



2021



SISTEMA DE RIEGO INTELIGENTE CON SENSORES Y MONITOREO AUTOMATIZADO DE CULTIVO

**SISTEMA DE RIEGO INTELIGENTE CON SENSORES Y MONITOREO
AUTOMATIZADO DE CULTIVO**

AÑO 2021

**DIRECCIÓN DE INVESTIGACIÓN Y PROYECCIÓN SOCIAL
ÁREA TECNOLOGÍA**



UNIVERSIDAD DOCTOR ANDRÉS BELLO
DIRECCIÓN DE INVESTIGACIÓN Y PROYECCIÓN SOCIAL
ÁREA TECNOLOGÍA

631.587

S623 Sistema de riego inteligente con sensores y monitoreo automatizado de cultivo [recurso electrónico] / Mario René Rivas Domínguez, Samuel Alejandro Cano, Josué Noé Monterroza Alemán, Juan José Escuintla Morán ; contribuciones especiales al proyecto Josué Noé Monterroza Alemán, Samuel Alejandro Cano, Oscar Antonio Peña Rodas. - - 1ª ed. - - San Salvador, El Salv. : Universidad Doctor Andrés Bello, 2022.

1 recurso electrónico, (104 p. : il. cuadros, mpas, grafico ; 29 cm.)

Datos electrónicos: (1 archivo, formato pdf, 5.50 mb) - -

<http://www.unab.edu.sv/publicaciones//page/4>.

ISBN 978 – 99961 – 65 – 42 – 9 (E-Book, pdf)

1. Riego-Automatización. 2. Irrigación. 3. Domótica. 4. Humedad de Suelos. 5. Riego-irrigación de cultivos. I. Rivas Domínguez, Mario René, 1981 coaut. II. Título.

BINA / jmh

©2021, Universidad Doctor Andrés Bello

Primera edición 2021

ISBN 978 – 99961 – 65 – 42 – 9 (E-Book, pdf).

La reproducción total o parcial del documento está permitida, solo deberá citarse de forma apropiada la fuente.¹

Código Institucional: **4CR/INV/H/2021**

Investigadores: Mario René Rivas Domínguez, Samuel Alejandro Cano, Josué Noé Monterroza Alemán y Juan José Escuintla Morán.

Contribuciones especiales al proyecto:

Josué Noé Monterroza Alemán: documentación de pruebas de sensores, código de Arduino y construcción de circuito de Prototipo de parcela de Chalatenango.

Samuel Alejandro Cano: documentación de pruebas de sensores, código de Arduino y construcción de circuito de prototipo de parcela de San Miguel.

Oscar Antonio Peña Rodas: traducción del resumen.

¹ Cualquier correspondencia deberá dirigirse a: Dr. Roberto Hernández Rauda, Director de Investigación y Proyección Social, Universidad Doctor Andrés Bello, 1ª Calle Poniente y 41 Av. Norte #2128, Col. Flor Blanca, San Salvador, El Salvador. Casilla de correo electrónico: roberto.rauda@unab.edu.sv , Tel. (503) + 25107455.

Contenido

Prólogo.....	i
Resumen.....	iii
Abstract.....	v
1. Introducción.....	1
2. Conceptos preliminares.....	3
2.1 Domótica.....	3
2.2 Sistemas domóticos.....	3
2.3 Elementos de un sistema domótico.....	4
2.4 Definición de Algoritmo.....	5
2.5 Definición de Arduino.....	5
2.6 App Inventor.....	7
2.7 Sistema de riego.....	8
2.8 Tipo de sistema de riego.....	8
3. Métodos.....	9
3.1 Generalidades.....	9
3.1.1 Metodología de investigación.....	9
3.1.2 Localización y Descripción de la Zona de Estudio.....	9
3.2 Selección de variable a monitorear para activar riego.....	12
3.3 Descripción del entorno de programación.....	12
3.3 Selección de elementos del prototipo.....	16
3.3.1 Descripción de unidades de control.....	16
3.3.2 Sensores.....	18
3.3.3 Actuadores.....	22
3.4 Calibración de sensores.....	28
3.5 Procesos de conexión y de programación de componentes, placa Arduino y aplicación móvil.....	29
3.6 Diagrama de flujo del prototipo de riego.....	37
4. Resultados.....	39
4.1 Características generales del prototipo de riego automatizado.....	39
4.2 Resultados específicos del prototipo de riego automatizado San Miguel....	42
4.2.1 Resultados de pruebas del sensor de humedad del suelo.....	42
4.2.2 Implementación del prototipo de riego.....	43
4.2.3 Código de programa utilizado en prototipo de riego San Miguel.....	46

4.3 Resultados específicos del prototipo de riego automatizado Chalatenango	55
4.3.1 Resultados de pruebas del sensor de humedad del suelo.	55
4.3.2 Implementación del prototipo de riego	62
4.3.3 Código de programa	65
4.4 Aplicación móvil del prototipo de Riego	74
4.4.1 Diseño de Aplicación Móvil (Interfaz del usuario)	74
4.4.2 Algoritmo y programación de la aplicación móvil	74
5. Discusión	77
6. Conclusiones	81
7. Referencias.....	82

Índice de Figuras

<i>Figura 1. Ubicación de parcela demostrativa de San Miguel.</i>	<i>10</i>
<i>Figura 2. Ubicación de parcela demostrativa Chalatenango.</i>	<i>11</i>
<i>Figura 3. Modelo de placa Arduino Uno Rev3.....</i>	<i>17</i>
<i>Figura 4. Modelo de placa Arduino Mega 2560.....</i>	<i>18</i>
<i>Figura 5. Modelo de sensor de humedad de suelo.</i>	<i>18</i>
<i>Figura 6. Sensor de temperatura.....</i>	<i>19</i>
<i>Figura 7. Modelo de módulo de reloj.</i>	<i>21</i>
<i>Figura 8. Módulo Relé</i>	<i>22</i>
<i>Figura 9. Electroválvula</i>	<i>24</i>
<i>Figura 10. Modelo de pantalla LCD.....</i>	<i>25</i>
<i>Figura 11. Modulo Bluetooth.</i>	<i>27</i>
<i>Figura 12. Diagrama de conexión de sensores.</i>	<i>29</i>
<i>Figura 13. Diagrama de conexión de electroválvula con relé y la placa Arduino. .</i>	<i>30</i>
<i>Figura 14. Conexión y adaptación de la electroválvula con relé y la placa Arduino.</i>	<i>31</i>
<i>Figura 15. Diagrama integrado del prototipo del sistema riego San Miguel.</i>	<i>34</i>
<i>Figura 16. Diagrama integrado del prototipo del sistema riego Chalatenango.</i>	<i>35</i>
<i>Figura 17. Diagrama general de conexión de elementos electrónicos del prototipo del sistema.</i>	<i>36</i>
<i>Figura 18. Diagrama de flujo del prototipo de riego.....</i>	<i>38</i>
<i>Figura 19. Parcela de San Miguel con cultivo.</i>	<i>40</i>
<i>Figura 20. Parcela de Chalatenango con cultivo.</i>	<i>41</i>
<i>Figura 21. Datos de prueba 1 de sensores de humedad de suelo.....</i>	<i>42</i>
<i>Figura 22. Datos de prueba 2 de sensores de humedad de suelo.</i>	<i>43</i>
<i>Figura 23. Instalación de tubería parcela San Miguel.</i>	<i>44</i>
<i>Figura 24. Instalación de electroválvula, parcela San Miguel.....</i>	<i>45</i>
<i>Figura 25. Instalación de sensor, parcela San Miguel.....</i>	<i>45</i>
<i>Figura 26. Instalación de unidad de control, parcela San Miguel.</i>	<i>46</i>
<i>Figura 27. Resultados de prueba de sensor de humedad de suelo en diferente ambiente.....</i>	<i>56</i>
<i>Figura 28. Datos de monitoreo prueba de sensores de humedad en software Arduino.</i>	<i>57</i>
<i>Figura 29. Resultados de prueba de dos sensores de humedad de suelo en tierra húmeda.</i>	<i>58</i>
<i>Figura 30. Datos de monitoreo de comportamiento de dos sensores de humedad del suelo en suelo húmedo.....</i>	<i>59</i>
<i>Figura 31. Resultado de prueba de dos sensores de humedad de suelo en tierra subhúmeda.....</i>	<i>60</i>
<i>Figura 32. Datos de monitoreo de comportamiento de sensores en ambiente semihúmedo.....</i>	<i>61</i>
<i>Figura 33. Porcentaje de humedad del suelo para activación de riego.</i>	<i>62</i>

<i>Figura 34. Instalación de tubería en parcela de Chalatenango.</i>	<i>63</i>
<i>Figura 35. Instalación de electroválvula en parcela de Chalatenango.</i>	<i>63</i>
<i>Figura 36. Instalación de sensor en parcela de Chalatenango.....</i>	<i>64</i>
<i>Figura 37. Instalación de unidad de control en parcela de Chalatenango.....</i>	<i>65</i>
<i>Figura 38. Caja de unidad de control en parcela de Chalatenango.....</i>	<i>65</i>
<i>Figura 39. Diseño de pantalla de aplicación móvil.</i>	<i>74</i>
<i>Figura 40. Codificación de aplicación móvil.</i>	<i>75</i>
<i>Figura 41. Pantalla final de aplicación móvil.....</i>	<i>76</i>

Índice de Tablas

<i>Tabla 1. Listado de librerías y funciones del lenguaje Arduino utilizadas.....</i>	<i>13</i>
<i>Tabla 2. Especificaciones técnicas de Microcontrolador Arduino Uno.</i>	<i>16</i>
<i>Tabla 3. Especificaciones técnicas de Microcontrolador Arduino Mega 2560.</i>	<i>17</i>
<i>Tabla 4. Equivalencia entre valor de sensor de humedad y otras unidades.</i>	<i>29</i>
<i>Tabla 5. Datos del monitoreo del sensor de humedad de suelo para calibración .</i>	<i>56</i>
<i>Tabla 6. Datos muestrales de monitoreo de dos sensores de humedad en suelo húmedo.</i>	<i>58</i>
<i>Tabla 7. Datos del monitoreo de dos sensores de humedad en suelo subhúmedo.</i>	<i>60</i>

Prólogo

En la actualidad el uso de los sistemas domóticos tiene mucha relevancia, a tal grado que la aplicabilidad de esta tecnología basada en la combinación de elementos de electrónica, informática y telecomunicaciones ha aumentado en gran proporción. De forma inicial se utilizaba para automatizar actividades domésticas, sin embargo, ahora se aplica en otros campos, por ejemplo, la agricultura de precisión.

Congruente con lo anterior, se considera el riego de cultivos como una actividad muy importante para la seguridad alimentaria, ya que permite la producción continua de cultivos como granos básicos y hortalizas; sin embargo si esta aplicación de riego se hace sin elementos de control lo que puede ocasionar un desperdicio considerable de agua; por tanto, es importante desarrollar herramientas domóticas que contribuyan al ahorro de ese recurso natural, sobre todo en zonas del país donde este recurso es escaso, por ejemplo en localidades ubicadas en el corredor seco.

Por todo ello, se ha desarrollado e implementado un prototipo de riego inteligente, que contiene elementos de control tales como sensores de humedad del suelo y programación de tiempo de riego que permiten hacer un uso óptimo del recurso hídrico destinado para el riego en dos parcelas cultivadas con maíz. De igual forma se ha desarrollado una aplicación para teléfono móvil destinado al monitoreo de los datos y a la comunicación entre el usuario y los componentes del prototipo.

Finalmente, la Universidad Doctor Andrés Bello pone a disposición todos los resultados de la presente investigación, con la finalidad de servir como insumo o complemento para estudios posteriores en domótica.

MBA. Iveth Escobar de Umanzor
Rectora
Universidad Doctor Andrés Bello

Resumen

La presente investigación tiene como objetivo desarrollar un prototipo de riego inteligente mediante el uso de un microcontrolador, sensores y actuadores; sistema que fue implementado en dos parcelas demostrativas, la primera ubicada en Sede Regional San Miguel y la otra en Sede Regional de Chalatenango. La metodología de investigación utilizada fue dirigida a dar respuesta a una problemática respecto a la automatización del suministro de agua al cultivo de maíz a través de un sistema domótico.

