored in the time-of-day/date alarm registers. The alarms can also be programmed to repeat every second, minute, hour, day, or date. Table 2 shows the possible settings. Configurations not listed in the table will result in illogical operation.

The $\text{DY}/\overline{\text{DT}}$ bits (bit 6 of the alarm day/date registers) control whether the alarm value stored in bits 0 to 5 of that register reflects the day of the week or the date of the month. If $\text{DY}/\overline{\text{DT}}$ is written to logic 0, the alarm will be the result of a match with date of the month. If $\text{DY}/\overline{\text{DT}}$ is written to logic 1, the alarm will be the result of a match with day of the week.

When the RTC register values match alarm register settings, the corresponding Alarm Flag 'A1F' or 'A2F' bit is set to logic 1. If the corresponding Alarm Interrupt Enable 'A1IE' or 'A2IE' is also set to logic 1 and the INTCN bit is set to logic 1, the alarm condition will activate the $\overline{\text{INT}}/\text{SQW}$ signal. The match is tested on the once-per-second update of the time and date registers.

Table 2. Alarm Mask Bits

$\text{DY}/\overline{\text{DT}}$	ALARM 1 REGISTER MASK BITS (BIT 7)				ALARM RATE
	A1M4	A1M3	A1M2	A1M1	
X	1	1	1	1	Alarm once per second
X	1	1	1	0	Alarm when seconds match
X	1	1	0	0	Alarm when minutes and seconds match
X	1	0	0	0	Alarm when hours, minutes, and seconds match
0	0	0	0	0	Alarm when date, hours, minutes, and seconds match
1	0	0	0	0	Alarm when day, hours, minutes, and seconds match
$\text{DY}/\overline{\text{DT}}$	ALARM 2 REGISTER MASK BITS (BIT 7)			ALARM RATE	
	A2M4	A2M3	A2M2		
X	1	1	1	Alarm once per minute (00 seconds of every minute)	
X	1	1	0	Alarm when minutes match	
X	1	0	0	Alarm when hours and minutes match	
0	0	0	0	Alarm when date, hours, and minutes match	
1	0	0	0	Alarm when day, hours, and minutes match	

Control Register (0Eh)

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
NAME:	$\overline{\text{EOSC}}$	BBSQW	CONV	RS2	RS1	INTCN	A2IE	A1IE
POR:	0	0	0	1	1	1	0	0

Special-Purpose Registers

The DS3231 has two additional registers (control and status) that control the real-time clock, alarms, and square-wave output.

Control Register (0Eh)

Bit 7: Enable Oscillator ($\overline{\text{EOSC}}$). When set to logic 0, the oscillator is started. When set to logic 1, the oscillator is stopped when the DS3231 switches to V_{BAT} . This bit is clear (logic 0) when power is first applied. When the DS3231 is powered by V_{CC} , the oscillator is always on regardless of the status of the $\overline{\text{EOSC}}$ bit. When $\overline{\text{EOSC}}$ is disabled, all register data is static.

Bit 6: Battery-Backed Square-Wave Enable (BBSQW). When set to logic 1 with $\text{INTCN} = 0$ and $V_{\text{CC}} < V_{\text{PF}}$, this bit enables the square wave. When BBSQW is logic 0, the $\overline{\text{INT/SQW}}$ pin goes high impedance when $V_{\text{CC}} < V_{\text{PF}}$. This bit is disabled (logic 0) when power is first applied.

Bit 5: Convert Temperature (CONV). Setting this bit to 1 forces the temperature sensor to convert the temperature into digital code and execute the TCXO algorithm to update the capacitance array to the oscillator. This can only happen when a conversion is not already in progress. The user should check the status bit BSY before forcing the controller to start a new TCXO execution. A user-initiated temperature conversion does not affect the internal 64-second update cycle.

A user-initiated temperature conversion does not affect the BSY bit for approximately 2ms. The CONV bit remains at a 1 from the time it is written until the conversion is finished, at which time both CONV and BSY go to 0. The CONV bit should be used when monitoring the status of a user-initiated conversion.

Bits 4 and 3: Rate Select (RS2 and RS1). These bits control the frequency of the square-wave output when

the square wave has been enabled. The following table shows the square-wave frequencies that can be selected with the RS bits. These bits are both set to logic 1 (8.192kHz) when power is first applied.

SQUARE-WAVE OUTPUT FREQUENCY

RS2	RS1	SQUARE-WAVE OUTPUT FREQUENCY
0	0	1Hz
0	1	1.024kHz
1	0	4.096kHz
1	1	8.192kHz

Bit 2: Interrupt Control (INTCN). This bit controls the $\overline{\text{INT/SQW}}$ signal. When the INTCN bit is set to logic 0, a square wave is output on the $\overline{\text{INT/SQW}}$ pin. When the INTCN bit is set to logic 1, then a match between the time-keeping registers and either of the alarm registers activates the $\overline{\text{INT/SQW}}$ output (if the alarm is also enabled). The corresponding alarm flag is always set regardless of the state of the INTCN bit. The INTCN bit is set to logic 1 when power is first applied.

Bit 1: Alarm 2 Interrupt Enable (A2IE). When set to logic 1, this bit permits the alarm 2 flag (A2F) bit in the status register to assert $\overline{\text{INT/SQW}}$ (when $\text{INTCN} = 1$). When the A2IE bit is set to logic 0 or INTCN is set to logic 0, the A2F bit does not initiate an interrupt signal. The A2IE bit is disabled (logic 0) when power is first applied.

Bit 0: Alarm 1 Interrupt Enable (A1IE). When set to logic 1, this bit permits the alarm 1 flag (A1F) bit in the status register to assert $\overline{\text{INT/SQW}}$ (when $\text{INTCN} = 1$). When the A1IE bit is set to logic 0 or INTCN is set to logic 0, the A1F bit does not initiate the $\overline{\text{INT/SQW}}$ signal. The A1IE bit is disabled (logic 0) when power is first applied.

Status Register (0Fh)

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
NAME:	OSF	0	0	0	EN32kHz	BSY	A2F	A1F
POR:	1	0	0	0	1	X	X	X

Status Register (0Fh)

Bit 7: Oscillator Stop Flag (OSF). A logic 1 in this bit indicates that the oscillator either is stopped or was stopped for some period and may be used to judge the validity of the timekeeping data. This bit is set to logic 1 any time that the oscillator stops. The following are examples of conditions that can cause the OSF bit to be set:

- 1) The first time power is applied.
- 2) The voltages present on both V_{CC} and V_{BAT} are insufficient to support oscillation.
- 3) The \overline{EOSC} bit is turned off in battery-backed mode.
- 4) External influences on the crystal (i.e., noise, leakage, etc.).

This bit remains at logic 1 until written to logic 0.

Bit 3: Enable 32kHz Output (EN32kHz). This bit controls the status of the 32kHz pin. When set to logic 1, the 32kHz pin is enabled and outputs a 32.768kHz square-wave signal. When set to logic 0, the 32kHz pin goes to a high-impedance state. The initial power-up state of this bit is logic 1, and a 32.768kHz square-wave signal appears at the 32kHz pin after a power source is applied to the DS3231 (if the oscillator is running).

Bit 2: Busy (BSY). This bit indicates the device is busy executing TCXO functions. It goes to logic 1 when the conversion signal to the temperature sensor is asserted and then is cleared when the device is in the 1-minute idle state.

Bit 1: Alarm 2 Flag (A2F). A logic 1 in the alarm 2 flag bit indicates that the time matched the alarm 2 registers. If the A2IE bit is logic 1 and the INTCN bit is set to logic 1, the \overline{INT}/SQW pin is also asserted. A2F is cleared when written to logic 0. This bit can only be written to logic 0. Attempting to write to logic 1 leaves the value unchanged.

Bit 0: Alarm 1 Flag (A1F). A logic 1 in the alarm 1 flag bit indicates that the time matched the alarm 1 registers. If the

A1IE bit is logic 1 and the INTCN bit is set to logic 1, the \overline{INT}/SQW pin is also asserted. A1F is cleared when written to logic 0. This bit can only be written to logic 0. Attempting to write to logic 1 leaves the value unchanged.

Aging Offset

The aging offset register takes a user-provided value to add to or subtract from the codes in the capacitance array registers. The code is encoded in two's complement, with bit 7 representing the sign bit. One LSB represents one small capacitor to be switched in or out of the capacitance array at the crystal pins. The aging offset register capacitance value is added or subtracted from the capacitance value that the device calculates for each temperature compensation. The offset register is added to the capacitance array during a normal temperature conversion, if the temperature changes from the previous conversion, or during a manual user conversion (setting the CONV bit). To see the effects of the aging register on the 32kHz output frequency immediately, a manual conversion should be started after each aging register change.

Positive aging values add capacitance to the array, slowing the oscillator frequency. Negative values remove capacitance from the array, increasing the oscillator frequency.

The change in ppm per LSB is different at different temperatures. The frequency vs. temperature curve is shifted by the values used in this register. At +25°C, one LSB typically provides about 0.1ppm change in frequency.

Use of the aging register is not needed to achieve the accuracy as defined in the EC tables, but could be used to help compensate for aging at a given temperature. See the *Typical Operating Characteristics* section for a graph showing the effect of the register on accuracy over temperature.

Aging Offset (10h)

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
NAME:	Sign	Data	Data	Data	Data	Data	Data	Data
POR:	0	0	0	0	0	0	0	0

Temperature Register (Upper Byte) (11h)

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
NAME:	Sign	Data	Data	Data	Data	Data	Data	Data
POR:	0	0	0	0	0	0	0	0

Temperature Register (Lower Byte) (12h)

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
NAME:	Data	Data	0	0	0	0	0	0
POR:	0	0	0	0	0	0	0	0

Temperature Registers (11h–12h)

Temperature is represented as a 10-bit code with a resolution of 0.25°C and is accessible at location 11h and 12h. The temperature is encoded in two's complement format. The upper 8 bits, the integer portion, are at location 11h and the lower 2 bits, the fractional portion, are in the upper nibble at location 12h. For example, 00011001 01b = +25.25°C. Upon power reset, the registers are set to a default temperature of 0°C and the controller starts a temperature conversion. The temperature is read on initial application of V_{CC} or I²C access on V_{BAT} and once every 64 seconds afterwards. The temperature registers are updated after each user-initiated conversion and on every 64-second conversion. The temperature registers are read-only.

I²C Serial Data Bus

The DS3231 supports a bidirectional I²C bus and data transmission protocol. A device that sends data onto the bus is defined as a transmitter and a device receiving data is defined as a receiver. The device that controls the message is called a master. The devices that are controlled by the master are slaves. The bus must be controlled by a master device that generates the serial clock (SCL), controls the bus access, and generates the START and STOP conditions. The DS3231 operates as a slave on the I²C bus. Connections to the bus are made through the SCL input and open-drain SDA I/O lines. Within the bus specifications, a standard mode (100kHz maximum clock rate) and a fast mode (400kHz maximum clock rate) are defined. The DS3231 works in both modes.

The following bus protocol has been defined (Figure 2):

- Data transfer may be initiated only when the bus is not busy.
- During data transfer, the data line must remain stable whenever the clock line is high. Changes in the data

line while the clock line is high are interpreted as control signals.

Accordingly, the following bus conditions have been defined:

Bus not busy: Both data and clock lines remain high.

START data transfer: A change in the state of the data line from high to low, while the clock line is high, defines a START condition.

STOP data transfer: A change in the state of the data line from low to high, while the clock line is high, defines a STOP condition.

Data valid: The state of the data line represents valid data when, after a START condition, the data line is stable for the duration of the high period of the clock signal. The data on the line must be changed during the low period of the clock signal. There is one clock pulse per bit of data.

Each data transfer is initiated with a START condition and terminated with a STOP condition. The number of data bytes transferred between the START and the STOP conditions is not limited, and is determined by the master device. The information is transferred byte-wise and each receiver acknowledges with a ninth bit.

Acknowledge: Each receiving device, when addressed, is obliged to generate an acknowledge after the reception of each byte. The master device must generate an extra clock pulse, which is associated with this acknowledge bit.

A device that acknowledges must pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable low during the high period of the acknowledge-related clock pulse. Of course, setup and hold times must be taken into account. A master must signal an end of data to the slave by not generat-

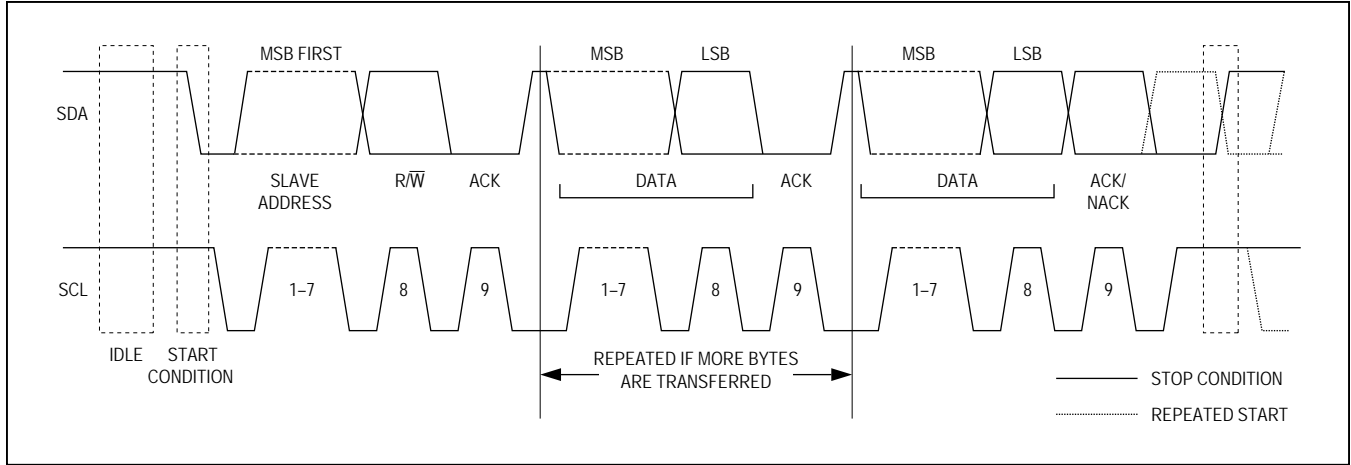


Figure 2. I²C Data Transfer Overview

ing an acknowledge bit on the last byte that has been clocked out of the slave. In this case, the slave must leave the data line high to enable the master to generate the STOP condition.

Figures 3 and 4 detail how data transfer is accomplished on the I²C bus. Depending upon the state of the R/W bit, two types of data transfer are possible:

Data transfer from a master transmitter to a slave receiver. The first byte transmitted by the master

is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte. Data is transferred with the most significant bit (MSB) first.