El resultado principal fue el diseño, programación e implementación de un prototipo de riego inteligente conformado por una placa Arduino Uno con función de microcontrolador, la cual interactúa con sensores de humedad de suelo ubicados en diferentes puntos de las parcelas y éstos emiten a través de una salida analógica la cuantificación de humedad de suelo, que es el factor determinante para la aplicación de riego; para ello la señal del sensor es procesada por la unidad de control, que posteriormente la envía hacia un relé cuya función es abrir o cerrar una electroválvula, permitiendo así la irrigación del cultivo durante un determinado periodo de tiempo. Todo el funcionamiento está monitoreado por una aplicación móvil la cual sirve como interfaz de comunicación entre el usuario y el sistema de riego.

Otro aspecto importante es que el diseño del prototipo permite adaptarlo a diferentes tipos de suelo y cultivo al modificar los parámetros provenientes del sensor y validados por el software para determinar el momento y la cantidad de tiempo a aplicar el riego de forma automática.

El desarrollo de este prototipo de riego inteligente es funcional en las dos parcelas donde se implementó; hace un uso óptimo de agua ya que controla automáticamente el riego en el área donde se encuentra el cultivo, con base al requerimiento del suelo según su valor de humedad y ha sido programado para regar en horas en las cuales es mayor la absorción del agua por el suelo.

Palabras claves: Domótica, riego automatizado, sensores de humedad de suelo, Arduino.

Abstract

The objective of this research is to develop an intelligent irrigation prototype through the use of a microcontroller, sensors and actuators; a system that was implemented in two demonstration plots, the first located in the San Miguel Regional Campus and the other in the Chalatenango Regional Campus. The research methodology used was aimed at responding to a problem regarding the automation of water supply to corn crops through a home automation system.

The main result was the design, programming and implementation of an intelligent irrigation prototype made up of an Arduino Uno board with a microcontroller function, which interacts with soil moisture sensors located at different points in the plots and these emit through an analog output the quantification of soil moisture, which is the determining factor for the application of irrigation; For this, the signal from the sensor is processed by the control unit, which subsequently sends it to a relay whose function is to open or close an electro-valve, thus allowing crop irrigation for a certain period of time. All operation is monitored by a mobile application which serves as a communication interface between the user and the irrigation system.

Another important aspect is that the design of the prototype allows it to be adapted to different types of soil and crops by modifying the parameters coming from the sensor and validated by the software to determine the moment and the amount of time to apply irrigation automatically.

The development of this intelligent irrigation prototype is functional in the two plots where it was implemented; makes optimal use of water since it automatically controls irrigation in the area where the crop is located, based on the soil's requirement according to its moisture value and has been programmed to irrigate at times when water absorption is greater by soil.

Keywords: Home automation, automated irrigation, soil moisture sensors, Arduino.

1. Introducción

La presente investigación trata sobre el desarrollo tanto del diseño como la programación de un prototipo de riego inteligente mediante el uso de microcontroladores, estudio que tiene por objetivos específicos los siguientes:

- Identificar requerimientos de información de un prototipo de sistema de riego inteligente controlado por variables ambientales.
- Definir los componentes del sistema (microcontrolador, sensores y actuadores), variables de control y tipo de riego que integraran el prototipo.
- Desarrollar una aplicación móvil como interfaz de comunicación entre el microcontrolador y el usuario que permita monitorear el riego automatizado.
- Implementar el prototipo diseñado en dos parcelas demostrativas para desarrollar pruebas operativas, comprobar su funcionamiento y sistematizarlo.

Posterior a la definición de los objetivos se procedió a explorar diferentes zonas para seleccionar las parcelas demostrativas donde se pondría en marcha el prototipo de riego automatizado; para ello se seleccionaron dos lugares, el primero ubicado en el polideportivo de la Universidad Doctor Andrés Bello Centro Regional San Miguel y el segundo una parcela aledaña al Centro Regional Chalatenango.

En la primera sección se presenta una investigación bibliográfica sobre el tema que contempla una colección de conceptos, los cuales son importantes de conocer para tener una contextualización de la investigación por parte del lector, algunas definiciones consideradas son: domótica, sistema domótico y sus elementos: como el microcontrolador y sus componentes de entrada y salida, los lenguajes de programación utilizados y los diferentes tipos de riego, entre otros. Se presenta también una breve descripción sobre la zona de estudio donde se implementa el prototipo de riego.

El segundo apartado de este estudio corresponde a métodos, en este se describen: el tipo de investigación realizada, los materiales y los elementos utilizados en el prototipo (placas Arduino o microcontroladores, los sensores, controladores con sus

especificaciones técnicas), el funcionamiento y procedimiento de integración al microcontrolador para efectos de ser retomados en estudios posteriores. Se listan también las principales funciones del lenguaje Arduino utilizadas, ya que son fundamentales para controlar el funcionamiento de los componentes electrónicos a fin de que produzcan los resultados esperados.

En este estudio se consideraron dos diseños, el primero describe el componente electrónico, para el cual se utilizaron diagramas individuales para componentes como sensores de humedad de suelo, electroválvula, módulo Bluetooth para la comunicación de datos y un diagrama integrado el cual contempla toda la circuitería del prototipo; se presenta también el diseño hidráulico que describe cómo se aplicará el riego en las parcelas a través de la interconexión de tuberías, fuente de agua, entre otros aspectos.

La fase de desarrollo conocida también como programación contempla los elementos siguientes: un flujograma que describe el comportamiento lógico del prototipo, desde la lectura de los sensores de humedad del suelo pasando por la verificación del parámetro de control de activación del riego hasta terminar en la activación del riego, a través del control de la electroválvula y el tiempo de aplicación de riego. Se incluye también el diagrama de flujo de la aplicación móvil diseñada para el monitoreo de variables y el control del proceso de riego por parte del usuario.

El resultado principal de este estudio es la integración de todos los componentes electrónicos y físicos de riego; por ello en la sección correspondiente, se presentan una serie de prueba de los sensores realizadas en diferentes condiciones de terreno que fue importante para garantizar el funcionamiento del prototipo de riego, la descripción de todo el sistema de riego, la codificación que hace funcionar los componentes del prototipo y el detalle de la pantalla de la aplicación móvil diseñada, que muestra los valores capturados en cada sensor y un mensaje de alerta al usuario indicándole si las condiciones son favorables para activar el riego, adicionalmente, se incluye la codificación de esta aplicación.

Como conclusión principal se demostró que el prototipo de riego inteligente implementado en las parcelas demostrativas de San Miguel y Chalatenango funciona satisfactoriamente de acuerdo a lo diseñado; además la automatización de este proceso permitió garantizar el uso óptimo de agua ya que se focaliza el riego. Adicionalmente, el diseño y programación del prototipo tienen la característica de adaptabilidad a otros estudios de este tipo.

2. Conceptos preliminares

2.1 Domótica

La domótica se puede definir como la tecnología que combina elementos de tres disciplinas tales como: electrónica, informática y telecomunicaciones, con el propósito de automatizar la realización actividades domésticas, agrícolas e industriales, a través de la programación de diferentes dispositivos cuyo funcionamiento pueden integrarse mediante una aplicación móvil o un sistema informático (Merino & Guzmán, 2015, p. 24).

De acuerdo a la Asociación Española de Domótica e Inmótica (CEDOM), en su sitio web (<http://www.cedom.es/es>) dos aspectos muy importantes a agregar a la definición anterior son: en primer lugar, el adjetivo de inteligente que adopta todo proceso al automatizarse su ejecución ya que en muchos casos el sistema domótico se activa al haber un cambio en la señal emitida por un sensor; y el otro es el enfoque de ahorro en el consumo de recursos o el uso eficiente de estos mediante la programación de procedimientos.

2.2 Sistemas domóticos

Según la CEDOM, un sistema domótico es “todo sistema capaz de recoger información proveniente de unos sensores o entradas, procesarla y emitir órdenes a unos actuadores o salidas, además estos sistemas pueden acceder a redes exteriores de comunicación o información”

2.3 Elementos de un sistema domótico.

Un sistema domótico está constituido por una serie de componentes electrónicos y algoritmos que hacen que éstos interactúen entre sí; dichos elementos se pueden clasificar en base a la función que realizan en el sistema de la siguiente manera:

Elementos de entrada

Estos se caracterizan por detectar cambios de estados en una variable del entorno, dato que captan y envían a través de un canal de comunicación hacia la unidad de control del sistema para ser procesado (Fernández, Maestro y Gabilondo, 2015).

El principal componente de entrada se conoce como sensor; éste se define como: “dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas” (Rojas & Areiza, 2015, p.28). Algunos ejemplos de variables medibles a través de sensores son: temperatura, humedad relativa del aire y del suelo, distancia, presión atmosférica, fuerza, etcétera.

Elementos de procesos:

Componentes encargados de procesar los datos enviados por los sensores y producir señales de salida; entre estos están:

Unidad de control: la función principal de este elemento es transformar los datos que recibe de los sensores y luego de procesados enviarlos en forma de órdenes a los actuadores; en este componente es donde radica lo que se conoce como inteligencia del sistema y es similar a la unidad aritmética lógica de una computadora (Fernández, Maestro y Gabilondo, 2015).

Elementos de salida:

Para Rojas & Areiza (2015), un actuador es el principal elemento de salida en un sistema domótico y se refiere a todo dispositivo que toma como insumo la energía transformada en la unidad de control; lo cual provoca la activación de un proceso con el propósito principal de generar un efecto cambiando la situación en el entorno o el estado en el proceso automatizado, por ejemplo, abrir o cerrar una electroválvula con base a un valor de humedad del suelo (p. 29).

2.4 Definición de Algoritmo

Es una secuencia de pasos ordenados y con una lógica de programación similar a una especie de receta que permite desarrollar el flujo de trabajo de todo sistema y/o aplicación; en este caso particular se refiere a la transformación de datos de sensores en órdenes para actuadores; en muchas ocasiones para efectos de análisis de la solución al problema que se trate también se hace uso de herramientas gráficas como flujogramas, los cuales facilitan el diseño de la solución a través de la construcción de secciones de código en diferentes lenguajes de programación (Méndez, 2013, p. 15).

2.5 Definición de Arduino

Para el desarrollo de todo proyecto de domótica es muy importante la plataforma donde realizar la programación y la integración de elementos, por ello se presenta una definición de la plataforma Arduino utilizada en la presente investigación; que fue tomada del sitio web de la empresa Ingeniería MCI (<https://arduino.cl/>) .

Es una plataforma de desarrollo basada en una placa electrónica de hardware libre que incorpora un microcontrolador reprogramable y una serie de pines hembra. Estos permiten establecer conexiones entre el microcontrolador y los diferentes sensores y actuadores de una manera muy sencilla.

Sin embargo, autores como Torrente (2013) amplían la definición de Arduino, considerando tres enfoques:

El primero, describe físicamente la placa Arduino que es de hardware libre, y enfatiza en la descripción físicamente, es decir los componentes tangibles de dicha placa; entre los cuales se tiene un microcontrolador que es donde carga y ejecutan los programas o líneas de código, pero también contiene una serie de pines que permite la conexión de elementos de entrada o sensores y elementos de salida o actuadores con la unidad central o microcontrolador.

El segundo enfoque, se refiere a la descripción de Arduino como entorno de desarrollo; el cual tiene la característica de ser libre y puede instalarse en diferentes

sistemas operativos, lo cual lo hace una herramienta multiplataforma; y para ponerlo en funcionamiento se debe instalar en una computadora. Posteriormente se deben cargar los programas desarrollados en la memoria de la unidad central de la placa y ejecutar los procedimientos.

El tercer y último enfoque, define Arduino como un lenguaje de programación y hace énfasis en el conjunto de instrucciones, comandos, operaciones con datos, bloques ya sea condicionales o cíclicos, los tipos de variables propios de dicho lenguaje, que se pueden programar a través de líneas de código y sobre todo las reglas de programación que rigen dicho lenguaje.

Ventajas de Arduino

A continuación, se describen algunas características ventajosas del uso de la plataforma Arduino en proyectos de domótica, de acuerdo a Torrente (2013).

Libre y extensible: el termino de libre se refiere a que la plataforma está abierta ya que cualquier usuario puede aportar al mejoramiento de ésta; y extensible es debido que se pueden agregar procedimientos, código, extensiones o interfaces para sensores y actuadores; es decir se pueden hacer modificaciones a nivel de software o lenguaje de programación y hardware o placas.

Uso extendido a una gran comunidad: esta ventaja está ligada a la anterior ya que al ser libre hace que muchos usuarios en todo el mundo ayuden al mejoramiento de la plataforma tanto en creación de tutoriales o documentación de procedimientos los cuales se comparten en un ambiente colaborativo.

El entorno de programación es multiplataforma: esto se refiere a la viabilidad de instalación y ejecución del entorno de desarrollo en múltiples sistemas operativos como Windows, Mac OS y Linux.

Su entorno y el lenguaje de programación son simples y claros: la interfaz de usuario de Arduino es intuitiva, por tal razón es una herramienta fácil de manejar, desde la instalación hasta la generación de código de programación; a esto hay que

agregar que la documentación se está actualizando constantemente para cada versión, formato e idiomas.

Las placas de bajo costo, reutilizables y versátiles: el concepto de reutilizable es debido a que ya finalizado el proyecto se puede cargar un nuevo código requerido por otro proyecto y la placa sigue siendo funcional; respecto a la versatilidad es porque las placas de Arduino al tener diferentes tipos de entrada y salida le permite recibir señales de múltiples sensores y enviar órdenes a múltiples actuadores.

2.6 App Inventor

De acuerdo a la Asociación Programa Ergo Sum en su sitio web (<https://www.programoergosum.es/>), define la plataforma App Inventor de la siguiente manera:

Es un entorno de desarrollo de software creado por Google para la elaboración de aplicaciones destinadas al sistema operativo de Android. El lenguaje es gratuito y se puede acceder fácilmente de la web. Las aplicaciones creadas con AppInventor están limitadas por su simplicidad, aunque permiten cubrir un gran número de necesidades básicas en un dispositivo móvil.

El mismo autor también señala las ventajas siguientes que se tienen al programar con AppInventor:

Creación de aplicaciones por medio de bloques de manera intuitiva y gráfica, sin necesidad de saber código de programación.

Acceso al entorno de desarrollo en cualquier momento y cualquier lugar siempre que estemos conectados a internet.

Diversidad de formas de conectividad: directa, wifi e incluso por medio del emulador.

Permite descargar la aplicación mediante el archivo en formato apk a nuestra computadora, lo cual es importante para hacer las respectivas pruebas de la aplicación desarrollada.

2.7 Sistema de riego

El tema principal de la presente investigación es la automatización de riego a través de procesos de domótica, por eso se adopta la definición de Pinto y Cristoph (2014): “se denomina sistema de riego a las técnicas y mecanismos que se utilizan para infiltrar el agua en la tierra con el fin de satisfacer las necesidades hídricas de un cultivo” (p.42).

2.8 Tipo de sistema de riego.

Las definiciones de algunos de los principales sistemas de riego, presentados a continuación, fueron obtenidas del sitio web oficial de Ambientum.com (https://www.ambientum.com/enciclopedia_medioambiental/suelos/sistemas_de_riego.asp) y de Cortez & Vargas (2020).

Riego por aspersión. Este método, de uso general, se aplica sobre grandes superficies lanzando un gran volumen de agua controlada y uniforme en forma de lluvia. Por lo general, actúa cubriendo toda el área y es muy adecuado para automatizar la operación.

Riego con difusores. Distribuye el agua en forma de pequeñas gotas. La difusión del agua se realiza a través del aire, siendo nula la participación del suelo en estos riegos. El difusor más utilizado es el emergente, que reparte el agua en zonas ajardinadas, plantaciones de césped o arbustos pequeños.

Riego por goteo. El agua se distribuye puntualmente, sin atomización y sin que empape el terreno. La densidad de puntos de riego humedece ciertas zonas, mientras la mayor parte del terreno permanece seco. Este tipo de riego es aplicable tanto para zonas tan limitadas como la plantación de arbolado, borduras de arbustos, jardineras, o bien para plantaciones extensas, ya que sólo humedece las zonas deseadas. Este sistema puede complementar al riego por aspersión generalizado en zonas concretas o mantener áreas de arbustos y arbolado de alineación de forma autónoma.

Riego exudante. Es una variación del riego por goteo y se conoce también como tubería exudante. Consiste en una serie de canalizaciones plásticas,

porosas, que permiten que el agua que circula pase al suelo. De esta manera se consigue un riego uniforme y constante en toda su longitud, ya que el agua se transmite por la propia capilaridad del terreno, alcanzando más o menos superficie en función de la estructura de los substratos.

Riego automatizado. Es un sistema construido para la distribución de agua a las plantas de manera controlada por ciertos parámetros previamente definidos, por medio de sistemas de aspersión, micro aspersión o goteo. Este sistema permite repartir el agua en la ubicación, cantidad, frecuencia y horario que se programe según las necesidades (Cortez & Vargas, 2020, p. 32).

3. Métodos

3.1 Generalidades

3.1.1 Metodología de investigación

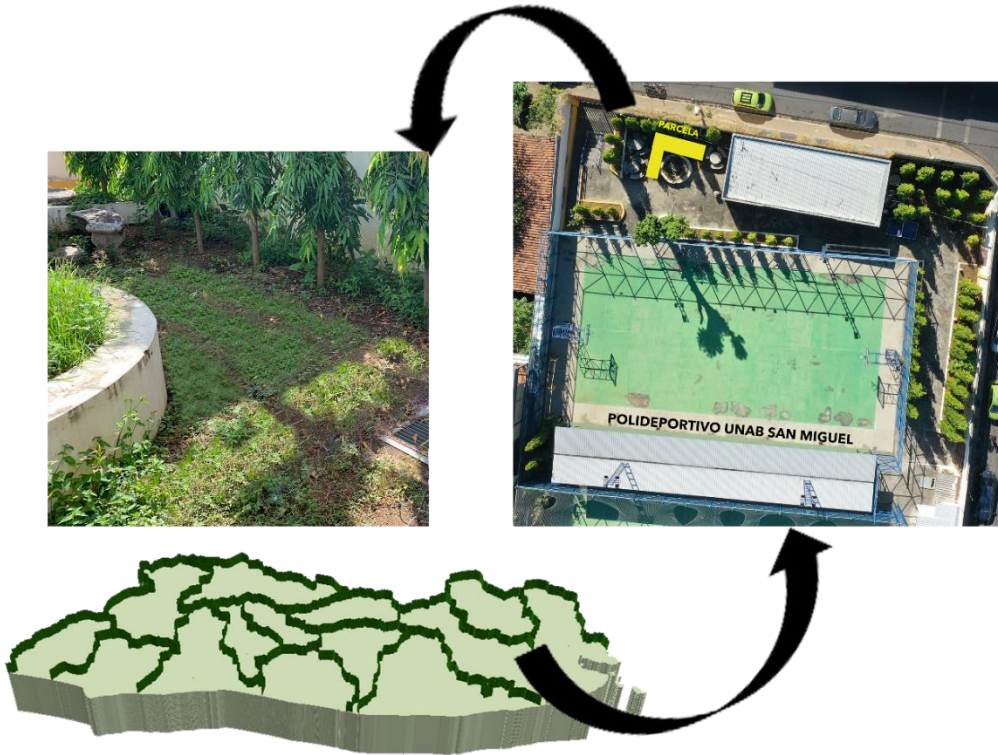
El tipo de investigación utilizada fue dirigida, la cual se centra en el desarrollo de actividades con el propósito de encontrar evidencia que permita resolver una problemática práctica o un cuestionamiento teórico (Mora, 2005); para este caso particular la optimización de recurso hídrico mediante la automatización del proceso de riego en dos parcelas de diferentes dimensiones cultivadas con maíz.

3.1.2 Localización y Descripción de la Zona de Estudio.

El prototipo de riego inteligente fue implementado en dos parcelas propiedad de la Universidad Doctor Andrés Bello, descritas en las figuras 1 y 2.

Parcela demostrativa en San Miguel

Figura 1. Ubicación de parcela demostrativa de San Miguel.



Fuente: Elaboración propia.

La parcela demostrativa está ubicada en el Polideportivo del Centro Regional de la Universidad Doctor Andrés Bello de la ciudad de San Miguel, tiene una extensión aproximada de 22.6 m², cuenta con una fuente de agua potable y de energía eléctrica, elementos básicos para implementar un sistema domótico.

Ubicación

Departamento: San Miguel

Municipio: San Miguel

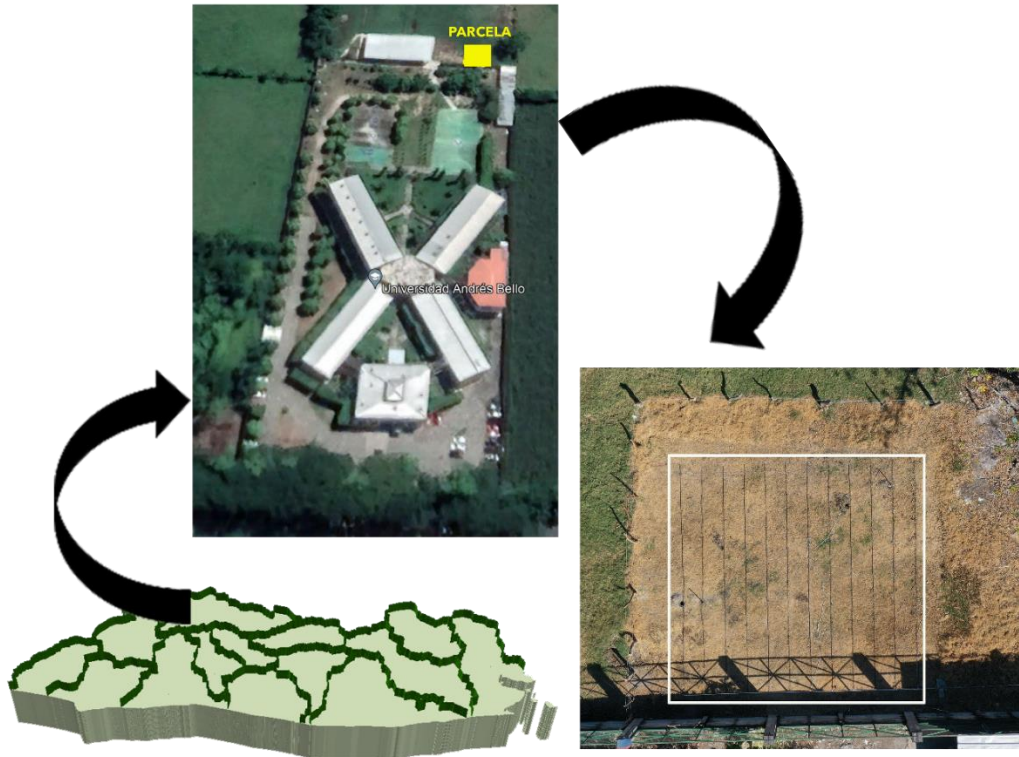
Latitud: 13°28'37.86"N

Longitud: 88°10'39.36"O

Altitud: 123 m.s.n.m.

Parcela demostrativa en Chalatenango

Figura 2. Ubicación de parcela demostrativa Chalatenango.



Fuente: Elaboración propia.

La segunda parcela demostrativa está ubicada en el predio aledaño al campus del Centro Regional de la Universidad Doctor Andrés Bello de la ciudad de Chalatenango, tiene una extensión de: 100 m², al igual que la parcela de San Miguel tiene suministro de agua potable y de energía eléctrica.

Ubicación

Departamento: Chalatenango

Municipio: Chalatenango

Latitud: 14° 2'39.81"N

Longitud: 88°58'46.01"O

Altitud: 292 m.s.n.m.

Luego de haber seleccionado la ubicación física donde implementar el prototipo de sistema de riego inteligente, se procedió a definir los elementos electrónicos a utilizar iniciando con la selección de unidades de control las cuales se describen a continuación.

3.2 Selección de variable a monitorear para activar riego.

Un elemento muy importante a considerar en todo sistema de riego automatizado es el factor condicionante que active la irrigación del suelo, por ello en este estudio se tomó la variable humedad del suelo como ese indicador; y de acuerdo a la Organización de las Naciones Unidas para la Alimentación (FAO) citado por Riego.org en su sitio web (<https://www.riego.org/glosario/humedad-del-suelo-fao/>) ésta se define como: “porcentaje de agua en el suelo (en peso). Normalmente se considera la cantidad de agua presente en el suelo entre el punto de marchitamiento y la capacidad de campo”; y es justamente ese factor de gradiente que tiene esta variable y la cantidad de sensores que hay para medirlo que llevó a tomar esa decisión de aplicarlo en el prototipo de riego.