Data transfer from a slave transmitter to a master receiver. The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. Next follows a number of data bytes transmitted by the slave to the master. The master returns an acknowledge bit after all received bytes other than the

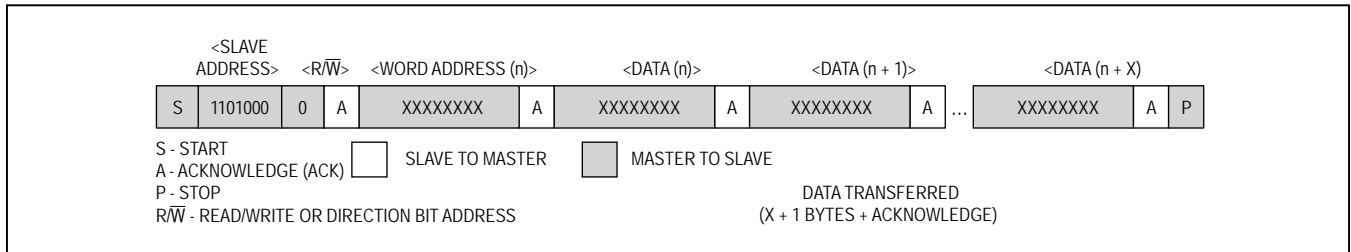


Figure 3. Data Write—Slave Receiver Mode

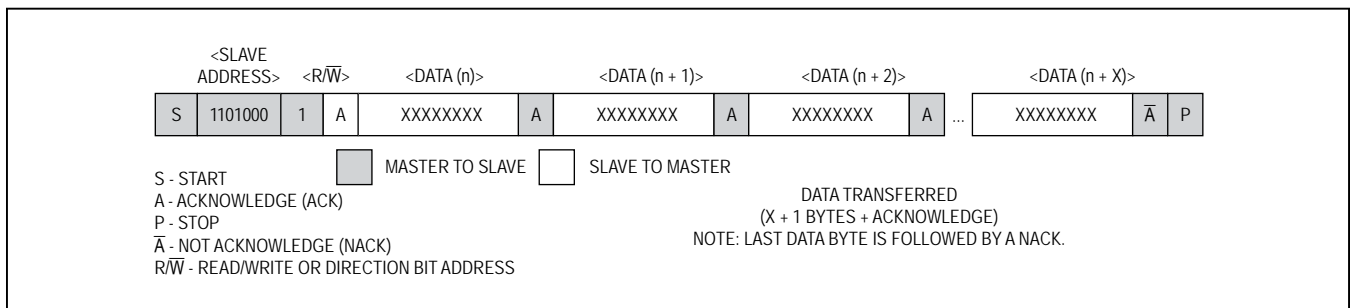


Figure 4. Data Read—Slave Transmitter Mode

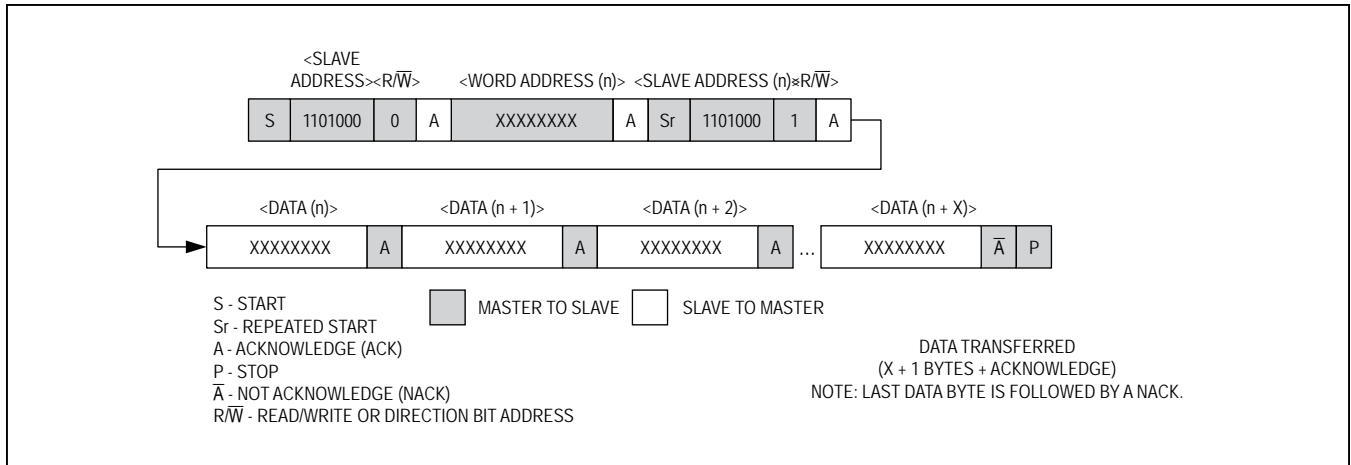


Figure 5. Data Write/Read (Write Pointer, Then Read)—Slave Receive and Transmit

last byte. At the end of the last received byte, a not acknowledge is returned.

The master device generates all the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Since a repeated START condition is also the beginning of the next serial transfer, the bus will not be released. Data is transferred with the most significant bit (MSB) first.

The DS3231 can operate in the following two modes:

Slave receiver mode (DS3231 write mode): Serial data and clock are received through SDA and SCL. After each byte is received, an acknowledge bit is transmitted. START and STOP conditions are recognized as the beginning and end of a serial transfer. Address recognition is performed by hardware after reception of the slave address and direction bit. The slave address byte is the first byte received after the master generates the START condition. The slave address byte contains the 7-bit DS3231 address, which is 1101000, followed by the direction bit (R/W), which is 0 for a write. After receiving and decoding the slave address byte, the DS3231 outputs an acknowledge on SDA. After the DS3231 acknowledges the slave address + write bit, the master transmits a word address to the DS3231. This sets the register pointer on the DS3231, with the DS3231 acknowledging the

transfer. The master may then transmit zero or more bytes of data, with the DS3231 acknowledging each byte received. The register pointer increments after each data byte is transferred. The master generates a STOP condition to terminate the data write.

Slave transmitter mode (DS3231 read mode): The first byte is received and handled as in the slave receiver mode. However, in this mode, the direction bit indicates that the transfer direction is reversed. Serial data is transmitted on SDA by the DS3231 while the serial clock is input on SCL. START and STOP conditions are recognized as the beginning and end of a serial transfer. Address recognition is performed by hardware after reception of the slave address and direction bit. The slave address byte is the first byte received after the master generates a START condition. The slave address byte contains the 7-bit DS3231 address, which is 1101000, followed by the direction bit (R/W), which is 1 for a read. After receiving and decoding the slave address byte, the DS3231 outputs an acknowledge on SDA. The DS3231 then begins to transmit data starting with the register address pointed to by the register pointer. If the register pointer is not written to before the initiation of a read mode, the first address that is read is the last one stored in the register pointer. The DS3231 must receive a not acknowledge to end a read.

DS3231

Extremely Accurate I2C-Integrated RTC/TCXO/Crystal

Handling, PCB Layout, and Assembly

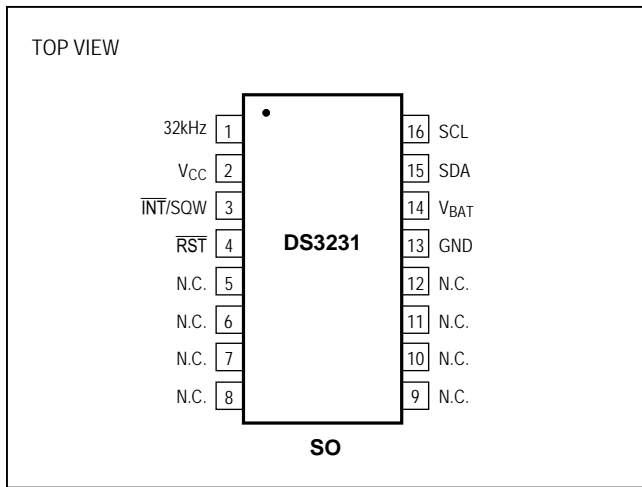
The DS3231 package contains a quartz tuning-fork crystal. Pick-and-place equipment can be used, but precautions should be taken to ensure that excessive shocks are avoided. Ultrasonic cleaning should be avoided to prevent damage to the crystal.

Avoid running signal traces under the package, unless a ground plane is placed between the package and the

signal line. All N.C. (no connect) pins must be connected to ground.

Moisture-sensitive packages are shipped from the factory dry packed. Handling instructions listed on the package label must be followed to prevent damage during reflow. Refer to the IPC/JEDEC J-STD-020 standard for moisture-sensitive device (MSD) classifications and reflow profiles. Exposure to reflow is limited to 2 times maximum.

Pin Configuration



Chip Information

SUBSTRATE CONNECTED TO GROUND
PROCESS: CMOS

Ordering Information

PART	TEMP RANGE	PIN-PACKAGE
DS3231S#	0°C to +70°C	16 SO
DS3231SN#	-40°C to +85°C	16 SO

#Denotes an RoHS-compliant device that may include lead (Pb) that is exempt under RoHS requirements. The lead finish is JESD97 category e3, and is compatible with both lead-based and lead-free soldering processes. A “#” anywhere on the top mark denotes an RoHS-compliant device.

Package Information

For the latest package outline information and land patterns (footprints), go to www.maximintegrated.com/packages. Note that a “+”, “#”, or “-” in the package code indicates RoHS status only. Package drawings may show a different suffix character, but the drawing pertains to the package regardless of RoHS status.

PACKAGE TYPE	PACKAGE CODE	OUTLINE NO.	LAND PATTERN NO.
16 SO	W16#H2	21-0042	90-0107

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	1/05	Initial release.	—
1	2/05	Changed Digital Temp Sensor Output from $\pm 2^{\circ}\text{C}$ to $\pm 3^{\circ}\text{C}$.	1, 3
		Updated <i>Typical Operating Circuit</i> .	1
		Changed $T_A = -40^{\circ}\text{C}$ to $+85^{\circ}\text{C}$ to $T_A = T_{\text{MIN}}$ to T_{MAX} .	2, 3, 4
		Updated <i>Block Diagram</i> .	8
2	6/05	Added "UL Recognized" to Features; added lead-free packages and removed S from top mark info in <i>Ordering Information</i> table; added ground connections to the N.C. pin in the <i>Typical Operating Circuit</i> .	1
		Added "noncondensing" to operating temperature range; changed V_{PF} MIN from 2.35V to 2.45V.	2
		Added aging offset specification.	3
		Relabeled TOC4.	7
		Added arrow showing input on X1 in the <i>Block Diagram</i> .	8
		Updated pin descriptions for V_{CC} and V_{BAT} .	9
		Added the I ² C Interface section.	10
		<i>Figure 1</i> : Added sign bit to aging and temperature registers; added MSB and LSB.	11
		Corrected title for rate select bits frequency table.	13
		Added note that frequency stability over temperature spec is with aging offset register = 00h; changed bit 7 from Data to Sign (Crystal Aging Offset Register).	14
		Changed bit 7 from Data to Sign (Temperature Register); correct pin definitions in <i>I²C Serial Data Bus</i> section.	15
		Modified the <i>Handing</i> , <i>PC Board Layout</i> , and <i>Assembly</i> section to refer to J-STD-020 for reflow profiles for lead-free and leaded packages.	17
3	11/05	Changed lead-free packages to RoHS-compliant packages.	1
4	10/06	Changed $\overline{\text{RST}}$ and UL bullets in <i>Features</i> .	1
		Changed EC condition " $V_{\text{CC}} > V_{\text{BAT}}$ " to " $V_{\text{CC}} = \text{Active Supply}$ (see Table 1)."	2, 3
		Modified Note 12 to correct t_{REC} operation.	6
		Added various conditions text to TOCs 1, 2, and 3.	7
		Added text to pin descriptions for 32kHz, V_{CC} , and $\overline{\text{RST}}$.	9
		Table 1: Changed column heading "Powered By" to "Active Supply"; changed "applied" to "exceeds V_{PF} " in the <i>Power Control</i> section.	10
		Indicated BBSQW applies to both SQW and interrupts; simplified temp convert description (bit 5); added "output" to INT/SQW (bit 2).	13
		Changed the <i>Crystal Aging</i> section to the <i>Aging Offset</i> section; changed "this bit indicates" to "this bit controls" for the enable 32kHz output bit.	14
5	4/08	Added Warning note to EC table notes; updated Note 12.	6
		Updated the <i>Typical Operating Characteristics</i> graphs.	7
		In the <i>Power Control</i> section, added information about the POR state of the time and date registers; in the <i>Real-Time Clock</i> section, added to the description of the RST function.	10
		In Figure 1, corrected the months date range for 04h from 00–31 to 01–31.	11

Revision History (continued)

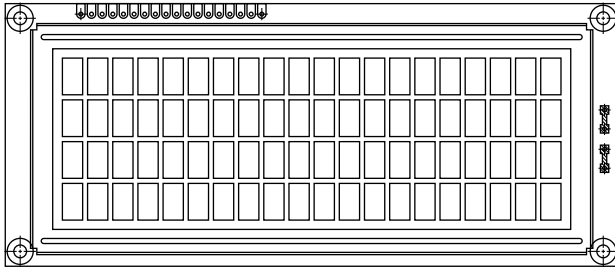
REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
6	10/08	Updated the <i>Typical Operating Circuit</i> .	1
		Removed the V_{PU} parameter from the <i>Recommended DC Operating Conditions</i> table and added verbiage about the pullup to the <i>Pin Description</i> table for I ² C/SQW, SDA, and SCL.	2, 9
		Added the Delta Time and Frequency vs. Temperature graph in the <i>Typical Operating Characteristics</i> section.	7
		Updated the <i>Block Diagram</i> .	8
		Added the V_{BAT} Operation section, improved some sections of text for the <i>32kHz TCXO</i> and <i>Pushbutton Reset Function</i> sections.	10
		Added the register bit POR values to the register tables.	13, 14, 15
		Updated the <i>Aging Offset</i> and <i>Temperature Registers (11h–12h)</i> sections.	14, 15
		Updated the I ² C timing diagrams (Figures 3, 4, and 5).	16, 17
7	3/10	Removed the “S” from the top mark in the <i>Ordering Information</i> table and the <i>Pin Configuration</i> to match the packaging engineering marking specification.	1, 18
8	7/10	Updated the <i>Typical Operating Circuit</i> ; removed the “Top Mark” column from the <i>Ordering Information</i> ; in the <i>Absolute Maximum Ratings</i> section, added the theta-JA and theta-JC thermal resistances and Note 1, and changed the soldering temperature to +260°C (lead(Pb)-free) and +240°C (leaded); updated the functional description of the V_{BAT} pin in the <i>Pin Description</i> ; changed the timekeeping registers 02h, 09h, and 0Ch to “20 Hour” in Bit 5 of Figure 1; updated the BBSQW bit description in the <i>Control Register (0Eh)</i> section; added the land pattern no. to the <i>Package Information</i> table.	1, 2, 3, 4, 6, 9, 11, 12, 13, 18
9	1/13	Updated <i>Absolute Maximum Ratings</i> , and last paragraph in <i>Power Control</i> section	2, 10
10	3/15	Revised <i>Benefits and Features</i> section.	1

For pricing, delivery, and ordering information, please contact Maxim Direct at 1-888-629-4642, or visit Maxim Integrated's website at www.maximintegrated.com.

Maxim Integrated cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim Integrated product. No circuit patent licenses are implied. Maxim Integrated reserves the right to change the circuitry and specifications without notice at any time. The parametric values (min and max limits) shown in the *Electrical Characteristics* table are guaranteed. Other parametric values quoted in this data sheet are provided for guidance.