3.3 Descripción del entorno de programación

La definición del tipo y lenguaje de programación, es vital importancia ya que de éste depende el funcionamiento de toda placa Arduino y sus dispositivos.

Tipo de programación

La metodología utilizada para la programación tanto para la funcionalidad de las placas Arduino como para la interfaz de usuario (aplicación móvil), fue programación estructurada la cual, según Pérez (2021) es un modelo que se basa en tres elementos: el primero es que cuenta una estructura básica para desarrollar programas compuesta por sentencias de asignación de valores, de bifurcación y de bucles de forma secuencial; el segundo se refiere al uso de recursos abstractos como pseudocódigo y/o flujogramas; herramientas que permiten desarrollar la lógica o flujo de trabajo de los programas, así también conocer resultados de un programa previo a su codificación y el tercero es que se basa en el diseño descendente, lo

cual significa que a medida se baja de nivel la complejidad va creciendo y se va haciendo más específica la tarea a programar (p. 2).

Lenguaje de programación

Se refiere al conjunto de instrucciones, palabras reservadas y reglas específicas con las cuales se construye un programa; a través de estos lenguajes, los usuarios pueden codificar cualquier algoritmo diseñado; además cuentan con la función de compilación la cual sirve para detectar errores en el código y transformarlo a lenguaje máquina (ceros y unos) que es lo que interpreta el CPU al momento de ejecutar un programa, en este caso la unidad de control de Arduino (Sznajdleder, 2017, p.24).

La codificación del programa cargado en el microcontrolador de la placa Arduino fue desarrollado en el lenguaje de programación del mismo nombre; la tabla 1 detalla las herramientas de programación utilizadas.

Tabla 1. Listado de librerías y funciones del lenguaje Arduino utilizadas.

Nombre	Descripción	Uso en el proyecto
Librerías		
SoftwareSerial	Su función es permitir la comunicación en serie en pines digitales del Arduino.	Se declaran al inicio de la codificación para tener acceso a las funciones respectivas de cada librería, para procesar los datos.
Wire	Sirve para establecer comunicación entre Arduino y dispositivos con tecnología I2C / TWI, referido a un bus de comunicaciones en serie.	
TimeLib	Esta librería contiene una colección de funciones relacionada con fecha y hora.	
RTClib	Su función permitir la interacción entre Arduino y una variedad de modelos de circuitos integrados a reloj en tiempo real (RTC), dispositivos capaces de almacenar y	

llevar la cuenta de la fecha y hora de manera autónoma.

LiquidCrystal Esta librería tiene como propósito controlar pantallas I2C.

DHT Biblioteca Arduino para sensores de temperatura y humedad DHT11, DHT22, etc.

Funciones

Setup	Es la función inicial de todo programa de Arduino, su utilidad es inicializar variables, modos de pin e inicio de uso de librerías.	Inicializar las variables relacionadas a los pines que vienen de los sensores de humedad y el de la electroválvula.
Loop	Esta función se utiliza después de la función setup y sirve para anidar código que se repetirá constantemente como lectura de entradas y activación de salidas.	
Serial.begin()	Esta función es para establecer la velocidad transmisión de datos su unidad de medida es bits por segundo o baudios, soporta diferentes tasas de transmisión de datos.	Establecer la comunicación entre la placa Arduino y la computadora al momento de cargar el código.
PinMode	Su función es configurar cada uno de los pines, especificando su comportamiento como entrada o como salida.	Configurar el pin de lectura de datos de sensor de humedad y el de salida para pantalla led, electroválvula entre otros.
AnalogRead	Lee el valor del pin analógico especificado, para ello se auxilia del convertidor de analógico a digital que posee Arduino.	Capturar el valor de los sensores de humedad.
DigitalRead	Lee el valor o estado de un pin digital especificado, ya sea HIGH o LOW.	Al igual que el anterior pero cuando ya ha sido convertida la señal a digital este valor se dirige ese valor a un pin de salida digital.

DigitalWrite	Escriben un valor HIGH o LOW en un pin digital de salida.	Dependiendo el valor de humedad leído así asigna ese valor al pin y posteriormente hace el efecto de activar o desactivar la electroválvula.
Delay	Pausa el programa durante el tiempo (en milisegundos) especificado como parámetro.	Controlar los tiempos de lectura de datos de los sensores de humedad.
setCursor	Coloca el cursor de la pantalla LCD; es decir, establece la ubicación en la que se mostrará el texto subsiguiente escrito en la pantalla LCD.	Mostrar valores leídos de los sensores en pantalla led y mensajes de alerta del sistema.
String	Función que sirve como constructor de instancias de la clase cadenas de caracteres; dependiendo de la versión que se use puede construir cadenas a partir de múltiples fuentes de datos.	Generar mensajes de texto que se desplegaran en móvil o pantalla led en base a interpretación de valores de los sensores.
Print	Esta función imprime datos en el puerto serie como texto ASCII legible por humanos, la impresión se refiere al envío de datos de salida por el canal correspondiente.	Desplegar datos en móvil o pantalla led generados con función string.
Println	Imprime datos en el puerto serie como texto ASCII legible por humanos seguido de un carácter de retorno de carro y un carácter de nueva línea.	Igual que print solo que ordena el texto de salida con saltos de línea.
Map	Su propósito es volver a mapear o asignar un número o valor de una variable de un rango a otro.	Mapear el resultado del pin analógico, entre más cercano se este valor del aire será 0%, y entre más cercano al valor de agua será el 100%.

Fuente: Tomado de Guía de Referencia de Arduino. Por Arduino CC, s.f. (<https://www.arduino.cc/reference/es/>)

3.3 Selección de elementos del prototipo

Anteriormente se ha descrito diferentes metodologías y elementos base para la elaboración de un prototipo de riego; sin embargo, el carácter inteligente se lo da el conjunto de dispositivos de entrada o sensores y salida o actuadores, por ello a continuación se describen los elementos utilizados en el prototipo de esta investigación.

3.3.1 Descripción de unidades de control

Para el desarrollo de este sistema automatizado se utilizaron placas Arduino de diferente modelo en cada una de las parcelas demostrativas (Tablas 2 y 3). La placa utilizada en Chalatenango cuenta con más capacidad de hardware; sin embargo, el objetivo era comprobar los requerimientos mínimos de funcionamiento del prototipo de riego automatizado.

Característica de placa Arduino utilizada en parcela demostrativa de San Miguel.

Un elemento básico para todo sistema domótico es el microcontrolador. La tabla 2 muestra las especificaciones técnicas de la placa utilizada en la parcela de San Miguel.

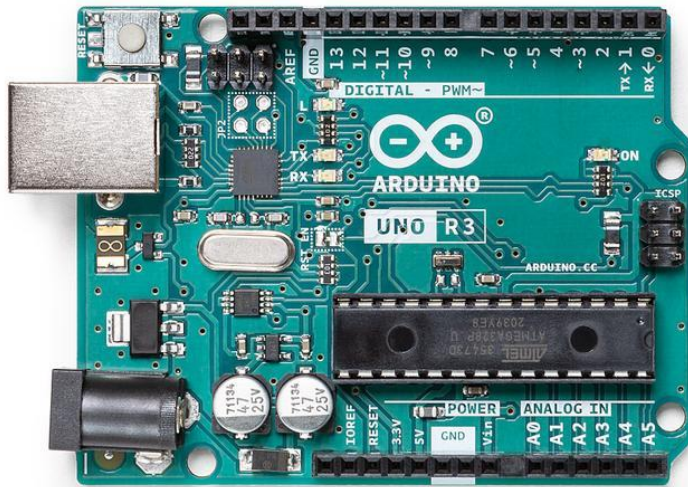
Tabla 2. Especificaciones técnicas de Microcontrolador Arduino Uno.

Categoría	Especificación
Modelo	Arduino Uno
Microcontrolador	ATMega328P
Velocidad de reloj:	16 MHz.
Voltaje de trabajo:	5V.
Voltaje de entrada:	7,5 a 12 voltios.
Pinout:	14 pines digitales (6 PWM) y 6 pines analógicos.
Memoria:	32 KB Flash (0,5 para bootloader), 2KB RAM y 1KB Eeprom

*Fuente: Tomado de Blog de Tecnologías. Por R.J. Suarez, 2021
(<https://www3.gobiernodecanarias.org/medusa/ecoblog/rsuagued/>)*

En la figura 3 se muestra un ejemplar de este modelo de microcontrolador.

Figura 3. Modelo de placa Arduino Uno Rev3.



Fuente: Arduino Uno Rev3. Por Arduino, 2021 (<https://store-usa.arduino.cc/collections/boards/products/arduino-uno-rev3>)

Característica de placa Arduino utilizada en parcela demostrativa de Chalatenango.

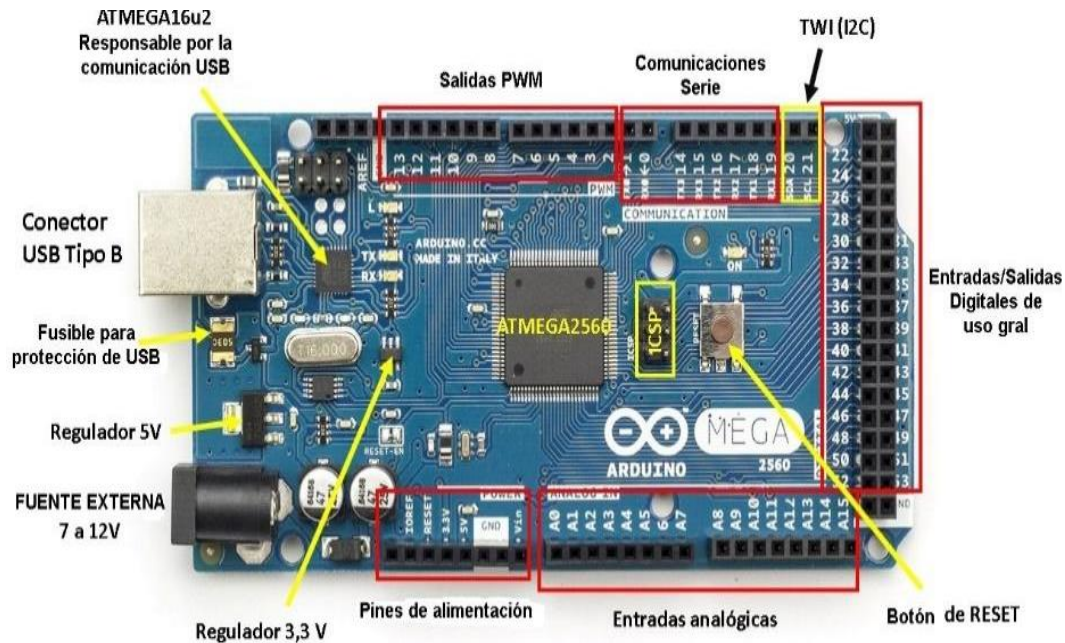
Al igual que en la parcela de San Miguel, en la tabla 3 se describe el microcontrolador utilizado en la parcela de Chalatenango.

Tabla 3. Especificaciones técnicas de Microcontrolador Arduino Mega 2560.

Categoría	Especificación
Modelo	Arduino Mega 2560
Microcontrolador	ATmega2560
Voltaje Operativo	5V
Tensión de Entrada	7-12V
Voltaje de Entrada(límites):	6-20V
Pines digitales de Entrada/Salida	54 (de los cuales 14 proveen salida PWM)
Pines análogos de entrada	16
Corriente DC por cada Pin	Entrada/Salida: 40 mA Corriente DC entregada en el Pin 3.3V: 50 mA
Memoria	Flash: 256 KB SRAM: 8KB EEPROM: 4KB
Velocidad de reloj	16 MHz

Fuente: Tomado de Arduino Mega 2560 Características. Por Electrontools, 2021 (<https://www.electrontools.com/Home/WP/arduino-mega-2560-caracteristicas/>)

Figura 4. Modelo de placa Arduino Mega 2560



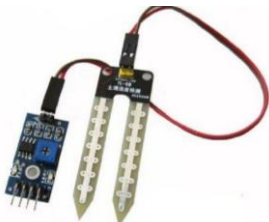
Fuente: Tomado de Arduino Mega 2560 Rev3. Por Arduino, 2021 (<https://store.arduino.cc/products/arduino-mega-2560-rev3>)

3.3.2 Sensores

Las figuras 5, 6 y 7 muestran los modelos de los diferentes sensores utilizados, una descripción de sus características principales y la forma cómo se conectan al microcontrolador.