Anexo 5. Datasheet LCD 20x4 Lcd-020n004l

20 x 4 Character LCD



FEATURES

- Type: Character
- Display format: 20 x 4 characters
- Built-in controller: ST 7066 (or equivalent)
- Duty cycle: 1/16
- 5 x 8 dots includes cursor
- + 5 V power supply (also available for + 3 V)
- LED can be driven by pin 1, pin 2, pin 15, pin 16 or A and K
- N.V. optional for + 3 V power supply
- Material categorization: For definitions of compliance please see www.vishay.com/doc?99912


RoHS
COMPLIANT

MECHANICAL DATA		
ITEM	STANDARD VALUE	UNIT
Module Dimension	146.0 x 62.5	mm
Viewing Area	123.5 x 43.0	
Dot Size	0.92 x 1.10	
Dot Pitch	0.98 x 1.16	
Mounting Hole	139.0 x 55.5	
Character Size	4.84 x 9.22	

ABSOLUTE MAXIMUM RATINGS					
ITEM	SYMBOL	STANDARD VALUE			UNIT
		MIN.	TYP.	MAX.	
Power Supply	V_{DD} to V_{SS}	- 0.3	-	7.0	V
Input Voltage	V_I	- 0.3	-	V_{DD}	

Note

- $V_{SS} = 0$ V, $V_{DD} = 5.0$ V

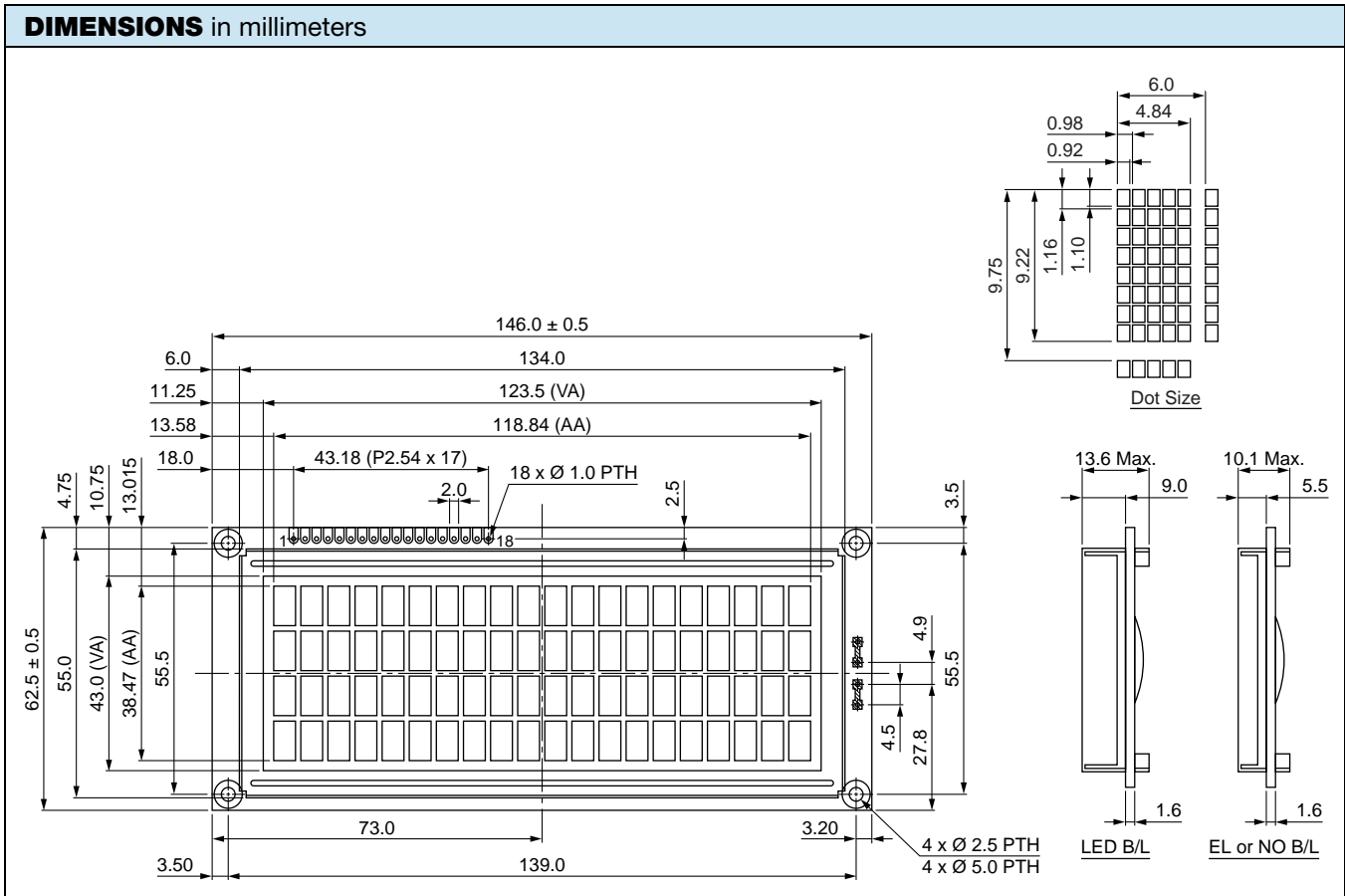
ELECTRICAL CHARACTERISTICS						
ITEM	SYMBOL	CONDITION	STANDARD VALUE			UNIT
			MIN.	TYP.	MAX.	
Input Voltage	V_{DD}	$V_{DD} = + 5$ V	4.7	5.0	5.3	V
		$V_{DD} = + 3$ V	2.7	3.0	5.3	
Supply Current	I_{DD}	$V_{DD} = + 5$ V	-	8.0	10.0	mA
Recommended LC Driving Voltage for Normal Temperature Version Module	V_{DD} to V_0	- 20 °C	5.0	5.1	5.7	V
		0 °C	4.6	4.8	5.2	
		25 °C	4.1	4.5	4.7	
		50 °C	3.9	4.2	4.5	
		70 °C	3.7	3.9	4.3	
LED Forward Voltage	V_F	25 °C	-	4.2	4.6	V
LED Forward Current	I_F	25 °C	-	540	1080	mA
EL Power Supply Current	I_{EL}	$V_{EL} = 110$ V _{AC} , 400 Hz	-	-	5.0	mA

OPTIONS									
PROCESS COLOR						BACKLIGHT			
TN	STN Gray	STN Yellow	STN Blue	FSTN B&W	STN Color	None	LED	EL	CCFL
x	x	x	x	x		x	x	x	

For detailed information, please see the "Product Numbering System" document.

DISPLAY CHARACTER ADDRESS CODE																				
Display Position																				
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
DD RAM Address	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13
DD RAM Address	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53
DD RAM Address	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27
DD RAM Address	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67

INTERFACE PIN FUNCTION		
PIN NO.	SYMBOL	FUNCTION
1	V_{SS}	Ground
2	V_{DD}	+ 3 V or + 5 V
3	V_0	Contrast adjustment
4	RS	H/L register select signal
5	R/\overline{W}	H/L read/write signal
6	E	H → L enable signal
7	DB0	H/L data bus line
8	DB1	H/L data bus line
9	DB2	H/L data bus line
10	DB3	H/L data bus line
11	DB4	H/L data bus line
12	DB5	H/L data bus line
13	DB6	H/L data bus line
14	DB7	H/L data bus line
15	A	Power supply for LED (4.2 V)
16	K	Power supply for B/L (0 V)
17	NC/ V_{EE}	NC or negative voltage output
18	NC	NC connection





Disclaimer

ALL PRODUCT, PRODUCT SPECIFICATIONS AND DATA ARE SUBJECT TO CHANGE WITHOUT NOTICE TO IMPROVE RELIABILITY, FUNCTION OR DESIGN OR OTHERWISE.

Vishay Intertechnology, Inc., its affiliates, agents, and employees, and all persons acting on its or their behalf (collectively, "Vishay"), disclaim any and all liability for any errors, inaccuracies or incompleteness contained in any datasheet or in any other disclosure relating to any product.

Vishay makes no warranty, representation or guarantee regarding the suitability of the products for any particular purpose or the continuing production of any product. To the maximum extent permitted by applicable law, Vishay disclaims (i) any and all liability arising out of the application or use of any product, (ii) any and all liability, including without limitation special, consequential or incidental damages, and (iii) any and all implied warranties, including warranties of fitness for particular purpose, non-infringement and merchantability.

Statements regarding the suitability of products for certain types of applications are based on Vishay's knowledge of typical requirements that are often placed on Vishay products in generic applications. Such statements are not binding statements about the suitability of products for a particular application. It is the customer's responsibility to validate that a particular product with the properties described in the product specification is suitable for use in a particular application. Parameters provided in datasheets and / or specifications may vary in different applications and performance may vary over time. All operating parameters, including typical parameters, must be validated for each customer application by the customer's technical experts. Product specifications do not expand or otherwise modify Vishay's terms and conditions of purchase, including but not limited to the warranty expressed therein.

Except as expressly indicated in writing, Vishay products are not designed for use in medical, life-saving, or life-sustaining applications or for any other application in which the failure of the Vishay product could result in personal injury or death. Customers using or selling Vishay products not expressly indicated for use in such applications do so at their own risk. Please contact authorized Vishay personnel to obtain written terms and conditions regarding products designed for such applications.

No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document or by any conduct of Vishay. Product names and markings noted herein may be trademarks of their respective owners.

Anexo 6. Datasheet I2C-LCD-Interface

Datasheet

I2C 1602 Serial LCD Module



Product features:

The I2C 1602 LCD module is a 2 line by 16 character display interfaced to an I2C daughter board. The I2C interface only requires 2 data connections, +5 VDC and GND to operate

For in depth information on I2C interface and history, visit: <http://www.wikipedia/wiki/i2c>

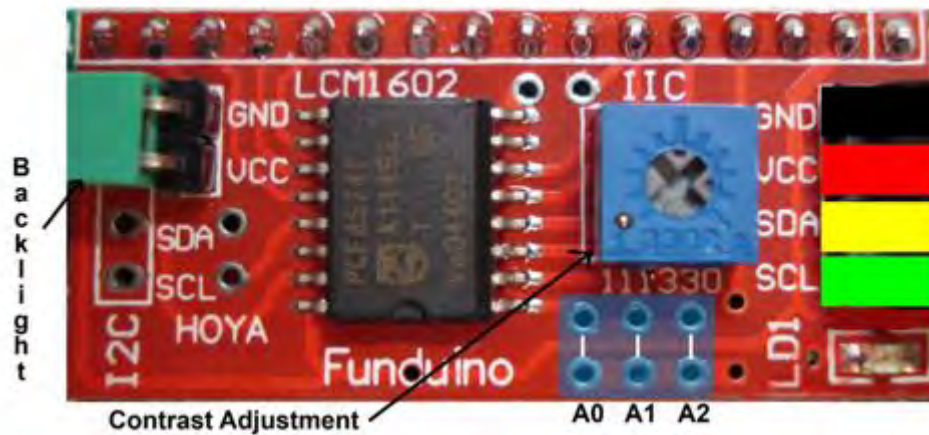
Specifications:

I2C Address Range	2 lines by 16 character 0x20 to 0x27 (Default=0x27, addressable)
Operating Voltage	5 Vdc
Backlight	White
Contrast	Adjustable by potentiometer on I2c interface
Size	80mm x 36mm x 20 mm
Viewable area	66mm x 16mm

Power:

The device is powered by a single 5Vdc connection.

Pinout Diagram:



Pin/Control Descriptions:

Pin #	Name	Type	Description
1	GND	Power	Supply & Logic ground
2	VCC	Power	Digital I/O 0 or RX (serial receive)
3	SDA	I/O	Serial Data line
4	SCL	CLK	Serial Clock line
A0	A0	Jumper	Optional address selection A0 - see below
A1	A1	Jumper	Optional address selection A1 - see below
A2	A2	Jumper	Optional address selection A2 - see below
Backlight		Jumper	Jumpered - enable backlight, Open - disable backlight
Contrast		Pot	Adiust for best viewing

Addressing:

A0	A1	A2	Address
Open	Open	Open	0x27
Jumper	Open	Open	0x26
Open	Jumper	Open	0x25
Jumper	Jumper	Open	0x24
Open	Open	Jumper	0x23
Jumper	Open	Jumper	0x22
Open	Jumper	Jumper	0x21
Jumper	Jumper	Jumper	0x20

Software:

Download the required LCD Arduino™ library for this device from:

<http://www.circuitattic.com/downloads/category/3-sample-code.html?download=9%3Aanother-i2c-library-easier-to-use>

Replace current liquid crystal library found in the Arduino library directory with the above
(Note: If you use the examples included with the library, be sure to change address to 0x27)

Simple example using library above.

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#if defined(ARDUINO) && ARDUINO >= 100
#define printByte(args) write(args);
#else
#define printByte(args) print(args, BYTE);
#endif
LiquidCrystal_I2C lcd(0x27,16,2); // set the LCD address to 0x27 for a
//chars and 2 line display
void setup()
{
    lcd.init(); // initialize the lcd
    lcd.backlight();
    lcd.clear();
    delay(100);
    for(int i = 0; i < 3; i++)
    {
        lcd.backlight();
        delay(250);
        lcd.noBacklight();
        delay(250);
    }
    lcd.backlight();
}