Sensores de humedad de suelo

Figura 5. Modelo de sensor de humedad de suelo.



Fuente: Tomado de Tutoriales: Sensor de humedad de suelo YL-69. Por Niple Blog, 2016 (<http://www.niplesoft.com/blog/2016/01/19/sensor-de-humedad-de-suelo-yl-69/>)

Descripción: para Félix Alises en su sitio de Google (<https://sites.google.com/site/tecnologiafelixalises/tema-3>); afirma que “estos

sensores están formados por dos láminas metálicas muy próximas cuya resistencia o capacidad varía con la humedad por medio de 2 electrodos resistivos”.

El modelo del sensor utilizado es YL69, se trata de un higrómetro que tiene por utilidad detectar la humedad del suelo; está conformado por dos componentes que son una tarjeta electrónica para la interconexión con la placa Arduino y una sonda con dos almohadillas en la punta que son las que al entrar en contacto con la superficie pueden detectar el contenido de agua en el suelo.

Conexión de sensor de humedad de suelo con Arduino

PIN YL-69

GND GND

VCC VCC 5V

Señal del sensor Pines analógicos A0 y A1

Características:

- Voltaje de alimentación: 3.3V - 5V DC (VCC).
- Corriente de operación: 35mA.
- Voltaje de señal de salida analógico (AO): 0 a VCC; -Voltaje de señal de salida digital (DO): 3.3V/5V TTL.
- Superficie de electrodo: Estaño.
- Vida útil electrodo sumergido: 3 a 6 meses.

Sensor de temperatura

Figura 6. Sensor de temperatura.



Fuente: Tomado de Sensor Digital de Temperatura y Módulo AM2302 DHT2. Por MCI Electronics, (<https://www.mcielectronics.cl/shop/product/sensor-digital-de-temperatura-y-modulo-am2302-dht2-29682>).

Descripción: según la empresa alemana Rechner Sensors, en su sitio web en español (<https://www.rechner-sensors.com/es>), define estos sensores de temperatura como: “componentes eléctricos y electrónicos que, en calidad de sensores, permiten medir la temperatura mediante una señal eléctrica determinada”.

De igual forma la empresa C&D TechNologia en su sitio web (<https://cdtecnologia.net/>), destaca como función principal del sensor DHT22 (AM2302) monitorear la temperatura y humedad relativa, así también menciona entre sus ventajas: buena precisión de lectura, buen rendimiento y un costo bajo; se compone de los siguientes elementos: sensor capacitivo de humedad y un termistor, que se utiliza para medir el aire circundante; los datos que captura este sensor se emiten a través de una señal digital.

Conexión de Sensor de Temperatura con Arduino

PIN Sensor DHT22.

4 GND GND

1 VCC VCC

2 DATAPIN 10

Características técnicas:

- Alimentación de 3,3v a 6v
- Consumo de corriente de 2,5mA
- Señal de salida digital
- Rango de temperatura de -40°C a 125°C
- Precisión para medir temperatura a 25°C de 0.5°C de variación
- La resolución para medir temperatura es de 8-bit, 0,1°C
- La humedad puede medir desde 0% HR (Humedad relativa) hasta los 100% HR
- Con precisión para la humedad del 2-5% HR para temperaturas que se encuentren entre 0-50°C

- La resolución es de 0,1% HR, no puede captar variaciones por debajo de ese límite.

Sensor: Modulo reloj

Figura 7. Modelo de módulo de reloj.



Fuente: Tomado de DS3231 Real Time Clock Module – AT24C32 RTC With Battery. Por IT-ELECTRONICS COMPONENTS, 2021 (<https://itelectroniccomp.com/product/ds3231-real-time-clock-module-at24c32-rtc-with-battery/>)

Descripción: de acuerdo a la empresa Naylamp Mechatronics SAC en su página en internet (<https://naylampmechatronics.com/>), los RTC (Real Time Clock) o también conocidos como reloj en Tiempo Real es un dispositivo que permite integrar mediciones de tiempo alimentados por baterías para no perder la sincronización de fecha y hora en la programación y planificación de tiempos en un sistema.

El modelo utilizado fue DS3231RTC para el control y visualización de diferentes datos de fecha y hora en múltiples unidades y formatos.

Conexión de Sensor de RTC con Arduino.

PIN DS3231RTC.

SCL SCL 21

SDA SDA 20

VCC 5V

GND GND

Características técnicas

- Voltaje de alimentación: 3.3V - 5V DC
- RTC de alta precisión DS3231 con oscilador interno
- Exactitud Reloj: 2ppm (partes por millón)

- Memoria EEPROM AT24C32 (4K * 8bit = 32Kbit = 4KByte)
- Salida de onda cuadrada programable
- La batería puede mantener al RTC funcionando por 10 años.

3.3.3 Actuadores

En las figuras 8, 9, 10 y 11 se describen los actuadores del sistema de riego, para ello se presenta una imagen de cada uno de los actuadores, luego una breve descripción de sus características y la forma como reciben señal por parte del microcontrolador.

Modulo Relé

Figura 8. Módulo Relé



Fuente: Tomado de MÓDULO RELAY 2CH 5VDC. Por Naylamp Mechatronics SAC, 2021 (<https://naylampmechatronics.com/drivers/31-modulo-relay-2-canales-5vdc.html>)

Descripción: este dispositivo funciona como un interruptor de control; cuenta con un circuito eléctrico, una bobina y un electroimán que interaccionan para el inicio o el cierre de otros circuitos, a fin de que trabajen de forma autónoma a través de la circulación de voltajes y corrientes altos. (Unit Electronics, 2021)

Relé utilizado. El Relay SRD-05V DC fabricado por Songle, tiene la función de controlar el encendido y apagado de la electroválvula de agua. Al ser energizada la bobina interna agita de forma mecánica y aislada un contacto móvil, éste a su vez cambia de estado a dos contactos, uno de estar normalmente cerrado (NC) a abierto y otro de estar normalmente abierto (NO) a cerrado. Los dos contactos se

comportan como interruptores, en los cuales circulan voltajes y corrientes relativamente altas que son controlado por la bobina con poca energía.

Conexión de RELE.

PIN RELE. Conexión Arduino.

1-VCC VCC

2-GND GND

3- PIN D13

Características técnicas más importantes:

- Voltaje máximo en AC del interruptor: 240V a 7A o 125V a 10A
- Corriente nominal en bobina 71.4mA
- Voltaje máximo en DC del interruptor: 30V a 10A
- Voltaje para activación de bobina: 5 VCD
- Resistencia en bobina $70(\Omega) \pm 10\%$
- Pines: 5
- Expectativa de vida: 100,000 accionamientos
- Temperatura de operación -25 a 70 °C
- Humedad ambiente: 45% ~ 85 % sin condensación

Conexión para energizar a la bobina de control con VCD:

- Pin1: Terminal de bobina
- Pin2: Terminal de bobina

Conexión para ingresar VCA/ VCD y dispositivo a controlar:

- Pin3: Contacto común o central (C)
- Pin4: Contacto normalmente cerrado (NC)
- Pin5: Contacto normalmente abierto (NO)

Electroválvula.

Figura 9. Electroválvula



Fuente: Tomado de ELECTROVALVULA NELSON PARA JARDIN 1" 9002. Por Almacenes Vidri, 2021 (<https://www.vidri.com.sv/producto/49328/electrovalvula-nelson-para-jardin-1-9002.html>)

Descripción: De acuerdo al sitio web del Laboratorio de electrónica y robótica (<http://mecabot-ula.org/>); una electroválvula es un dispositivo mecánico que se activa eléctricamente y que habilita el paso de un fluido en una tubería. Contiene una válvula en interior la cual es activada por medio de un solenoide, que permite abrir o cerrar el flujo de agua.

Electroválvula utilizada. Electroválvula de 24VAC-50/60HZ Signatura 1 PULG; C&D TechNologia en su sitio web (<https://cdtecnologia.net/>), presenta las especificaciones de este dispositivo, el cual posee un solenoide que realiza el funcionamiento de electroimán que al ser energizado se desplaza junto con el diafragma de la válvula y permite el paso del agua. La válvula se mantiene abierta mientras el solenoide está energizado. Cuando no está alimentado, un resorte se encarga de regresar la válvula a su posición de reposo, que en este caso es del tipo normalmente cerrada (NC).

Conexión de electroválvula con Arduino.

PIN Electroválvula.

1-Cable: Una salida de 12VAC del transformador de 24VAC

2-Cable: PIN tipo normalmente cerrada (NC) del relay.

Características técnicas:

- Material: PVC
- Conexión: 1 pulgada, hembra
- Presión de trabajo: 0.7-10 bar

Pantalla LCD

Figura 10. Modelo de pantalla LCD.



Fuente: Tomado de LCD 16x2. Por Herramientas Tecnológicas Profesionales (HETPRO), 2021 (<https://hetpro-store.com/lcd-16x2-blog/>)

Descripción: El sitio web de Herramientas tecnológicas profesionales en su sitio web (<https://hetpro-store.com/>) especifica que una LCD (Liquid Crystal Display), posee un tamaño de 16x2 que hace referencia a que la pantalla de cristal líquido cuenta con dos filas y cada fila tiene la capacidad de presentar 16 caracteres o símbolos alfanuméricos; los cuales se pueden definir en la programación desde Arduino.

Pantalla LCD utilizada. LCD1602A 16x2 la cual contiene dos filas y 16 caracteres fila de dieciséis pines; los primeros catorce de éstos se utilizan para controlar la visualización y los restantes son para la iluminar el fondo. (Del Valle, 2021)

Conexión de LCD con Arduino

Pin LCD1602A

PIN 1-VSS GND

PIN 2-VDD 5V

PIN 3-Vo Potenciómetro
PIN 4-RS D8
PIN 5-RW D7
PIN 6-ED6
PIN 7- D0 N/C N/C
PIN 8- D1 N/C N/C
PIN 9- D2 N/C N/C
PIN 10- D3 N/C N/C
PIN 11 D4D5
PIN 12 D5D4
PIN 13 D6D3
PIN 14 D7D2
PIN 15 A RESISTENCIA +5V
PIN 16 K GND GND

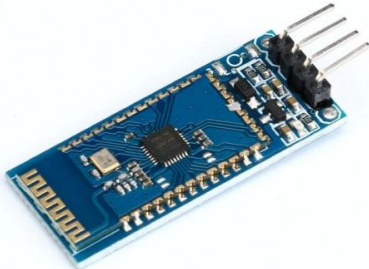
Características técnicas:

- Tipo: Pantalla LCD Monocromática
- Modelo: 16×2 “2 filas y 16 caracteres”
- Controlador: HD44780
- Color: Fondo azul y texto blanco
- Modo de operación: 4 y 8 bits
- Voltaje de alimentación: 5V DC
- Corriente máximo: 25mA

Método de comunicación

Modulo Bluetooth

Figura 11. Modulo Bluetooth.



Fuente: Tomado de Módulo Bluetooth HC-06. Por BricoGeek, (<https://tienda.bricogeek.com/modulos-bluetooth/1351-modulo-bluetooth-hc-06.html>)

El módulo se utiliza para establecer contacto entre la placa Arduino y un móvil que tiene instalada la aplicación de monitoreo; se utilizó el bluetooth HC-06.

Descripción: Este módulo permite agregar conectividad inalámbrica a través de una interfaz serial TTL entre Microcontroladores (PIC, Arduino) y otros dispositivos como PC, laptops o un Smartphone. El módulo Bluetooth HC-06 viene configurado de fábrica para trabajar como esclavo, es decir, preparado para escuchar peticiones de conexión.

Características técnicas:

Firmware BT SPP V3.0.

Voltaje de Operación: 5 V.