void loop()
{
    int x=0;
    lcd.clear();
    lcd.setCursor(2,0); //Start at character 0 on line 0
    lcd.print("Hello World");
    lcd.setCursor(0,1); //Start at character 0 on line 1
    lcd.print(" opencircuit.nl");
    delay(3000); //Wait 3 seconds
    lcd.clear();
    lcd.setCursor(0,0); //Start at character 0 on line 0
    lcd.print("Cursor Blink");
    lcd.blink();
    delay(2000);
    lcd.setCursor(0,0);
    lcd.print("Cursor noBlink");
    lcd.noBlink();
    delay(2000);
}
```



Plagiarism Checker X Originality Report

Similarity Found: 0%

Date: jueves, septiembre 06, 2018

Statistics: 0 words Plagiarized / 12337 Total words

Remarks: No Plagiarism Detected - Your Document is Healthy.

/ "Responsabilidad con pensamiento positivo" UNIVERSIDAD TECNOLÓGICA ISRAEL
TRABAJO DE TITULACIÓN EN OPCIÓN AL GRADO DE: INGENIERO EN ELECTRÓNICA
DIGITAL Y TELECOMUNICACIONES TEMA: AUTOMATIZACIÓN DE UN CONTENEDOR DE
CARGA PARA EL CONTROL DE CLIMA E ILUMINACIÓN DE CULTIVOS HIDROPÓNICOS.
AUTOR: FRANKLIN PATRICIO MONTAHUANO PACHECO TUTOR: Ing. Luis H. Montoya L.,
Msc.

QUITO, ECUADOR AÑO: 2018

UNIVERSIDAD TECNOLÓGICA ISRAEL APROBACIÓN DEL TUTOR En mi calidad de tutor del trabajo de titulación certifico: Que el trabajo de titulación "AUTOMATIZACIÓN DE UN CONTENEDOR DE CARGA PARA EL CONTROL DE CLIMA E ILUMINACIÓN DE CULTIVOS HIDROPÓNICOS", presentado por el Señor Franklin Patricio Montahuano Pacheco, estudiante de la carrera de Electrónica Digital y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación. Quito D.M. 16 de Agosto del 2018 TUTOR Ing. Luis H. Montoya L., Msc.

AGRADECIMIENTOS: Deseo que estas palabras que sirvan para mostrar el más profundo agradecimiento que siento hacia Dios, por guiarme toda la vida y traerme hasta este momento. También, quiero agradecer con todo mi corazón a José y Luz, quienes siempre han estado a mi lado en la felicidad y en la tristeza, a ellos a los que con mucho orgullo los llamo Papá y Mamá.

Tampoco voy a olvidar a toda mi familia, a mis hermanos Luz y Jorge, así como también a mi sobrino José David por aguantarme tanto en las noches de desvelo, mi malgenio, por brindarme siempre su apoyo y entenderme a pesar de todo. A mi novia Tania por ser durante mucho tiempo una pieza clave de apoyo, motivación en el estudio y superación personal. A mis 3 mejores amigos Santiago B.,

Santiago B, y Luis que Dios los puso en mi camino y planeo conservarlos para el resto de la vida. A mis compañeros y amigos José, Cristian, Iván y Mauricio por ser parte de este reto, que emprendimos hace casi 4 años y también por servirme de apoyo cuando los necesite. A Todos ustedes muchas gracias desde el fondo de mi corazón.

Patricio DEDICATORIA: Al principio me resultó un poco difícil, como suele suceder cuando realizas la mayoría de las cosas por primera vez, pero después de reflexionar, comprendí, que las personas a las que debo agradecer este trabajo son las que siempre estuvieron ahí y las llamo con mucho cariño y amor "mi familia". También quiero dedicar este trabajo a Dios, porque siempre me ha brindado la fortaleza y sabiduría, se lo dedico a todos y cada uno de los miembros de mi familia, que pese a todas las dificultades siempre han estado a mi lado, apoyándome y dándome la fuerza y valor para seguir adelante.

A mis padres, José y Luz que siempre me enseñaron e inculcaron principios morales, religiosos y hábitos de estudio. A mis hermanos, Jorge y Luz que han sido unas personas muy importantes en mi vida siempre motivándome como estudiante y persona, que siempre pasaron pendientes en el transcurso de mi carrera. A mi sobrino José David por ser mi mayor motivación para nunca rendirme en los estudios y así poder llegar a ser un ejemplo para él.

Patricio ÍNDICE DE CONTENIDO Resumen ix Abstract x INTRODUCCIÓN 1 Antecedentes 2 Invernaderos 2 Estructuras y control ambiental 3 Cultivos en invernadero con suelo 3 Cultivos hidropónicos 4 Formulación 10 Justificación: 11 Alcance: 12 OBJETIVOS 13 Objetivo general: 13 Objetivos específicos: 13 CAPÍTULO 1 15 FUNDAMENTOS TEÓRICOS 15 1.1. Marco teórico 15 1.2. Microcontrolador 17 1.2.1. Arduino Uno Wi-Fi 17 1.3. Dispositivos de entrada 18 Sensor de humedad relativa y temperatura DTH22 18 1.4.

Dispositivos Especiales 21 1.4.1. Reloj En tiempo real RTC 21 1.4.2. Modulo I2C 22 1.5.
Dispositivos de salida 23 CAPÍTULO 2 30 DISEÑO DEL PROGRAMA 30 2.1 Diagrama de
Flujo. 30 2.2. Máquina de estados finitos. 32 Capítulo 3 43 Diseño del Hardware 43 3.1.
Desarrollo 44 CAPÍTULO 4 50 PRUEBAS Y ANÁLISIS DE RESULTADOS 50 4.1 Pruebas 50
4.2.

Análisis de resultados 55 CONCLUSIONES 56 RECOMENDACIONES 58 Bibliografía 59
Anexo 1. Código Fuente: 61 Anexo 2. Datasheet ArduinoUno Wifi: 62 Anexo 3. Datasheet
DTH22 Digital Humidity-Temperature Sensor: 63 Anexo 4. Datasheet Real Time Clock
Rtc-Ds3231 64 Anexo 5. Datasheet LCD 20x4 Lcd-020n004l 65 Anexo 6. Datasheet
I2C-LCD-Interface 66

TABLA DE FIGURAS (ILUSTRACIONES): Figura 1. 1. Diagrama de bloques 16 Figura 1. 2. Arduino Uno Wi-Fi 17 Figura 1. 3.

Sensor de humedad relativa y temperatura DTH22. 18 Figura 1. 4. Esquema de conexión DTH22 21 Figura 1. 5. Módulo RTC de Arduino 22 Figura 1. 6. Módulo I2C de Arduino 22 Figura 1. 7. Distribución espectral de luz de los LED para crecimiento 25 Figura 1. 8. Plantas con Iluminación LED para crecimiento 26 Figura 1. 9. Pantalla LCD 26 Figura 1. 10. Módulo de relé +5V 28 Figura 1. 11. Relé normalmente abierto (NO).

28 Figura 1. 12. Relé normalmente cerrado (NC). 29 Figura 2. 13 Diagrama de flujo 31 Figura 2. 14. Sistema de control retroalimentado 32 Figura 2. 15. Definición de botones y pines 32 Figura 2. 16. Botones para el control Automático. 33 Figura 2. 17. Definición de los estados 33 Figura 2. 18. Flanco de bajada y subida. 34 Figura 2. 19. Rutina para controlar el flanco de subida. 34 Figura 2. 20. Pantalla principal de MENU 35 Figura 2. 21.

Estado inicial de HOME y SET TEMP. 35 Figura 2. 22. Estado inicial SET TEMP. 36 Figura 2. 23. MENU de la temperatura de operación 36 Figura 2. 24. Estado inicial SET HUM 37 Figura 2. 25. MENU de la humedad de operación. 37 Figura 2. 26. Estado inicial SET ILIM 38 Figura 2. 27. MENU del control de la iluminación. 38 Figura 2. 28. Estado inicial SET TIME 39 Figura 2. 29. MENU del control del tiempo de riego. 40 Figura 2. 30. Máquina de estados completa. 41 Figura 2. 31.

MENU del control del intervalo de tiempo para riego. 41 Figura 3. 32. Simulación en Proteus. 43 Figura 3. 33. circuito de prueba en protoboard. 44 Figura 3. 34. Cálculo del ancho de banda de una pista PCB 45 Figura 3. 35. Diseño de la PCB Fuente: 45 Figura 3. 36. PCB (Placa de circuito impreso). 46 Figura 3. 37. Soldadura de elementos en la PCB (Placa de circuito impreso). 47 Figura 3. 38. Montaje de elementos en la PCB (Placa de circuito impreso). 47 Figura 3. 39.

Montaje de los pulsadores en la placa acrílica. 48 Figura 3. 40. Armado del contenedor a escala. 49 Figura 3. 41. Proyecto finalizado. 49 Figura 4. 42. Prueba de funcionamiento del circuito una vez ensamblado. 50 Figura 4. 43. Comparación de valores medidos por el circuito. 51 Figura 4. 44. Luces LED para crecimiento. 52 Figura 4. 45. Planta creciendo con luz LED. 52 Figura 4. 46. Prueba de funcionamiento de la bomba de agua. 53 Figura 4. 47.

Prueba de funcionamiento del aire acondicionado. 53 Figura 4. 48. Crecimiento de hortalizas en ambiente controlado. 54 Figura 4. 49. Plantas cultivadas con control automático 54

CONTENIDO DE TABLAS 18 Tabla 1. 1. Características eléctricas del DTH22 19 Tabla 1. 2. Bits de datos del DTH22 20 Tabla 1. 3. Módulos y pines I2C 23 Tabla 1. 4. Contenido espectral de los LED para crecimiento 24 Tabla 1. 5.

Especificaciones de los LED's de crecimiento 25 Tabla 1. 6. Pines del LCD 20x4 27

Resumen El presente proyecto presenta el diseño e implementación de un sistema de control automático y manejo de cultivo hidropónico para la producción de cultivos dentro de contenedores, teniendo en cuenta que los resultados estén dentro del rango de producción esperado.

El sistema de control realiza el monitoreo de los parámetros y permite su visualización en una pantalla LCD. También entrega datos bidireccionales para la comunicación entre el microcontrolador y un computador, dando al usuario la capacidad de controlar de forma automática y manual todo el sistema.

El proceso del diseño del prototipo, construcción, programación y pruebas de laboratorio se detallan a lo largo del presente documento, así como las dificultades y desventajas que se pueden presentar en los cultivos tradicionales, pruebas y comparaciones en cuanto a crecimiento, rendimiento y calidad, finalizando con un detalle de conclusiones y recomendaciones.

Palabras Clave: Automatización Wifi Hidroponía Clima

Abstract In this project the design and implementation of a system of automatic control and management of hydroponic crops for the production within containers is presented, taking into account the results within the expected production range. The control system performs the monitoring of the parameters and allows it to be shown on a LCD screen.

It also delivers bidirectional data for communication between the microcontroller and a computer, giving the user the ability to control the entire system both automatically and manually. The process of prototype design, construction, programming and laboratory tests are detailed throughout the document, as well as the difficulties and disadvantages that may arise compared to traditional crops, tests and comparisons in terms of growth, performance and quality, ending with a detail of conclusions and recommendations.

Keywords:

Automation Wifi Hydroponics Weather

INTRODUCCIÓN El cultivo de hortalizas en invernaderos es una de las formas más comunes en todas las empresas agrícolas a nivel mundial. En una combinación con los invernaderos, la hidroponía se está volviendo cada vez más popular, especialmente en los países más desarrollados del mundo como Estados Unidos, Canadá, Japón y algunos otros países en Europa.

Esta tecnología es altamente productiva, ya que conserva el agua y la tierra lo que genera que se proteja el medio ambiente. La hidroponía es común para la producción de verduras, hierbas, cultivos verticales (tomate, pepino y pimientos). Mediante la utilización de sistemas de control automático se operan cientos de dispositivos dentro de un invernadero mediante la medición de parámetros de entrada, para mantener el entorno de crecimiento ideal.

La tecnología de producción de alimentos actualmente está cambiando rápidamente ya que los nuevos sistemas de control existentes producen rendimientos nunca alcanzados, por lo que se espera que en un futuro muy cercano su utilización sea a escala mundial (Dalrymple DG, 1973). La tecnología para la producción de alimentos en invernaderos ha avanzado muchísimo en los últimos 30 años.

La producción de alimentos en invernadero a menudo se denomina agricultura de ambiente controlado (Controlled Environment Agriculture) y por lo general va acompañada con la hidroponía. Los cultivos hidropónicos posiblemente son el método que más demanda tiene en la producción de la industria agrícola actual, ya que requiere habilidades básicas de agricultura.

Hay muchos tipos de sistemas hidropónicos para la producción agrícola en ambientes controlados, estos tienen lugar dentro de recintos diseñados para controlar el aire, temperatura, luz, agua, nutrición de las plantas y clima. Cada componente de la agricultura en ambientes controlados tiene la misma importancia, ya sea el diseño estructural, el control ambiental o el sistema de crecimiento.

No todos los sistemas son rentables ya que con bastante frecuencia se utilizan solo una o dos de las principales variables y si no se presta una atención adecuada, ningún sistema hidropónico resultaría económicamente sustentable. Antecedentes Invernaderos Los invernaderos a partir de los años 60 se han convertido en más que un protector de plantas, hoy en día se los pueden ver como una especie de fábricas de vegetales.

En casi todos los aspectos de la producción el sistema está 100% automatizado, ya que cuenta con un entorno artificial controlado por una computadora casi en su totalidad. Cuando el sistema de invernaderos se encuentra en un entorno de investigación en un

sistema totalmente cerrado se le denomina cámara de crecimiento o un fitotrón, que es un laboratorio equipado para estudiar las condiciones físicas y químicas en las que se desarrollan las plantas.

La agricultura ambiental controlada ha ganado muchísima importancia en el campo hortícola, no solo por la producción de cultivos vegetales y ornamentales, sino también en la producción de plántulas de plantas, ya sea desde la semilla o mediante procedimientos de cultivo de tejidos. En los últimos 15 años ha habido un interés creciente en el uso de técnicas hidropónicas para producir cultivos hortícolas en invernaderos.

El futuro crecimiento en invernadero mediante la agricultura ambiental controlada generará que la hidroponía se utilice para la producción de hortalizas a gran escala y depende en gran medida del desarrollo de sistemas de producción que sean más competitivos en términos de costos en comparación con la agricultura de campo abierto. La economía en la producción de alimentos en invernaderos es equilibrada en relación con el alto capital y los costos operacionales de los invernaderos, ya que es significativamente mayor la productividad de la agricultura de ambiente controlado en comparación con la agricultura de campo abierto (OFA).

(Jensen, 2017) Estructuras y control ambiental Para lograr un ambiente de cultivo más uniforme, sin fluctuaciones de temperatura rápidas, se está utilizando un mayor volumen de espacio dentro del área dedicada a un invernadero, donde hoy en día los soportes de las estructuras de invernadero alcanzan una altura de más de 5 m sobre el nivel del suelo. Los materiales más nuevos como los policarbonatos y los acrílicos se han vuelto mucho más comunes, aunque su popularidad aún no ha sido compensada por los altos costos de producción.

Hace algunos años había un gran énfasis en la investigación para el uso de energía solar como fuente de alimentación y calor del invernadero ya que es técnicamente factible, pero se ha demostrado que no es económica debido a los altos costos de recolección y almacenamiento. En los últimos diez años, ha habido un gran interés en el desarrollo de plantas de cogeneración, donde pequeñas plantas eléctricas utilizan el calor residual que desprenden los generadores eléctricos para calentar hortalizas de invernadero o agua para la producción de pescado.

Cualquiera que sea la fuente de energía, es importante conservarla dentro del invernadero. En el futuro los invernaderos tendrán techos retráctiles que permitirán la entrada de luz y ventilación si es deseado mediante sistemas de control automático. (Jensen, 2017).

Cultivos en invernadero con suelo El método estándar para cultivar vegetales de invernadero en todo el mundo está en el suelo y un agricultor exitoso generalmente tiene un buen conocimiento de horticultura, suelos, plantas, patología, entomología y fisiología vegetal, así como la capacidad de ingeniería para proporcionar el ambiente más adecuado para el crecimiento de la planta.