Corriente de Operación: < 40 mA

Corriente modo sleep: < 1 mA

Chip: BC417143

Alcance 10 metros

Velocidad de transmisión: 1200 bps hasta 1.3 Mbps

Baudrate por defecto: 9600,8,1, n.

Bluetooth: V2.0+EDR

Longitud de cable: 21.5 cm

Frecuencia: Banda ISM de 2,4 GHz

Modulación: GFSK (Gaussian Frequency Shift Keying)

Potencia de emisión: 4 dBm, clase 2

Sensibilidad: -84 dBm a 0.1% VER

Velocidad asíncrona: 2.1 Mbps (máx.) / 160 kbps.

Velocidad síncronos: 1 Mbps/1 Mbps

Seguridad: Autenticación y encriptación

Interfaz: Bluetooth - Puerto serie UART TTL

3.4 Calibración de sensores

El desarrollo de este procedimiento fue muy importante ya que es una especie de configuración del sensor de humedad de suelo previo a ponerlo en funcionamiento; la forma de realizar este procedimiento fue recrear tres escenarios de humedad: tierra humedad, tierra seca y exposición del sensor al aire, el objetivo fue determinar los valores máximos y mínimos con base en el entorno de prueba; luego estos datos obtenidos se incluyeron en la programación a fin de tener mayor control del comportamiento de dichos sensores; los resultados de las pruebas realizadas se muestran en la siguiente sección.

Los sensores de humedad de suelo utilizados tienen dos tipos de salida, una analógica que cuantifica el valor de la humedad; y otra digital que da una descripción cualitativa especificando presencia o ausencia de humedad, es por ello que en este estudio se tomaron las unidades de salida analógica (usa) para la realización de las pruebas.

Sin embargo, también basado en los resultados de las pruebas de sensores (Tabla 4), se detallan datos de equivalencia entre estas unidades de salida analógicas de dichos sensores y las magnitudes físicas relacionadas.

Tabla 4. *Equivalencia entre valor de sensor de humedad y otras unidades.*

Unidades	Húmedo					Seco	
mA	35	28	21	14	10.5	7	0
VoltiosCC	5	4	3	2	1.28	1	0
Equivalente	0	203	408	613	750	818	1023

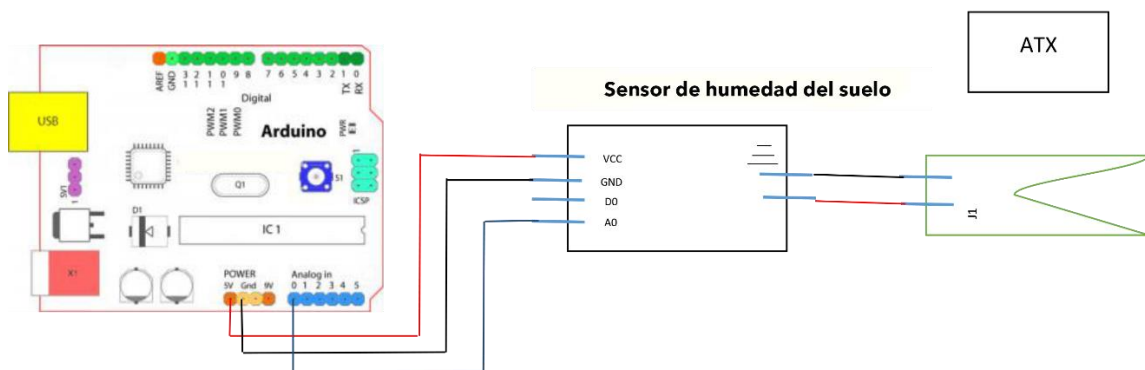
Fuente: Elaboración propia.

3.5 Procesos de conexión y de programación de componentes, placa Arduino y aplicación móvil.

Conexión de sensores

Dos sensores de humedad del suelo se instalaron considerando la distribución entre surcos y el área total de la parcela demostrativa, el primero a ocho metros de la tubería principal y el segundo a 15 metros; lo cual se hizo con el objetivo de conocer los valores de humedad en diferentes puntos de la parcela. Los datos de estos sensores se promediaron para determinar el parámetro que permita el proceso de encendido y apagado de la electroválvula y por consiguiente la activación y desactivación de riego. La figura 12, describe como se conectan estos sensores al microcontrolador.

Figura 12. *Diagrama de conexión de sensores.*

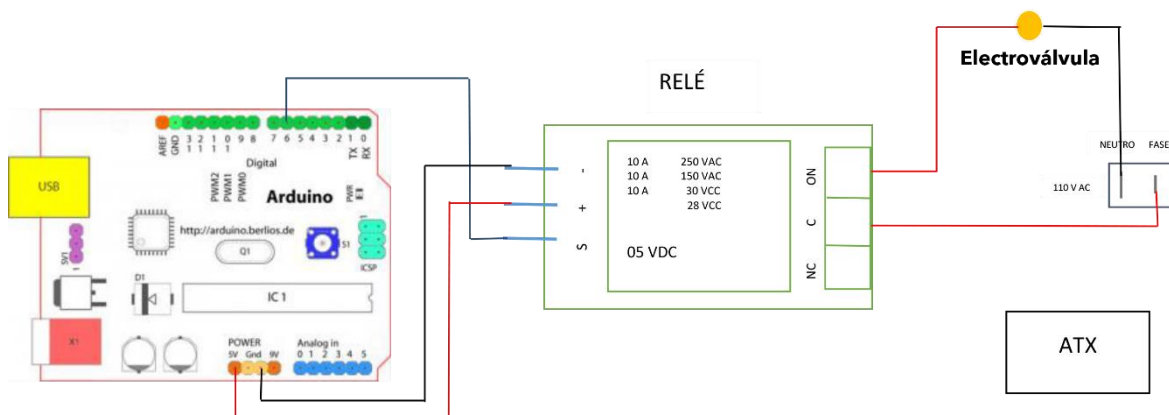


Fuente: Elaboración propia.

Conexión y control electroválvula

La electroválvula es un elemento fundamental en la automatización del riego y para ponerla en funcionamiento y tener control de la apertura y cierre de esta se requiere de la conjunción de dos elementos: 1) Un módulo de relé que funciona con 5v, que al recibir la señal del microcontrolador realiza la función de interruptor de encendido y apagado de la electroválvula; 2) Un transformador de voltaje de 110 a 12 voltios, que convierte la energía requerida para conectar el relé y la electroválvula. Todo este proceso de conexión para el funcionamiento de la electroválvula se describe en la figura 13.

Figura 13. Diagrama de conexión de electroválvula con relé y la placa Arduino.



Fuente: Elaboración propia.

Conexión módulo de comunicación Bluetooth

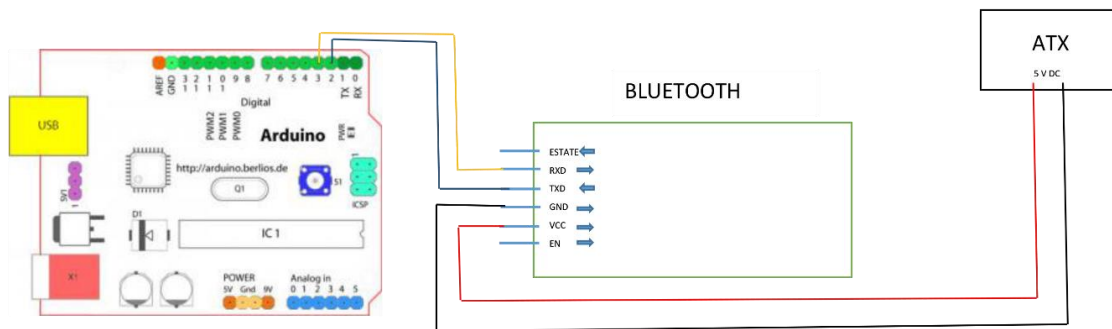
Para establecer comunicación entre la placa Arduino y la aplicación móvil cuyo objetivo es el monitoreo del comportamiento de los sensores de humedad del suelo, se hizo uso de un módulo Bluetooth, la figura 14 muestra el diagrama de conexión de este módulo.

Diseño integrado del prototipo del sistema riego.

Finalmente, se realizó la integración de todos los elementos que componen el prototipo de riego automatizado, tanto los módulos antes descritos conectados al microcontrolador o placa Arduino, lo cual conforma lo que se conoce en el prototipo

de diseño electrónico como la unidad de control, que se describió con mayor detalle en los diagramas previos. Este diseño incluye la colocación de los sensores de humedad de suelo cuyo cable de conexión fue protegido con coraza metálica. Se colocaron dos sensores, el primero a una distancia de 8 metros de la unidad de control y el segundo a 15 metros.

Figura 14. Conexión y adaptación de la electroválvula con relé y la placa Arduino.



Fuente: Elaboración propia.

Para los canales de riego que forman parte del diseño hidráulico, se utilizó PVC de media pulgada para el montaje de la tubería madre, que recorre desde la fuente de agua pasando por la electroválvula hasta las tuberías secundarias o de riego ubicadas entre los surcos de los cultivos y que, a diferencia de la anterior, estas son de poliducto de media pulgada. La distancia de separación entre tuberías de riego es de 70 centímetros; para efectos de irrigar por goteo al cultivo, se perforó cada tubería secundaria con una separación de 60 centímetros entre cada horadación.

El cultivo seleccionado para sembrar en la parcela demostrativa fue maíz a continuación se presentan las especificaciones de las variedades sembradas:

Maíz criollo

Las características de esta variedad fueron tomadas del estudio de Flores et al., (2018, pp 11-13); entre las cuales podemos mencionar:

Altura de la planta: 2.5 m en promedio

Diámetro de mazorca: 4.7 cm en promedio

Número de hileras: 16 hileras

Potencial de rendimiento: 56.89 qq/mz

CENTA H-59

Las especificaciones que se presentan de esta variedad híbrida se tomaron en una guía para la siembra de maíz desarrollada por el Centro Nacional de Tecnología Agropecuaria y Forestal “Enrique Álvarez Córdova” ([CENTA], s.f., p. 13).

Híbrido Triple

Ciclo vegetativo 110-115 días

Días a floración 55

Altura de planta (cm) 247

Altura de mazorca (cm) 135

Reacción al acame Tolerante

Aspecto de tallo Vigoroso

No. Hileras/mazorca 14

Color/tipo de grano Blanco semi dentado

Potencial de rendimiento (qq/mz) 95-100

Deklab Monsanto DK – 357

Esta variedad tiene las siguientes características según el sitio oficial de la productora Deklab (<https://www.dekalb.com.mx/es-mx/productos-dekalb-/maiz/product-detail-template.html/dk-357-noreste.html>).