Muchos agricultores que hacen cultivos en invernadero, la mayoría de las veces fallan porque carecen de la educación y la capacitación adecuada. Además, un problema importante en los cultivos tradicionales son las enfermedades transmitidas por el suelo, plantas creciendo continuamente sin rotación de cultivos o interrupción en la producción. Actualmente existen muchas restricciones ambientales y de salud por la falta de funguicidas para suelo disponibles para uso en invernaderos.

Este problema sumado al alto costo de esterilizar el ambiente con vapor está creando la necesidad de utilizar métodos alternativos de agricultura como la hidropónica en ambientes controlados. (Dalrymple DG, 1973) (Jensen, 2017). Cultivos hidropónicos Durante los últimos años, ha habido un interés creciente en la hidroponía o cultivos sin suelo para poder producir técnicas de cultivos hortícolas en invernaderos. El futuro de la hidroponía depende en gran medida del desarrollo de sistemas de producción que sean competitivos en costos con agricultura de campo.

Hay varios tipos de sistemas hidropónicos, así como algunos diseños de estructuras para invernadero y muchos métodos de control del medio ambiente pero no todos los sistemas son rentables en cada ubicación. Si bien las técnicas de cultivo hidropónico en los trópicos pueden ser bastante similares a los utilizados en las regiones templadas, las propias estructuras de los invernaderos y los métodos del control pueden diferir en gran medida.

La Hidroponía es una tecnología para cultivar plantas en soluciones de nutrientes (agua y fertilizantes) con el uso de suelo como medio artificial (por ejemplo, arena, grava, cascara de arroz, lana de roca) para proporcionar el soporte mecánico de las plantas. También existen los sistemas hidropónicos líquidos que tienen un medio sólido de soporte (bases plásticas) para las raíces de las plantas.

Los sistemas hidropónicos también se clasifican en abiertos y cerrados. En los sistemas abiertos una vez que se entrega la solución de nutrientes a las raíces de la planta, esta no se reutiliza y en los sistemas cerrados la solución nutritiva sobrante se recupera y se reabastece al sistema nuevamente (Jensen, 2017).

Algunos productores y agencias de publicaciones continúan limitando la definición de hidroponía a sistemas líquidos solamente y esta exclusión de la hidroponía la utilizan para desenfocar datos estadísticos que pueden llevar a una subestimación del alcance de la tecnología y sus implicaciones económicas. Prácticamente todos los sistemas hidropónicos del mundo están encerrados en estructuras de tipo invernadero para proporcionar control de temperatura, reducir la pérdida de agua por evaporación, reducir infestaciones de enfermedades, plagas y proteger los cultivos contra los elementos del clima, como el viento y lluvia.

(Dalrymple DG, 1973). Las principales ventajas del cultivo hidropónico incluyen el máximo rendimiento del cultivo con una alta densidad de producción, es decir en donde no exista un suelo adecuado se puede cultivar, además son virtualmente indiferentes a la temperatura ambiente, también realizan un uso más eficiente del agua y de los fertilizantes adicionalmente es altamente adecuado para la automatización y el control de las enfermedades.

Otra gran ventaja de la hidroponía, en comparación con el cultivo de las plantas en el suelo, es el aislamiento que tiene el cultivo, ya que a menudo el suelo tiene muchos problemas de enfermedades, salinidad, carencia de nutrientes y drenaje. Una de las tareas más costosas y que requieren muchísimo tiempo es la esterilización, la cual es innecesaria en los sistemas hidropónicos, asimismo otra ventaja importante si se desea realizar un cambio de cultivos se lo logra fácil y rápidamente.

Las principales desventajas de la hidroponía, en relación con la agricultura convencional, son los altos costos de los insumos, la energía requerida y el alto grado de gestión para lograr una producción exitosa. Los costos iniciales pueden ser excesivos si las estructuras son calentadas artificialmente y enfriadas por evaporación mediante sistemas de ventiladores o control ambiental, aunque no siempre son necesarios en los trópicos.

Debido a sus costos significativamente más altos, las aplicaciones exitosas de la tecnología hidropónica se limitan a cultivos de alto valor económico para regiones específicas, a menudo confinadas a épocas específicas del año. Por su naturaleza, los sistemas de cultivo hidropónicos son sistemas cerrados en donde las raíces de las plantas están expuestas directamente a la solución de nutrientes sin otro medio de cultivo y esta solución se reutiliza nuevamente.

Existen varios sistemas en esta categoría, como la técnica Nutrient Film Technique (NFT) y Deep Flow Hydroponics (Hidroponía De Flujo Profundo). Este último, se ha convertido posiblemente en el sistema hidropónico más popular para producción de vegetales de

hojas y hierbas. (Graves CJ, 1983) (Arnon, 1938) (Collins, 1985) (Winsor, 1979). Deep Flow Hydroponics (DFT).

Es un método para cultivar lechuga y otras verduras, fueron desarrolladas de forma independiente por Jensen (Jensen, Collins 1985) en Arizona y Massantini (1976) en Italia para la producción a gran escala y se han popularizado en muchos países del mundo. Por ejemplo, en el Caribe, la producción de lechuga ha sido posible combinando este sistema de hidroponía con el enfriamiento de la solución nutritiva.

La solución de nutrientes es monitoreado, reabastecido, recirculado y aireado. En la tabla 1 se observa una comparación entre las ventajas y desventajas. Tabla 1 Ventajas y desventajas Deep Flow Hydroponics (DFT). VENTAJAS _DESVENTAJAS _ _Ayudan a mantener la temperatura adecuada de la solución de nutrientes para que las plantas crezcan y se desarrollen.

La solución nutritiva circula alrededor de las raíces de la planta utilizando una bomba o gravedad. Es especialmente útil en la distribución de nutrientes esenciales como el calcio y el potasio. Las raíces de las plantas se sumergen constantemente en la solución de nutrientes. _Es probable que se necesite un enfriador para la solución de nutrientes.

Puede formar espuma cuando usa productos orgánicos. Existen fluctuaciones de pH por lo que se requiere que este sea siempre monitoreado. Las raíces pueden crecer tanto que pueden obstruir las líneas de irrigación. _ _Fuente: (Hawthorne Gardening, 2017) Nutrient Film Technique (NFT). Esta técnica se desarrolló a finales de 1960 por el Dr. Allan Cooper en el Glasshouse Crops Research Institute en Littlehampton en Inglaterra.

NFT ha dado lugar a varios sistemas modificados que se utilizan para la producción de vegetales principalmente lechuga. Para cultivos como los tomates, la solución de nutrientes se realiza a través de canales plásticos que contienen las raíces de las plantas. Las paredes de los canales son flexibles lo que permite que se junten alrededor de la base de cada planta, excluyendo la luz lo que genera que se prevenga la evaporación.

La solución de nutrientes se bombea al extremo superior de cada canal y fluye por gravedad pasado por las raíces en las tuberías de captación y un sumidero. La solución se controla para la reposición de sales y agua antes de que se recicle. Una ventaja principal del sistema NFT en comparación con otros es que requiere mucho menos solución de nutrientes.

En la tabla 2 se aprecia una comparación entre las ventajas y desventajas. Tabla 2.

Ventajas y Desventajas de Nutrient Film Technique (NFT). VENTAJAS _DESVENTAJAS _
_El costo de construcción de un sistema NFT es bajo. El movimiento constante del agua evita que el agua se estanque. Se pueden cultivar muchas plantas. Puede usar cualquier medio de cultivo para mantener la planta.

No se necesita un temporizador para que fluya la solución nutritiva porque el agua siempre fluye por gravedad. _Existen fluctuaciones de pH por lo que se requiere que este sea siempre monitoreado. Si la bomba se daña, las plantas morirán rápidamente ya que no hay medios para contener el agua.

Las plantas con raíces cortas necesitarán ser alimentadas por la parte superior o incorporar líneas de goteo, hasta que las raíces crezcan hasta el fondo del PVC. _
_Fuente: (Hawthorne Gardening, 2017) Aeroponics. En una aplicación inusual de hidroponía de sistema cerrado, en donde las plantas se cultivan en agujeros de poliestireno expandido u otro material.

Las raíces de las plantas están suspendidas en el aire y encerrado en una caja sellada de pulverización para que las raíces estén en oscuridad (para evitar el crecimiento de algas) y en la humedad de saturación. Un sistema de nebulización rocía la solución nutritiva sobre las raíces periódicamente. El sistema normalmente está encendido por solo unos pocos segundos cada 2-3 minutos.

Esto es suficiente para mantener las raíces húmedas y la solución de nutrientes gaseada. Los sistemas fueron desarrollados en Arizona para lechuga, espinacas e incluso tomates, aunque se consideró que el último no era económicamente viable. No se conocen operaciones aeropónicas comerciales a gran escala, aunque varias pequeñas empresas comercializan sistemas para uso doméstico.

En la tabla 3 se detalla una comparación entre las ventajas y desventajas.

Tabla 3 Ventajas y desventajas de Aeroponics. VENTAJAS _DESVENTAJAS _ _Alta relación entre el oxígeno y el agua. Existe una reutilización el agua. Existe un aumento de la absorción de la solución de nutrientes y un crecimiento acelerado. _Tiene las más altas fluctuaciones de pH.

Las raíces necesitan ser regadas con más frecuencia, por cortos períodos de tiempo, por lo tanto, se pueden secar muy rápido si no se les da suficiente solución nutritiva. El rociador se obstruye fácilmente, por lo que no es bueno para los cultivos orgánicos. _
_Fuente: (Hawthorne Gardening, 2017) Sistemas Hidropónicos Agregados. En los sistemas hidropónicos agregados un medio sólido e inerte proporciona soporte para las plantas.

Al igual que en los sistemas líquidos, la solución de nutrientes se administra directamente a las raíces de las plantas. Los sistemas agregados pueden ser abiertos o cerrados. Los sistemas abiertos no reciclan la solución de nutrientes y los sistemas cerrados si lo hacen. En la mayoría de los sistemas hidropónicos abiertos, se recupera el exceso de solución de nutrientes, pero el excedente no se recicla, sino que se desecha en estanques de evaporación o se utiliza para regar las plantaciones adyacentes y debido a que la solución de nutrientes no se recicla, dichos sistemas abiertos son menos sensibles a la composición del medio utilizado o a la salinidad del agua. En la tabla 4 se aprecia una comparación entre las ventajas y desventajas.

Tabla 4 Ventajas y desventajas de los Sistemas Hidropónicos Agregados VENTAJAS _DESVENTAJAS _ _El costo de implementación es bajo. Puede tener una alimentación de la solución de nutrientes constante o mediante un sistema temporizado. Se puede optar por las opciones de reutilizar o no la solución de nutrientes.

Está constantemente limpiando la solución de arriba hacia abajo, por lo que no tiene mucha acumulación de sal. _Puede obstruirse fácilmente, por lo que no es bueno para los cultivos orgánicos. Necesita mucho más tiempo para su configuración inicial. Dado que usualmente usas más medios que en comparación con otros métodos hidropónicos, la tasa de absorción es más lenta.

_ _Fuente: (Hawthorne Gardening, 2017)

Formulación Según la FAO (Organización de las Naciones Unidas para la Alimentación y la Agricultura) que es una Organización sin ánimo de lucro, se estima que para el año 2050 la población mundial será de 9100 millones de personas, un 34 % superior a la de hoy en día, y prácticamente la totalidad de este incremento de la población tendrá lugar en los países en desarrollo.

La urbanización continuará a un ritmo acelerado, aproximadamente el 70 % de la población mundial será urbana (en la actualidad esta cifra es del 49 %) y para poder alimentar a esta población tan numerosa, la producción de alimentos deberá aumentar aproximadamente en un 70%. Si se considera que el rendimiento de la producción se incrementa linealmente en el ámbito mundial como es la tendencia registrada durante los últimos 50 años, ese provee que para las próximas décadas la producción de alimentos no será suficiente para satisfacer las necesidades alimentarias del mundo.

(Organización de las Naciones Unidas para la alimentación y la agricultura, 2017). Por lo tanto, el desafío actual es que con la utilización de tecnología de punta en automatización y nuevas técnicas de cultivo alternativo en ambientes controlados como la hidroponía se pueda solucionar en parte la producción de alimentos.

Justificación:

En la actualidad existen muchos métodos de cultivo, los invernaderos tradicionales y los cultivos hidropónicos cada uno con sus pros y contras. El método hidropónico tiene muchas más ventajas que los cultivos tradicionales ya que es un sistema totalmente confiable, con un bajo nivel mantenimiento, que produce cultivos de mejor calidad y una tasa menor de enfermedades.

La continua investigación y el desarrollo de sistemas automáticos pueden llevar en un futuro a mejorar la relación costo-beneficio entre la energía y los materiales utilizados, logrando así que los nuevos cultivos se los realicen en ambientes controlados más apropiados para tener un mejor control. También existe un alto interés para impulsar el desarrollo tecnológico de automatización para mejorar la producción de los invernaderos en la agricultura, ya que utilizando la tecnología más avanzada se puede solucionar los problemas actuales de producción debido a que los alimentos en invernaderos son altamente productivos.

Sin embargo, parece que no hay una base para el uso de una tecnología mucho más amplia y rápida de lo que está ocurriendo actualmente para el crecimiento en ambientes controlados. La agricultura depende en gran medida del desarrollo de sistemas de producción que son competitivos en costos con los de la agricultura tradicional (Castilla, 1995) (Jensen, 2017).

Alcance: En el siguiente proyecto se plantearán las bases teóricas, el programa y el algoritmo para el funcionamiento de un prototipo de un control automático para un contenedor de carga para lograr una funcionalidad básica basada en crear un ambiente controlado en donde se puedan sembrar cultivos y verificar su crecimiento. Funciones avanzadas como: poder modificar y calibrar remotamente los parámetros de temperatura, humedad, iluminación, tiempo e intervalo de riego, no serán realizadas.

Tampoco se creará alguna aplicación de software para PC, Tablet y Smartphone. Al ser un prototipo se creará un contenedor a escala donde se implementará el circuito de control para verificar su funcionamiento y este a su vez podrá ser utilizado como base para en un futuro desarrollar un circuito comercial mucho más elaborado, con una interfaz de control remoto para uso mediante el internet y también el manejo de otros parámetros como el control de pH, temperatura de la solución nutritiva, los niveles de los estanques, los cuales fueron incluidos en el diseño.

OBJETIVOS Objetivo general: Desarrollar un sistema de automatización de un contenedor de carga para el control de clima e iluminación de cultivos hidropónicos. Objetivos específicos: Definir los parámetros de control de temperatura, humedad, iluminación nutrición y riego. Diseñar el sistema de automatización de un contenedor de carga para el control de clima y crecimiento de cultivos hidropónicos según los parámetros establecidos en el objetivo general. Desarrollar el algoritmo de control y automatización para el control automático.