Días a floración:65-70

Días a cosecha: 155-160

Altura de planta: 250-260 cm

Altura de inserción de mazorca: 110 cm

Cobertura de mazorca: muy buena

Tipo de grano: semi-dentado

Hileras por mazorca: 18-20

Tolerancia a enfermedades acame: tolerante, pudrición de tallo: muy tolerante

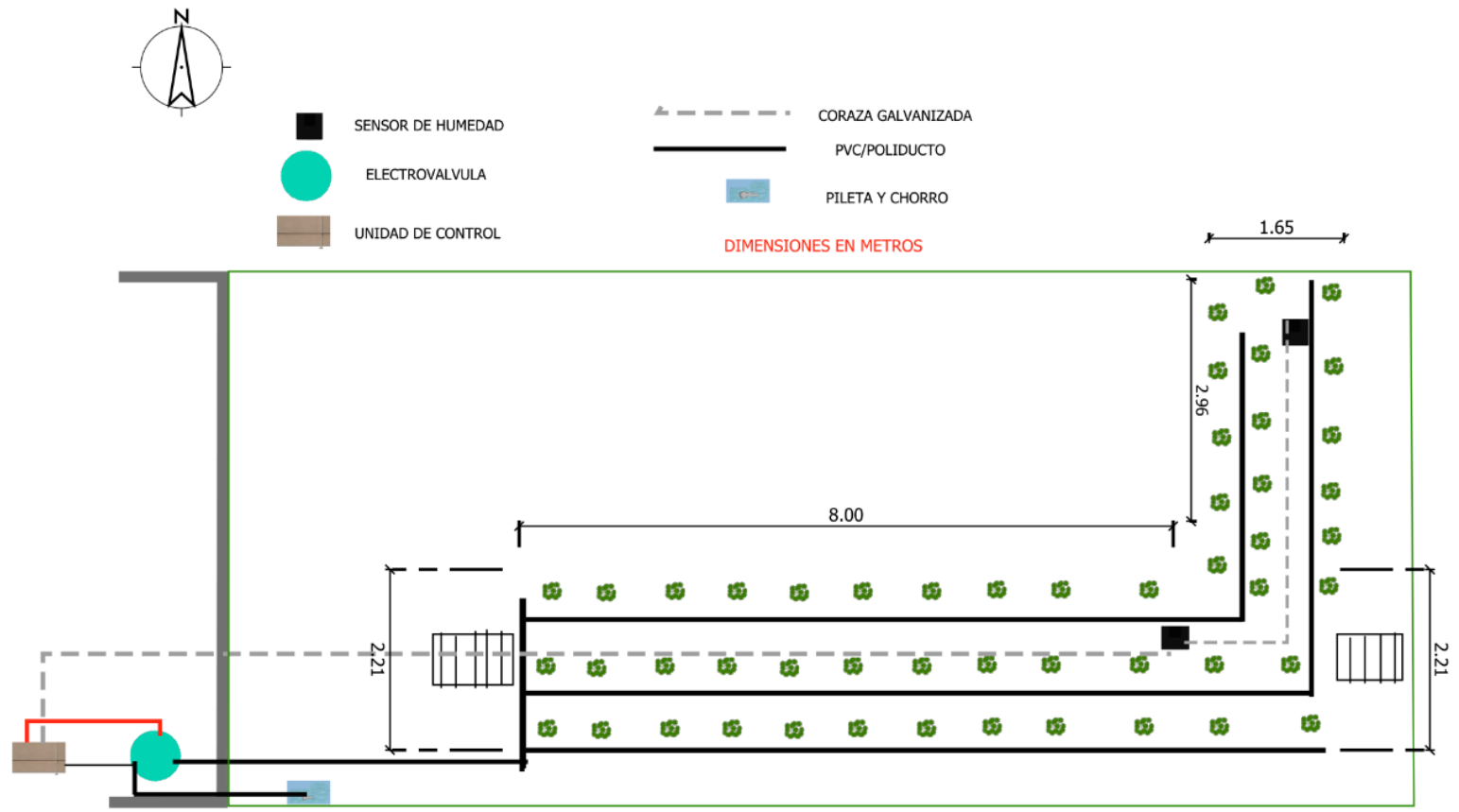
Densidad de siembra (semillas/hectárea): surco convencional: 55,000-60,000

Ciclo intermedio tardío

Todas las variantes antes descritas se sembraron con una distancia entre surcos de 74 centímetros y a una distancia de 80 centímetros entre cada planta.

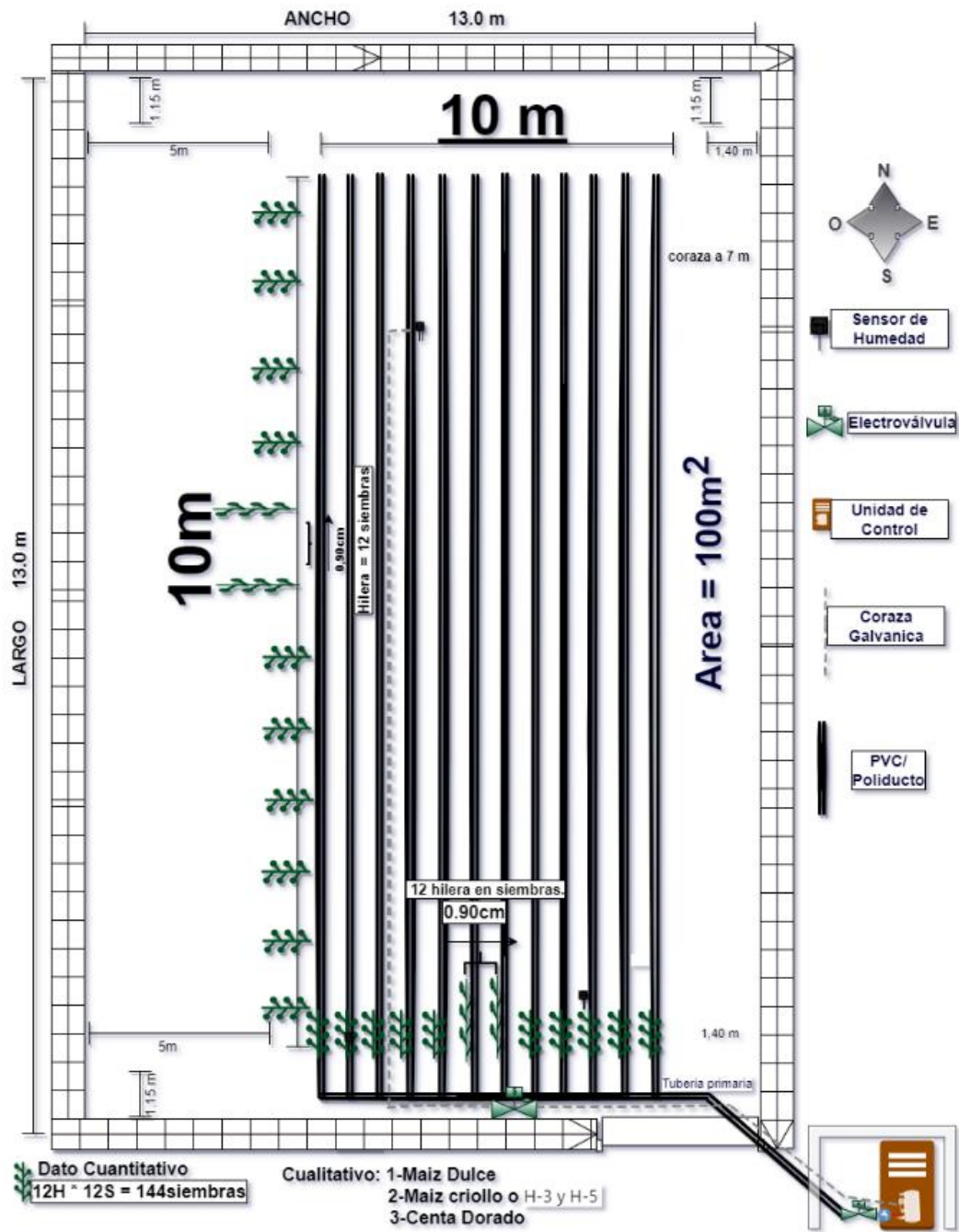
Las figuras 15 y 16 muestran cómo se ubican todos los elementos del prototipo en cada una de las parcelas demostrativas; de igual forma la figura 17 muestra un diagrama general de la conexión de los elementos.

Figura 15. Diagrama integrado del prototipo del sistema riego San Miguel.



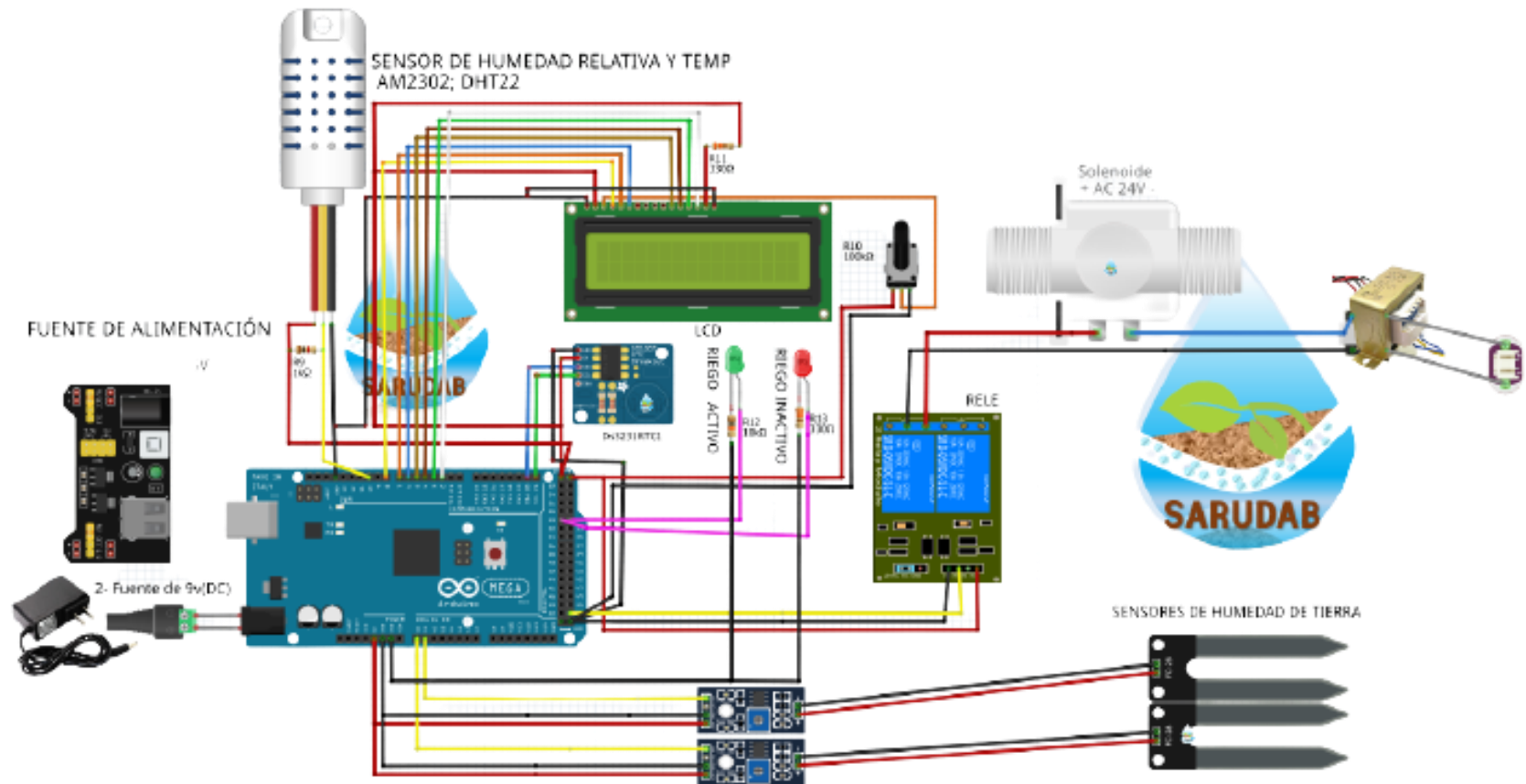
Fuente: Elaboración propia.

Figura 16. Diagrama integrado del prototipo del sistema riego Chalatenango.



Fuente: Elaboración propia.

Figura 17. Diagrama general de conexión de elementos electrónicos del prototipo del sistema.