Construir el sistema automatizado y circuito electrónico para controlar temperatura, humedad, nutrición, riego e iluminación para cultivos hidropónicos. Realizar pruebas de funcionamiento y validación de resultados para cultivos hidropónicos.

Descripción del contenido de los capítulos: Capítulo 1: en este capítulo se presenta el marco teórico relacionado con el dispositivo desde un punto de vista de elaboración, se hará una breve explicación de los componentes y principios sobre los cuales funciona el prototipo.

Capítulo 2: en este capítulo se evalúan las opciones de diseño e implementación del programa y se analiza la máquina de estados finitos. Capítulo 3: en este capítulo se documenta el proceso de elaboración y las opciones para la construcción del circuito. Capítulo 4: en este capítulo se resumen las pruebas realizadas de operación y se documentan las experiencias de implementación.

Conclusiones y recomendaciones: se presentan los resultados de las pruebas realizadas durante la implementación del prototipo. Se presentarán algunas recomendaciones de mejora para trabajos futuros.

CAPÍTULO 1 FUNDAMENTOS TEÓRICOS En este Capítulo se detalla el diagrama de bloques, los fundamentos teóricos del funcionamiento del microprocesador, los sensores para adquirir los datos de humedad/temperatura, también los elementos externos como RTC, el módulo I2C, las luces Led para horticultura y circuitos para controlar la etapa de potencia.

El concepto de garantizar el suministro de alimentos en el futuro no consiste solo en sembrar en un buen lugar los cultivos, lo que realmente necesitaremos será una granja de última tecnología, con una gran cantidad de técnica de cultivo y una eficiente automatización que nos permita proporcionar las condiciones óptimas para que las plantas se desarrollen rápido y sanamente.

Esto lo lograremos controlando los parámetros más importantes para el crecimiento como la humedad relativa, temperatura, el tipo de Iluminación, tiempos de riego e intervalos de alimentación. Marco teórico Hay varios tipos de microcontroladores e interfaces accesibles en el mercado, pero debido al conocimiento recibido en las materias de Microprocesadores y Control Automático para desarrollar este proyecto integrador de carrera se eligió como controlador al módulo Arduino UNO Wifi que es una interfaz de electrónica de código abierto fundamentada en hardware y con una programación simple de utilizar.

La arquitectura del sistema se muestra en el diagrama de bloques de la figura 1.1, donde se puede observar que consta con dos sensores de entrada, uno de humedad y otro de temperatura, también se utiliza el módulo de Arduino UNO Wifi como controlador para procesar las entradas y salidas a través de conversores análogos/digitales, el cual incorpora un módulo Wifi ESP8266 que genera una red Wifi local (servidor) y también sirve para conectarse al Internet a través de una dirección IP publica privada programada en el Arduino Uno Wifi.

Mediante la interface del módulo Wifi se puede acceder de forma remota mediante una computadora, tablets o smartphones (Android o iOS) visualizar como varían los valores de los parámetros de los sensores tanto de temperatura como de humedad, verificar que la iluminación luz LED de crecimiento está encendida o apagada, ver el tiempo programado de riego y también los intervalos programados para que el riego se lo realice periódicamente.

Una de las salidas de módulo Arduino UNO Wifi controlan una pantalla LCD (liquid-crystal display) monocromática y alfanumérica de 20 caracteres por 4 filas (20x4) que es controlada por un módulo Arduino I2C Inter-Integrated Circuit. Las otras salidas del módulo Arduino controlan la etapa de potencia las cuales nos sirven para manejar

los dispositivos electrónicos de alto voltaje mediante relés que funcionan como un interruptor para activar las luces LED de crecimiento, el calefactor, aire acondicionado el humidificador y el deshumificador.

En la figura 1.1 se detalla en general el diagrama de bloques que se utiliza para el funcionamiento del sistema de automatización. / Figura 1. 1. Diagrama de bloques Fuente: Elaborado por el autor A continuación, se describirá el componente teórico de cada bloque, así como cada uno de sus componentes. Las hojas técnicas (datasheets) de cada componente se lo presentan en la sección de Anexos.

Microcontrolador Arduino Uno Wi-Fi El Arduino Uno Wi-Fi es similar al Arduino Uno Rev.3 pero con un módulo de comunicación inalámbrica Wi-Fi integrado. La placa se basa en ATmega328P con un módulo de Wi-Fi ESP8266 integrado. Tiene 14 pines digitales de IN/OUT (de los cuales 6 pueden usarse como salidas PWM), 6 entradas analógicas, un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP y un botón de reinicio (Reset), para su alimentación solo hace falta conectarlo a una computadora con un cable USB, un adaptador de CC de 12VCD o a una batería. El módulo Arduino UNO Wifi se detalla en la figura 1. 2.

El Módulo Wi-Fi ESP8266 es un SoC (System on a Chip) autónomo con pila de protocolos TCP / IP integrados que puede brindar acceso a una red Wi-Fi con la ventaja de poder actuar como un punto de acceso. / Figura 2 Arduino Uno Wi-Fi (Arduino, 2016) 1.3. Dispositivos de entrada Sensor de humedad relativa y temperatura DTH22 El sensor de humedad relativa y temperatura DHT22, usa la tecnología de recolección de datos digitales exclusivamente, asegurando su confiabilidad y seguridad.

Todos sus elementos están conectados con un solo chip de 8 bits, cada sensor internamente está compensado por temperatura, para lo cual el fabricante utiliza una cámara de calibración y el resultado de este proceso que es un coeficiente de temperatura se almacena en el programa de la memoria interna OTP. En la figura 1. 3. Se detalla el sensor de humedad y temperatura DTH22. / Figura 3 Sensor de humedad relativa y temperatura DTH22. Fuente: (Rduinostar.com, 2015) Por su tamaño reducido, bajo consumo de energía y transmisión de datos de larga distancia aproximadamente de 20 metros, hace que el módulo DHT22 sea adecuado para el diseño e implementación del proyecto ya que el módulo esta empaquetado en una placa de tres pines, lo que hace que la conexión sea más simple y versátil.

Cuando se suministra energía al sensor, este no envía ninguna información dentro del primer segundo hasta poder pasar del estado inestable al estable. Se recomienda colocar un capacitor de 100nF entre VDD y GND para realizar un filtro en la

alimentación del sensor. En la tabla 1.1 se describe las características eléctricas del Sensor DTH22.

Tabla 4 Características eléctricas del DTH22 DESCRIPCIÓN _CONDICIÓN _MÍNIMO _TÍPICO _MÁXIMO _UNIDADES _ Fuente Alimentación _DC _3.3 _5 _6 _[V] _Consumo Energía _Medición _1 _N/A _1.5 _[mA] _Stand-by _40 _N/A _50 _[uA] _Tiempo Medida _Segundos _N/A _2 _N/A _[s] _Fuente: (Adafruit.com, 2018, pág.

5) El sensor DTH22 de acuerdo con las especificaciones del fabricante tiene las siguientes características para medir la temperatura: Resolución Decimal: que son los valores de humedad y temperatura representados con una cifra decimal. Tiempo de muestreo: que es de 2 segundos, sólo puede entregar datos correctos cada 2 segundos. Rango de valores: que va desde los -40°C hasta los 80°C de temperatura. Precisión: que es de $\pm 0.5^{\circ}\text{C}$, y hasta $\pm 1^{\circ}\text{C}$ cuando se encuentra en condiciones adversas.

Tiempo de respuesta: es menor a 10 segundos, es decir que tarda menos de 10 segundos en mostrar un cambio de temperatura en el entorno. Adicionalmente el sensor DTH22 para medir la humedad relativa tiene las siguientes características: Rango de valores: que va desde el 0% hasta el 99.9% de Humedad Relativa. Precisión: que es de $\pm 2\%$ de Humedad Relativa, a una temperatura de 25°C .

Tiempo de respuesta: que es menor a 5 segundos, es decir que tarda menos de 5 segundos en mostrar un cambio de humedad relativa en el entorno y adicionalmente la velocidad del aire debe ser de 1 [m/s]. (Adafruit.com, 2018, pág. 1). Para la comunicación se utiliza un bus simple de datos entre el Módulo Arduino y el sensor DHT22.

El tiempo requerido para la comunicación es de 5 ms y los datos tienen una longitud de 40 bits los cuales se dividen en dos partes, una para la parte entera y la otra para la parte decimal. Para el proceso de transmisión de los datos se envía primero un bit de datos en alto y luego la trama de datos de 40 bits la cual se divide de la siguiente manera: Datos de humedad relativa de los valores enteros en formato de 8 bits. Datos de humedad relativa de los valores decimales en formato de 8 bits.

Datos de temperatura de los valores enteros en formato de 8 bits. Datos de temperatura de los valores decimales en formato de 8 bits. Checksum (suma de chequeo) de verificación de 8 bits. Si la transmisión de la trama de datos es correcta, la suma de verificación (Checksum) debe ser los últimos 8 bits de datos enteros humedad relativa + 8 bits de humedad relativa decimales + 8 bits de temperatura datos enteros + 8 bits de datos de temperatura decimales.

Cuando el Módulo Arduino envía la señal de inicio, el DHT22 pasa del modo de bajo consumo de energía al modo de desempeño y enviará la señal de respuesta de datos de 40 bits con la información de la temperatura y humedad relativa. Mediante el siguiente ejemplo se explicará como con los 40 bits de datos que envía el sensor DHT22, el módulo Arduino interpreta los valores, esto se muestra en la tabla 1.2.

Tabla 5 Bits de datos del DHT22 0000 0010 1000 1100 _0000 0001 0101 1111 1110
_1110 _16 Bits Humedad Relativa (HR) _16 Bits Temperatura (T) _Checksum _ Fuente:
(Adafruit.com, 2018, pág. 3) Primero se convierte los datos de Humedad Relativa (HR) de 16 bits del sistema binario al sistema decimal, 0000 0010 1000 1100 ? 652 HR = 652/10 = 65.2% Luego se convierte los datos de Temperatura (T) de 16 bits del sistema binario al sistema decimal, 0001 0101 1111 ? 351 T = 351/10 = 35.1

? Y al final se realiza la suma total para verificar el Checksum (Suma de Chequeo) Suma = 0000 0010 + 1000 1100 + 0000 0001 + 0101 1111 = 1110 1110 El checksum es igual a los últimos 8 bits de Suma = 1110 1110 Nota: Cuando el bit más alto de temperatura es 1, significa que la temperatura está por debajo de 0 grados Celsius. Ejemplo: 1000 0000 0110 0101, T = -10.1 ?. En la figura 1.4.

se observa cómo se conecta el módulo DHT22 al Arduino UNO y como la trama de datos a través del pin digital es enviada. / Figura 6 Esquema de conexión DHT22 Fuente: (Adafruit.com, 2018, pág. 2) 1.4. Dispositivos Especiales 1.4.1. Reloj En tiempo real RTC El RTC (Real Time Clock). de Arduino es un módulo que sirve para realizar un seguimiento de la hora actual y que se puede utilizar para programar acciones en un momento determinado.

La mayoría de los RTC usan un oscilador de cristal cuya frecuencia es 32.768 [kHz] que es la misma frecuencia utilizada en relojes de cuarzo. Como esta frecuencia es igual a 2 15 ciclos por segundo, esta velocidad es muy conveniente para utilizarlo con circuitos de contador binarios. Además, el RTC puede continuar funcionando en modo de suspensión o Sleep, por lo que se puede utilizarlo para activar el dispositivo desde cualquier parte del programada.

Cada vez que él modulo se enciende, el RTC se restablece y comienza a partir de una fecha estándar preprogramada. Para mantener el tiempo y el RTC en funcionamiento, es necesario mantener el módulo alimentado conectado a través de un diodo al pin de 3.3V. Para evitar que se pierda la hora el módulo RTC contiene una batería de litio el rango de 3V que sirve como respaldo. En la figura 1.5.

se muestra el módulo RTC de Arduino con los pines utilizados para la alimentación y datos. / Figura 7 Módulo RTC de Arduino Fuente: (Maxim Integrated, 2014) 1.4.2. Modulo I2C El bus de comunicación I2C es muy popular y ampliamente utilizado por muchos dispositivos electrónicos, ya que permite implementar fácilmente muchos diseños electrónicos que requieren comunicación entre un maestro y múltiples dispositivos esclavos o incluso múltiples dispositivos maestros.

Cuando se utiliza un direccionamiento de 7 bits, se puede conectar hasta casi 128 (112) dispositivos y cuando se utiliza el direccionamiento de 10 bits hasta 1024 (1008) dispositivos según los datos del fabricante. En algunas situaciones, puede ser útil configurar dos o más módulos Arduino para compartir información entre ellos.

Por ejemplo, el módulo de Display LCD y el RTC están programadas para comunicarse entre sí mediante una configuración de Receptor principal / Receptor secundario a través del protocolo en serie síncrono I2C. En la figura 1.6. se observa el módulo I2C de Arduino con los pines utilizados para la alimentación y datos. / Figura 8 Módulo I2C de Arduino Fuente: (Opencircuit, 2018, pág.

1) El protocolo I2C implica el uso de dos buses para enviar y recibir datos: un pin de reloj serial (SCL) que el módulo Arduino genera en intervalos regulares y un pin de datos serie (SDA) sobre el cual se envían los datos entre los dispositivos. A medida que la línea del reloj cambia de baja a alta (conocida como el flanco ascendente del pulso del reloj), un solo bit de información - que formará en secuencia la dirección de un dispositivo específico y un comando o datos - se transfiere del módulo a el dispositivo I2C sobre la línea SDA.

Cuando se envía esta información, poco a poco el dispositivo llamado ejecuta la solicitud y transmite sus datos utilizando la señal de reloj aún generada por el maestro en SCL como temporización. Los ocho bits iniciales (es decir, ocho pulsos de reloj) del maestro a los esclavos contienen la dirección del dispositivo maestro que quiere los datos.

Los siguientes bits contienen la dirección de memoria en el esclavo de la que el maestro desea leer o escribir datos. Los módulos con sus respectivos pines se detallan en la tabla 1.3. Tabla 9 Módulos y pines I2C Modulo Arduino _I2C / TWI Pines _ _Uno, Ethernet _A4 (SDA), A5 (SCL) _ _Mega2560 _20 (SDA), 21 (SCL) _ _Leonardo _2 (SDA), 3 (SCL) _ _Due _20 (SDA), 21 (SCL), SDA1, SCL1 _ Fuente: (Naylamp Mechatronics SAC, 2016) 1.5.

Dispositivos de salida Led's para Horticultura En la agricultura tradicional durante el día las hojas reciben directamente la luz del sol. Las longitudes de onda del espectro de luz

visible desempeñan papeles específicos para el crecimiento de la planta ya que producen pigmentos fotosintéticos dentro de las plantas.

Utilizando el avance de la tecnología para aumentar la tasa de crecimiento de los cultivos se ha creado un tipo de luces LED diseñadas para el crecimiento de las plantas las cuales emiten el espectro de luz de una manera muy específica para lograr resultados superiores en el crecimiento de los cultivos, de manera que mejora tanto el metabolismo primario como el secundario de las plantas el cual influye directamente con el crecimiento y la calidad de la planta.

Las luces LED para crecimiento están diseñadas para la gran mayoría de aplicaciones de cultivo (interior o exterior) donde se necesita iluminación principal y suplementaria. Estas luces LED emiten luz en la región fotosintética del espectro de luz visible y pueden soportar la mayoría de los entornos existentes en los invernaderos durante toda su vida útil, además el diseño optimizado de las luces LED para crecimiento las hace fáciles de instalar y su avanzado diseño es el catalizador para lograr una óptima fotosíntesis para mejorar la productividad de la planta. En la Tabla 1.4.

Se realiza una comparación entre el espectro de luz visible y el porcentaje del contenido espectral de acuerdo con la longitud de onda. Tabla 10 Contenido espectral de los LED para crecimiento

Longitud de Onda	Contenido Espectral (% DEL TOTAL)
400-500 nm	33%
500-600 nm	2-14 %
6000-700 nm	41% 1-5 %
81-97%	26%
100 %	81-97%

Fuente: (OSRAM Opto Semiconductors, 2018, pág.

4) En la horticultura utilizan 3 tipos LED's con diferentes rangos de longitud de onda para irradiar a los cultivos: Deep Blue (azul intenso) con una longitud de onda de 450 [nm]. Hyper Red (rojo hiper) con una longitud de onda de 660 [nm] Far Red (rojo lejano). con una longitud de onda de 730 [nm]. Estas longitudes de onda están consideradas como las más recomendables para tratar de cubrir el mayor espectro de luz visible según OSRAM Opto Semiconductors.

Su dispersión es en tres ángulos de radiación 80°, 120° y 150° lo que ayuda a proporcionar la iluminación perfecta para todo tipo de plantas y permite adaptar la luz exactamente a las necesidades de diversos cultivos. La distribución espectral de los LED's para crecimiento en la actualidad ha mejorado muchísimo y los ha llevado a un nivel tan alto, que actualmente de la absorción de luz de las plantas se maneja a niveles tan pequeños como de los fotones.

Además, los LED's de crecimiento ofrecen una infinidad de posibilidades especialmente

para las técnicas de cultivo modernas, como la iluminación de superficies, agricultura vertical de varias capas o fuentes de iluminación individuales. En la figura 1.7. se observa la distribución espectral de luz de los LED para crecimiento. / Figura 11 Distribución espectral de luz de los LED para crecimiento Fuente: (OSRAM Opto Semiconductors, 2018, pág.

12) Este tipo de iluminación cubre la gama completa de requisitos específicos para la mayoría de los tipos de plantas, como su diseño es muy compacto con un consumo de energía es bajo, permite que su utilización sea fácil, simple y eficiente. Los LED's de crecimiento utilizados para este diseño para mejorar el rendimiento y la productividad del cultivo tienen las características que se observan en la tabla 1.5.:

Tabla 12 Especificaciones de los LED's de crecimiento ESPECIFICACIONES
_Alimentación _120 VAC / 50-60 Hz _Consumo de Energía _110 W _Salida de luz _2.2 - 2,4 umol/s per-Watt _Cobertura de luz _Hasta 12(?? ??) _Regulador de intensidad _10 – 100 % _Depreciación de la cantidad de luz _L85 B10 50.000 horas _Temperatura de operación _0-104 °F (0-40°C) _Grado de Protección _IP54 _Fuente: (OSRAM Opto Semiconductors, 2018) En la figura 1.8

se puede observar la tonalidad violeta generada por la utilización de 3 tipos de LED con diferentes rangos de longitud de onda para irradiar a los cultivos. / Figura 13 Plantas con Iluminación LED para crecimiento Fuente: Elaborado por el autor Pantalla LCD (Liquid Crystal Display) Una pantalla de cristal líquido (LCD) es una pantalla plana que utiliza las propiedades de modulación de la luz de los cristales líquidos.

Los cristales líquidos no emiten luz directamente por lo que necesitan una luz de retroalimentación para iluminarse. En la figura 1.9. se observa una pantalla LCD de 20x4. / Figura 14 Pantalla LCD Fuente: Elaborado por el autor Una pantalla LCD monocromática y alfanumérica de 20x4 significa que se pueden mostrar 20 caracteres en cada una de las 4 filas de la pantalla, por lo que se pueden mostrar un total de 80 caracteres en cualquier instancia de tiempo adicionalmente tiene una interfaz de 16 pines, los cuales se detallan en la tabla 1.6.

Tabla 15 Pines del LCD 20x4 Pin _Símbolo _Función
_1 _Vss _Tierra (0 V)
_2 _Vdd _5 V +
_3 _Vo _Ajuste de Contraste
_4 _RS _H/L código de instrucción
_5 _R/W _H/L señal lectura/escritura
_6 _E _H/L señal de habilitación
_7-14 _DB0 – DB7 _H/L bus de datos para modo de 4- o 8-bits
_15 _A (LED+) _Ánodo luz posterior
_16 _K (LED-) _Cátodo luz posterior
_Fuente: (Vishay, 2016, pág.

1) Etapa de potencia Para la etapa de potencia se utilizó varios módulos de relé de

Arduino, los cuales son dispositivos mecánicos que funcionan a partir de un pequeño voltaje de 5VCD aplicado a su bobina y con sus contactos puede manejar grandes cargas de potencia y también le permite aislar la etapa de potencia con la etapa de control.

El relé de 5VCD tiene 2 tipos diferentes de contactos eléctricos en el interior: normalmente abierto (NO) y normalmente cerrado (NC), su uso dependerá de si desea que la señal de +5V encienda el interruptor o apague el interruptor. El voltaje de funcionamiento es de 120-240 VAC, adicionalmente tiene el terminal común C en ambas configuraciones las cuales se conectan al dispositivo que desea controlar. El relé de 5VCD también tiene tres pines de bajo voltaje (tierra, Vcc y señal) que se conectan directamente al módulo de Arduino. En la figura 1.10.

se muestra un Módulo de Relé +5V. / Figura 16 Módulo de relé +5V Fuente: Elaborado por el autor Entre sus principales configuraciones están las siguientes: Normalmente abierto (NO) En esta configuración, cuando el relé recibe una señal ALTA (+5V), el interruptor se cierra y permite que la corriente fluya desde el terminal C al terminal NO.

Una señal BAJA desactiva el relé y detiene el paso la corriente, lo cual se ilustra en la figura 1.11. / Figura 17 Relé normalmente abierto (NO). Fuente: Elaborado por el autor Normalmente cerrado (NC) En esta configuración cuando el relé recibe una señal ALTA (+5V), el interruptor se abre e interrumpe la corriente. Una señal BAJA cierra el interruptor y permite que la corriente fluya desde el terminal C al terminal NC. Los detalles se observan en la figura 1.12.

/ Figura 18 Relé normalmente cerrado (NC). Fuente: Elaborado por el autor

CAPÍTULO 2 DISEÑO DEL PROGRAMA En este Capítulo se detalla el diagrama de flujo, el funcionamiento de los sistemas retroalimentados, las máquinas de estados finitos, el desarrollo y la elaboración del programa de automatización. Diagrama de Flujo.

Para el proyecto se diseñó un sistema de control automático para el manejo del cultivo hidropónico dentro de un contenedor y para poder iniciar con el programa lo primero a realizar es elaborar el diagrama de flujo del programa o algoritmo de programación, ya que de esa manera será mucho más fácil escribir el código de programa en el compilador de Arduino.

La ventaja de los diagramas de flujo es que permiten explicar de una manera general el programa a otros y como se trata de un programa complejo el diagrama de flujo permitirá visualizar gráficamente la secuencia lógica del programa. En la figura 2.13. se puede observar el diagrama de flujo para el sistema de control automático con los parámetros de temperatura, humedad, iluminación, tiempo e intervalo de riego.

/ Figura 19 Diagrama de flujo Fuente: Elaborado por el autor Luego de realizar el diagrama de flujo se procedió a la elaboración del código de programa creando una máquina de estados finitos que es un sistema de lazo cerrado, como se observa en la figura 2.14. y será el sistema de control principal para monitorear y controlar los parámetros de temperatura, humedad, iluminación, tiempo e intervalo de riego todo esto en una pantalla LCD o mediante un computador o Tablet mediante un explorador web, ingresando la dirección IP publica privada programada en el módulo Arduino Uno Wifi, lo que permite al usuario, la capacidad de controlar de forma automática y manual todo el sistema.

/ Figura 20 Sistema de control retroalimentado Fuente: Elaborado por el autor 2.2. Máquina de estados finitos. Las máquinas de estados son sistemas de entradas y salidas, en donde las salidas dependen de las señales de entradas actuales y también de las anteriores, con lo cual se puede resumir que una máquina de estados es un autómata o un conjunto de estados que relacionan las entradas y salidas para cada instante, para que la salida dependa únicamente del estado actual de las entradas.

Para la programación del sistema se definió los botones que vamos a necesitar BTN MENU, BTN EXIT, BTN UP y BTN DOWN y los pines de la placa Arduino Wifi a utilizar. Como ilustra la figura 2.15. / Figura 21 Definición de botones y pines Fuente: Elaborado por el autor / Figura 22 Botones para el control Automático. Fuente: Elaborado por el autor Luego se definió los estados a utilizarse como se detallan en la figura 2.17.

los cuales van a controlar el orden en el cual se van a desplazar dentro de la máquina de estados finitos y el nombre de la variable donde se va a guardar el estado actual. / Figura 23 Definición de los estados Fuente: Elaborado por el autor Cabe mencionar que se creó un arreglo en el programa para que cada vez que se pulse un botón el sistema detecte un flanco de subida en el tiempo, para lo cual se hizo que la entrada digital sea un PULL UP, y con eso, cuando se apriete cualquier botón el voltaje caerá hasta 0V y al soltarlo volverá a estado anterior de 5V.

Así, el programa solo detectará cuando exista un cambio en el estado anterior evitando así crear filtro contra ruido o un sistema anti rebotes, estos valores se los guarda en una variable llamada flancoSubida. Esto se explica en la figura 2.18. / Figura 24 Flanco de bajada y subida. Fuente: Elaborado por el autor A continuación, en la figura 2.19 se muestra cómo se escribe el código en el software de Arduino para poder controlar el flanco de subida. / Figura 25 Rutina para controlar el flanco de subida.

Fuente: Elaborado por el autor Para este circuito se utilizó un autómata que solo puede estar en un solo estado, es decir solo se podrá ejecutar en un solo menú a la vez y que

se lo representa con un círculo. A este estado se lo llamó HOME., Cuando estemos en este estado se mostrará el mensaje programado para el MENU y no realizará ninguna tarea, simplemente va a mostrar ese mensaje.

En el mensaje MENU se va a desplegar la información del Título del proyecto, la temperatura actual medida en grados centígrados, la humedad relativa expresada en tanto por ciento, la hora y la fecha actual, como se muestra en la figura 2. 20. / Figura 26 Pantalla principal de MENU Fuente: Elaborado por el autor Otro estado que se utiliza es SET TEMP el cual se detalla en la figura 2.21.

en el cual se puede cambiar el valor de la temperatura. El cambio desde el MENU hacia el estado SET TEMP se lo realiza mediante una transición al momento en que se presiona el botón MENU. / Figura 27 Estado inicial de HOME y SET TEMP.

Fuente: Elaborado por el autor En el estado SET TEMP se utilizará los botones BTN UP para subir la temperatura y BTN DOWN para bajar la temperatura, el cual se aprecia en la figura 2.22. La temperatura se podrá incrementar y decrementar en un rango programado desde los 10 °C hasta 40 °C (grados centígrados) en intervalos de 1°C activando automáticamente un Calefactor para aumentar la Temperatura y un Aire Acondicionado para disminuir la temperatura.

Cuando se desee pasar al siguiente estado se deberá pulsar el botón BTN MENU y cuando se desee salir e ir a la pantalla del estado inicial debe presionar el botón BTN EXIT. / Figura 28 Estado inicial SET TEMP. Fuente: Elaborado por el autor En el menú TEMPERATURA se despliega la opción para setear la temperatura de operación y se mostrará cómo se detalla en la figura 2.23.

/ Figura 29 MENU de la temperatura de operación Fuente: Elaborado por el autor Se agregó otro estado con el cual se puede controlar la humedad relativa al cual, el estado se llama SET HUM, el cual se observa en la figura 2.24. donde se utiliza los botones BTN UP para subir la Humedad Relativa y BTN DOWN para bajar la Humedad relativa.

Con este control se puede incrementar y decrementar la humedad en un rango programado desde el 40% hasta 100% en intervalos de 10%, activando automáticamente un Humidificador para aumentar la Humedad y un Deshumidificador para disminuir la humedad. Cuando se desee pasar al siguiente estado se deberá pulsar el botón BTN MENU y para salir e ir a la pantalla del estado inicial se deberá presionar el botón BTN EXIT.

/ Figura 30 Estado inicial SET HUM Fuente: Elaborado por el autor En el menú

HUMEDAD se va a desplegar la opción para escoger la humedad relativa de operación y se observa en la figura 2.25. / Figura 31 MENU de la humedad de operación. Fuente: Elaborado por el autor Después de controlar la humedad relativa se agregó otro estado para controlar la iluminación dentro del contenedor, al cual lo se lo llamó SET ILUM como se muestra en la figura 2.26. Para el encendido y apagado de los LED se utiliza los botones BTN UP y BTN DOWN respectivamente.

Con este estado solo se tiene un control automático de encendido y apagado ON/OFF de las luces led diseñadas para el cultivo hidropónico. Cuando se desee pasar al siguiente estado se deberá pulsar el botón BTN MENU y para salir e ir a la pantalla del estado inicial se presionará el botón BTN EXIT. / Figura 32 Estado inicial SET ILIM Fuente: Elaborado por el autor En el menú ILUMINACION se va a desplegar la opción para encender o apagar la iluminación dentro del contenedor y se mostrara como se muestra en la figura 2.27. / Figura 33 MENU del control de la iluminación.

Fuente: Elaborado por el autor Para controlar el tiempo de riego se agregó otro estado con el nombre SET TIME que se muestra en la figura 2.28. Se utilizan los botones BTN UP para subir el tiempo de regadío y BTN DOWN para disminuir el tiempo de regadío. Con este control automático se puede incrementar y decrementar el tiempo de riego en un rango programado desde 5 minutos hasta 60 minutos en intervalos de ± 5 minutos, activando automáticamente la bomba sumergible para transportar la solución alimenticia a las plantas.

Cuando se desee pasar al siguiente estado se debe pulsar el botón BTN MENU y para salir e ir a la pantalla del estado inicial se deberá presionar el botón BTN EXIT. / Figura 34 Estado inicial SET TIME Fuente: Elaborado por el autor En el menú TIEMPO DE RIEGO se va a desplegar la opción para seleccionar el tiempo que necesario para el riego de los cultivos dentro del contenedor y se detalla en la figura 2.29. / Figura 35 MENU del control del tiempo de riego.

Fuente: Elaborado por el autor Finalmente, se agregó otro estado para controlar el intervalo de riego, el estado tiene como nombre SET INTER, se utiliza los botones BTN UP para subir el tiempo de regadío y BTN DOWN para disminuir el tiempo de regadío. Con este control automático se puede incrementar y decrementar el tiempo de riego en un rango programado desde 1 hora hasta 24 horas en intervalos de ± 1 hora, activando automáticamente la bomba sumergible para transporta la solución alimenticia a las plantas. En la figura 2.30 se muestra la maquina de estados completa con todos los estados. / Figura 36 Máquina de estados completa.

Fuente: Elaborado por el autor En el menú HORA DE RIEGO se va a desplegar la opción

para seleccionar el intervalo de tiempo que se necesite para el riego de los cultivos dentro del contenedor y se mostrara como se detalla en la figura 2.31. / Figura 37 MENU del control del intervalo de tiempo para riego. Fuente: Elaborado por el autor Como ya no existen más estados cuando se pulse el botón BTN MENU el programa direccionará nuevamente al estado SET TEMP y así seguirá constantemente en un lazo cerrado completado la máquina de estados finitos.

Cuando se desee salir e ir a la pantalla del estado inicial se deberá pulsar el botón BTN EXIT.

Capítulo 3 Diseño del Hardware En este Capítulo se detalla el diseño del hardware, el desarrollo, la simulación, la construcción de la placa PCB y el ensamblaje del circuito.

Una vez terminada la fase de diseño del software se procedió a la implementación del circuito y lo primero que fue realizado fue el diseño del diagrama esquemático a través del software de Proteus. Esta herramienta informática se utilizó para realizar la simulación de un control de temperatura y humedad, el diagrama esquemático y la PCB.

La simulación del control de temperatura y humedad se lo presenta en la figura 3.32. de la cual se puede observar la simulación de un sensor DTH22 conectado a un módulo Arduino Uno mostrando los valores de humedad relativa y temperatura. / Figura 38 Simulación en Proteus.

Fuente: Elaborado por el autor Luego de realizar varias simulaciones y verificar que el circuito funciona correctamente se procedió a armar en un protoboard el circuito para el control automático con todos los elementos necesarios para empezar a realizar las pruebas y verificar el funcionamiento. En la figura 3.32. se observa el circuito de prueba en protoboard. / Figura 39 Circuito de prueba en protoboard. Fuente: Elaborado por el autor 3.1.

Desarrollo Una vez que se verificó que el sistema funciona en el protoboard, se procedió a crear la placa PCB siguiendo las reglas establecidas para la elaboración de circuitos impresos entre las cuales se establece que las pistas deben ser lo más cortas y directas posibles para evitar que la impedancia y otros efectos parásitos pueden limitar el rendimiento del sistema.

Para el ancho de las pistas se utilizó la norma ANSI, que define los requisitos generales para el diseño de PCB y algunos aspectos de ensamble de tarjetas electrónicas, principios y recomendaciones, con lo cual se realizó un cálculo para que estas puedan soportar una corriente de hasta 1 Amperio sin generar daño de las pistas y también que no se genere un incremento de temperatura más allá de 10°C y de acuerdo con los cálculos realizados se optó por un grosor de pista de 0.608 mm.

En la figura 3.34 se detalla el cálculo del ancho de una Pista PCB con respecto a la corriente. ANSI PCB Cálculo del ancho de una Pista PCB con respecto a la corriente "AMPERES" / Figura 40 Cálculo del ancho de banda de una pista PCB Fuente: Elaborado por el autor Una vez que se realizaron todos los cálculos de diseño se procedió a la elaboración de la placa PCB en el software Proteus, la cual se muestra en la figura 3.35.

/ Figura 41 Diseño de la PCB Fuente: Elaborado por el autor Una vez que se diseñó la

placa PCB en el software Proteus, se realizó la impresión del diseño en un material termo-sensible para transferir a través de calor a la placa de cobre donde quemó con un ácido (percloruro férrico) para eliminar el cobre excedente del circuito impreso en la baquelita y las perforaciones fueron realizadas con brocas pequeñas hechas de material de carburo tungsteno.

Para este circuito impreso se utilizaron los estándares IPC-2220 que son aquellos que regulan el diseño, ensamblado y control de calidad de los circuitos impresos PCB, como se puede apreciar en la figura 3.36. / Figura 42 PCB (Placa de circuito impreso). Fuente: Elaborado por el autor Para el montaje de los componentes en la PCB, se fijaron los elementos y luego se soldaron a los pines con estaño, como se aprecia en las figuras 3.37. y 3.38. / Figura 43 Soldadura de elementos en la PCB (Placa de circuito impreso).

Fuente: Elaborado por el autor / Figura 44 Montaje de elementos en la PCB (Placa de circuito impreso). Fuente: Elaborado por el autor Una vez que concluimos con el ensamblaje de los componentes internos, procedemos a montar los circuitos en una caja acrílica transparente, la cual servirá para la presentación final del proyecto como se observa en la figura 3.39. / Figura 45 Montaje de los pulsadores en la placa acrílica.

Fuente: Elaborado por el autor Luego que se concluyó con la parte electrónica del circuito, se procedió al diseño, armado, construcción y montaje del contenedor de carga a escala, donde se colocaron los elementos de control, calefactor, aire acondicionado, humidificador, deshumificador como se muestran en las figuras 3.40. y 3.41. / Figura 46 Armado del contenedor a escala. Fuente: Elaborado por el autor / Figura 47 Proyecto finalizado.

Fuente: Elaborado por el autor

CAPÍTULO 4 PRUEBAS Y ANÁLISIS DE RESULTADOS En este Capítulo se detallan las pruebas realizadas y el análisis de los resultados del circuito de control automático para cultivos hidropónicos. 4.1 Pruebas La primera parte de las pruebas, se la realizó en el laboratorio cuando todo el circuito de control estaba ensamblado y consistió en confirmar que el módulo de automatización encienda la pantalla LCD mostrando el menú principal donde se observe la hora, la fecha, los valores de la temperatura y humedad relativa actual.

También se verificó que los botones MENU, SALIDA, ARRIBA y ABAJO funcionen de manera correcta y que mediante su uso se pueda ingresar a los submenús y modificar los parámetros de control. En la figura 4.42. se observa el circuito funcionando una vez ensamblado. / Figura 48 Prueba de funcionamiento del circuito una vez ensamblado. Fuente: Elaborado por el autor Luego se comprobaron que los parámetros de temperatura y humedad relativa medidos por el circuito estaban acorde a los valores reales.

Como se puede observar en la figura 4.43, la temperatura medida es 12 grados centígrados y la humedad relativa es del 67%. En la figura 4.43. se puede apreciar los datos medidos con los actuales, los cuales se comparados son aproximadamente los mismos. Ya que al momento de tomar la muestra la temperatura era de 12 grados centígrados y la humedad relativa era del 50%, con lo que se pudo comprobar que las mediciones del circuito eran bastante aproximadas a los valores reales. / Figura 49 Comparación de valores medidos por el circuito.

Fuente: Elaborado por el autor Una vez concluido el funcionamiento del circuito se puso a funcionar la etapa de potencia conectando las luces LED de crecimiento como se muestra en las figuras 4.44 y 4.45. / Figura 50 Luces LED para crecimiento. Fuente: Elaborado por el autor / Figura 51 Planta creciendo con luz LED. Fuente: Elaborado por el autor También se puso en funcionamiento el circuito para probar la etapa de potencia conectando la bomba de agua como se muestra en la figura 4.46. y también el prototipo del aire acondicionado en la figura 4.47. / Figura 52 Prueba de funcionamiento de la bomba de agua.

Fuente: Elaborado por el autor / Figura 53 Prueba de funcionamiento del aire acondicionado. Fuente: Elaborado por el autor La segunda parte de las pruebas se las realizó en el contenedor de carga a escala donde se sembraron varios tipos de hortalizas y se fue monitoreando por varios meses su crecimiento y el funcionamiento del control automático como se observa en las figuras 4.48. y 4.49.

/ Figura 54 Crecimiento de hortalizas en ambiente controlado. Fuente: Elaborado por el

autor / Figura 55 Plantas cultivadas con control automático Fuente: Elaborado por el autor 4.2. Análisis de resultados Al analizar la evolución de la hidroponía en el Ecuador, se puede observar que existen grandes deficiencias y limitaciones tecnológicas para la implementación de este conocimiento hasta la actualidad, ya que en la actualidad la inversión es mucho más costosa que para un cultivo tradicional.

(Ministerio de agricultura y ganadería, 2018) Luego de verificar las múltiples ventajas que tiene en utilizar en este método de cultivo con control automático para cultivos hidropónicos, analizar la mayoría de los factores que son los esenciales y que se deben tener en cuenta para el crecimiento de la planta, como el cuidado de su temperatura, humedad, la iluminación y la nutrición, se les debe brindar la mayor prioridad a la hora de automatizar ya que gracias a ellos se genera la culminación con éxito de los cultivos y adicionalmente hay que tomar en cuenta también que existen otros factores menos importantes; tales como el control del pH, el control de la temperatura de la solución nutritiva, el nivel de la solución, etc. que también pueden ser mejorados.

Este sistema de control automático tiene gran factibilidad para ser implementado en cualquier invernadero, el cual puede ser adaptado para poder desarrollar el proceso deseado, lo cual agregaría un mayor valor a la producción y permitiría llevar al máximo el aprovechamiento del cultivo utilizando la tecnología de punta para automatizar el sistema.

CONCLUSIONES Se definieron los parámetros de control de temperatura, humedad, iluminación nutrición y riego para el sistema de automatización, los cuales fueron controlados de manera satisfactoria a través del sistema de automatización microprocesado cumpliendo con las expectativas planteadas. Se diseñó el sistema de automatización de un contenedor de carga para el control de clima y crecimiento de cultivos hidropónicos según los parámetros establecidos en el objetivo general, haciendo que sea muy versátil ya que se puede adaptar a la mayoría de los cultivos por el amplio rango en su funcionamiento.

Se desarrolló un algoritmo de control y automatización que al ser un sistema autónomo de precisión, permitió manejar todos los parámetros pre-establecidos satisfactoriamente y que los equipos de control funcionen adecuadamente de acuerdo a la programación requerida. Se construyó el sistema automatizado y circuito electrónico para controlar temperatura, humedad, nutrición, riego e iluminación tomando en cuenta que son los factores más importantes para el crecimiento para cultivos hidropónicos, lo cual genero el crecimiento adecuado de las plantas.

Se realizaron las pruebas de funcionamiento y validación de resultados para cultivos hidropónicos y se concluye que el sistema de automatización funciona adecuadamente para los cultivos hidropónicos y podría utilizarse a gran escala ya que los resultados de crecimiento fueron satisfactorios. El manejo de los cultivos hidropónicos automatizados en un contenedor permite un control inteligente de las condiciones ambientales, la iluminación y los nutrientes, los cuales resultan inmanejables en los cultivos de tierra, logrando así una mayor producción y mejorando la calidad de los cultivos.

De datos disponibles de investigaciones de la FAO (<http://www.fao.org/3/a-ah501s.pdf>) se deduce que, si se agrega el sistema de automatización en un contenedor, los costos de inversión inicial son casi cuatro veces mayores que el precio al respecto de un cultivo tradicional en tierra.

Pero viendo la realidad y la producción obtenida, se puede predecir que en aproximadamente 3 años se podría recuperar la inversión inicial. El resultado de este proyecto demuestra que si es posible controlar los parámetros elegidos en la automatización para mejorar la calidad de los cultivos

RECOMENDACIONES En este proyecto se utilizaron varios equipos externos como aire acondicionado, calefactor, humidificador y deshumificador para controlar los parámetros programados, pero hay que tener cuidado al momento de calcular las capacidades de acuerdo con el área de cobertura ya que algunas veces es insuficiente y otras sobrepasa los valores requeridos. Mediante las pruebas realizadas en el contenedor se podría escalar a un desarrollo comercial para aplicaciones en el mundo real.

Otra recomendación es incluir en el diseño un módulo con un reloj en tiempo real RTC (Real Time Clock) ya que para crear los horarios de los tiempos y los intervalos de riego la programación se complica muchísimo ya que existen muchas variantes de acuerdo con el tipo de cultivo y mediante el RTC se simplifica significativamente la programación del microcontrolador.

Es una buena práctica, analizar y escoger el tipo de microcontrolador que se necesite de acuerdo con los requerimientos del proyecto ya que en el mercado actual existen muchos tipos de dispositivos y elegir el más adecuado simplifica mucho el diseño, por ejemplo: en este proyecto se utilizó un módulo Arduino que tenía el módulo Wifi incorporado, lo que generó que la implementación sea mucho más sencilla y en caso de fallo sería mucho más fácil realizar el cambio de un solo componente.

Hay que tener en cuenta que cuando se realice la programación del módulo LCD a través del módulo I2C, para que no existan errores cuando se utilice un display de 16x2 en la programación se debe colocar el número hexadecimal 0x3f y para el display de 20x4 se debe colocar el número hexadecimal 0x27.

Bibliografía Adafruit.com. (2018). Datasheet digital relative humidity & temperature sensor AM2302/DHT22. Retrieved from <https://cdn-shop.adafruit.com/datasheets/Digital+humidity+and+temperature+sensor+AM2302.pdf> Arduino. (2016, noviembre 2). Arduino Uno Wifi. Retrieved from <https://store.arduino.cc/usa/arduino-uno-wifi> Arnon, H. D. (1938).

The water culture method for growing plants without soil. Calif. Agr. Expt. Sta. Cir. 347, 39p. Castilla, W. S. (1995). Protected cultivation of horticultural crops worldwide. Hort technology 5(1), 6-23. Collins, J. M. (1985). Hydroponic vegetable production. Hort. Review 7, 483-558. Dalrymple DG. (1973). A global review of greenhouse food production. USDA. Graves CJ. (1983). The nutrient film technique. Horticultural Review. 5: 1-44. GW, W. (1979).

Nutrient Film Technique. Grower Bulletin 5. Glasshouse Crops Research Institute. Littlehampton, England. Hawthorne Gardening. (2017). Flo-n-Gro Grow Systems. Retrieved from <http://www.flo-n-gro.net/page/five-methods-of-hydroponics/> Jensen, M. H. (2017). Hydroponics Worldwide - A technical Overview, University of Arizona. Tucson, Arizona. Massantini F. (1976). Floating hydroponics; A method of soilless culture, 91-98. Maxim Integrated. (2014). Datasheet DS3231 Extremely Accurate I2C-Integrated.

Retrieved from Datasheet DS3231 Extremely Accurate I2C-Integrated: <https://datasheets.maximintegrated.com/en/ds/DS3231.pdf> Ministerio de agricultura y ganadería. (2018, agosto 16). Lechuga se produce de forma hidropónica. Retrieved from <https://www.agricultura.gob.ec/lechuga-se-produce-de-forma-hidroponica/> Naylamp Mechatronics SAC. (2016). Naylamp Mechatronics. Retrieved from https://www.naylampmechatronics.com/blog/35_Tutorial--LCD-con-I2C-controla-un-LCD-con-so.html Opencircuit. (2018). Datasheet I2C 1602 Serial LCD Module. Retrieved from <https://opencircuit.nl/ProductInfo/1000061/I2C-LCD-interface.pdf> Organización de las Naciones Unidas para la alimentación y la agricultura. (2017). FAO. Retrieved from <http://www.fao.org/home/es/> OSRAM Opto Semiconductors. (2018).

LEDs for horticultural lighting applications. Regensburg. Rduinostar.com. (2015). Características DHT22. Retrieved from <http://rduinostar.com/documentacion/datasheets/dht22-caracteristicas-am2302/> Vishay. (2016). Datasheet LCD-020N004L LCD- 0 x 4 Character LCD. Retrieved from <https://www.vishay.com/docs/37314/lcd020n004l.pdf>

Anexo 1. Código Fuente:

Anexo 2. Datasheet ArduinoUno Wifi:

Anexo 3. Datasheet DTH22 Digital Humidity-Temperature Sensor:

Anexo 4.

Datasheet Real Time Clock Rtc-Ds3231

Anexo 5. Datasheet LCD 20x4 Lcd-020n004l

Anexo 6. Datasheet I2C-LCD-Interface

INTERNET SOURCES:

0% - Empty