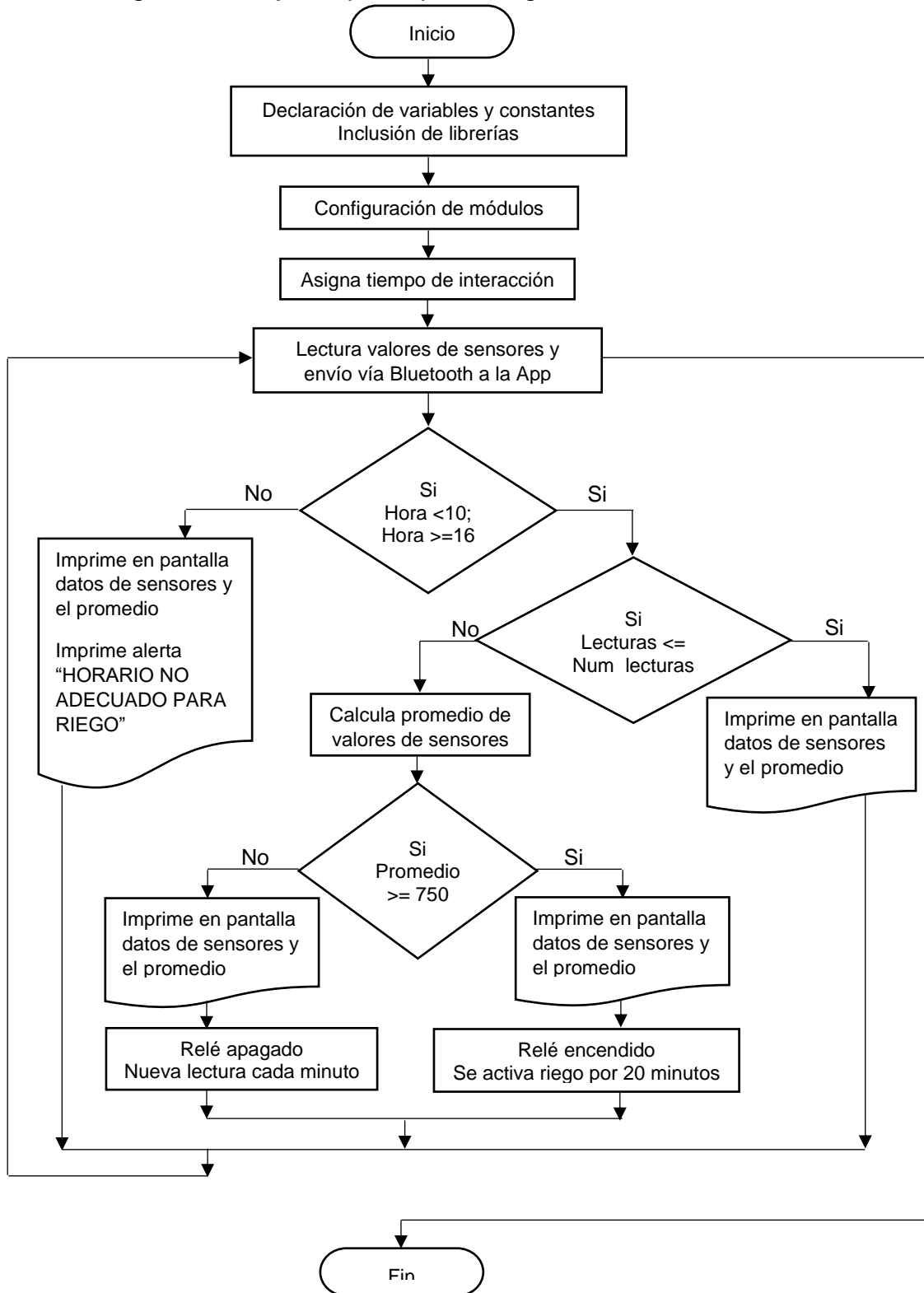
Fuente: Elaboración propia.

3.6 Diagrama de flujo del prototipo de riego

A partir de la funcionalidad descrita para cada módulo, se realizó la programación del microcontrolador, para eso se diseñó un diagrama de flujo basado en el proceso de generar líneas de código en lenguaje Arduino para volver operativos a todos los elementos del prototipo de riego.

Los factores a destacar en el diagrama de flujo son: el monitoreo periódico del valor de humedad del suelo capturado por los sensores, cuyo promedio es el determinante para aplicar riego y que debe ser mayor o igual a 750 (usa) para encender el relé; este dispositivo interactúa con la electroválvula abriéndola en su totalidad lo cual permite aplicar riego durante un período predeterminado de veinte minutos, además de controlar la franja horaria óptima para riego, estableciendo evitar la activación del riego entre las 10:00 y las 16:00 horas, considerando que a esas horas la evaporación es mayor. La figura 18 muestra de manera sintetizada todo el flujo de trabajo e información del prototipo de riego.

Figura 18. Diagrama de flujo del prototipo de riego.



Fuente: Elaboración propia.

4. Resultados

4.1 Características generales del prototipo de riego automatizado.

El método de riego implementado en las dos parcelas demostrativas es por goteo y la fuente de hídrica es agua potable de tubería cuya presión no es controlada por el sistema; sin embargo, de acuerdo a las pruebas realizadas, tiene la capacidad de humedecer en su totalidad el área cultivada, a pesar de que el flujo laminar tiene cambios abruptos debido a la colocación de codos en el ramal, pero estos cambios no son perceptibles ya que se constató que es poco el tiempo que tarda en irrigarse la zonas más alejadas de la fuente hídrica.

Los ramales de riego, también llamados de conducción, son de material PVC de media pulgada para la tubería transversal, la cual conforma la tubería madre del sistema y de poliducto de igual diámetro para cada línea de riego; respecto a lo que se conoce como cabeza de despacho, se generó el goteo con una pequeña perforación equidistante de 60 centímetros en los ramales de riego.

La unidad de control del prototipo está compuesta por una serie de circuitos basados en Arduino y la programación respectiva en este lenguaje con el cual se controla incluso el tiempo de riego; esto es muy importante ya que al no ser constante la irrigación se evita que haya anegamiento, condición que puede dañar el cultivo.

En ambas parcelas solamente fue necesario el uso de una electroválvula, considerando que el riego es localizado, ya que se aplica para humedecer solo la parte del suelo donde está cultivado, condición que posibilita el ahorro de agua.

Las figuras 19 y 20 muestran una vista de las parcelas demostrativas de San Miguel y de Chalatenango con el cultivo de maíz irrigado por el prototipo instalado.

Figura 19. Parcela de San Miguel con cultivo.



Fuente: Elaboración propia.

Figura 20. Parcela de Chalatenango con cultivo.



Fuente: Elaboración propia.

4.2 Resultados específicos del prototipo de riego automatizado San Miguel

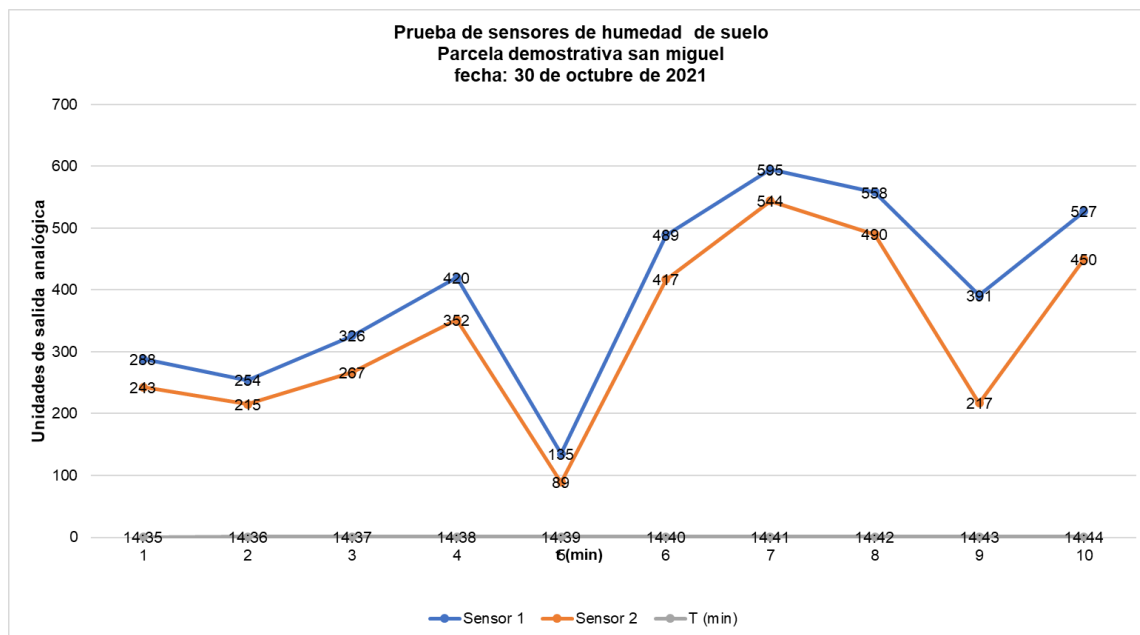
4.2.1 Resultados de pruebas del sensor de humedad del suelo.

Previo a la instalación de los sensores en la parcela demostrativa se hicieron pruebas preliminares, para ello se utilizó una maceta y en la tierra de esta se insertó el sensor de humedad a fin de conocer los valores en diferentes momentos con tierra seca y tierra humedad y la variación de estos respecto al tiempo; luego de un tiempo también se monitorearon los sensores ya instalados en la parcela.

Las siguientes figuras muestran los valores de dos sensores en diferente tiempo para un intervalo de diez minutos, en los cuales se monitoreó la humedad del suelo a través de dichos sensores y medida en unidades de salida analógica, cuando el sensor está en presencia de mucha humedad o en agua registra un valor de 0, cuando está en un ambiente totalmente seco indica un valor de 1023.

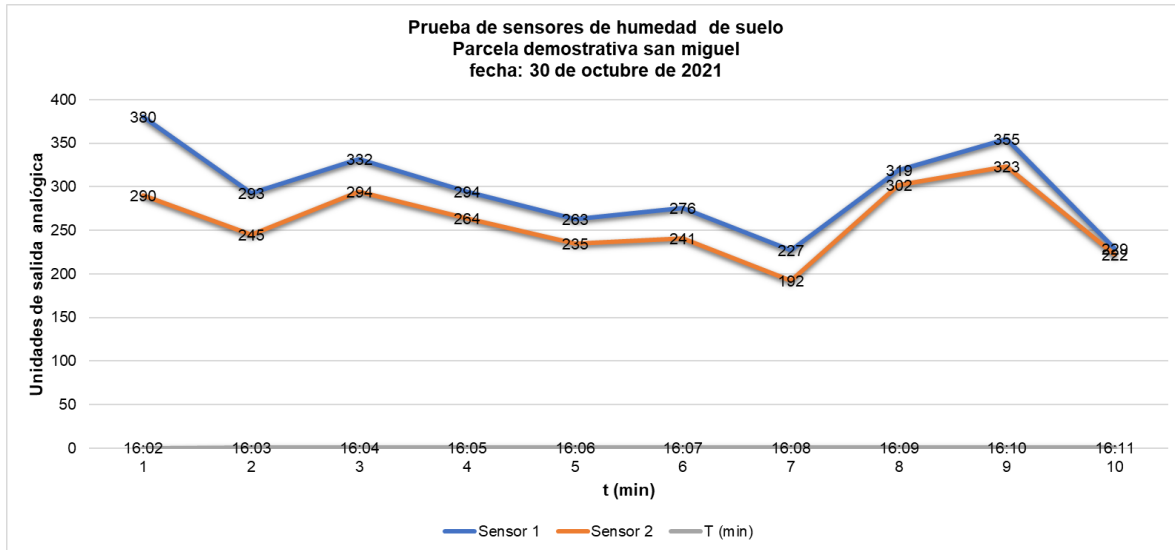
A continuación, las figuras 21 y 22 muestran algunos resultados representativos de las pruebas realizadas en horas diferentes.

Figura 21. Datos de prueba 1 de sensores de humedad de suelo



Fuente: Elaboración propia.

Figura 22. Datos de prueba 2 de sensores de humedad de suelo.



Fuente: Elaboración propia.

Todas las pruebas de los sensores presentadas fueron de utilidad para calibrar estos dispositivos; y de acuerdo a los resultados plasmados en las figuras 21 y 22 se observa una diferencia entre los valores de cada uno de los sensores tomados en el mismo tiempo, esto demuestra la variación de humedad de suelo en los diferentes sectores de la parcela demostrativa; pero también se observa que los dos sensores tuvieron la misma tendencia durante todo el periodo de evaluación lo cual indica el buen funcionamiento del sensor.

4.2.2 Implementación del prototipo de riego

Luego de tener todas las especificaciones del prototipo de riego, en el caso de San Miguel se procedió a la instalación de los elementos físicos en una parcela demostrativa de 22.6 metros cuadrados m^2 de la siguiente manera:

Instalación de tubería de riego:

Considerando que el área es pequeña y para efectos de aprovechar al máximo el espacio se instalaron tres tuberías de riego perforadas para hacer el riego lo más próximo a la planta; tuberías que fueron adaptadas a una tubería madre de PVC de media pulgada; el objetivo de esta distribución de tuberías fue garantizar que el agua llegue a toda la zona cultivada.

Figura 23. Instalación de tubería parcela San Miguel.



Fuente: Elaboración propia.

Instalación de Electroválvula:

Luego de instalado el ramal de irrigación, se conectó la tubería madre a una electroválvula y este dispositivo a otra tubería que conduce directamente a la fuente de agua del prototipo. Los cables que van de la electroválvula al transformador de voltaje y de éste a la unidad de control se protegieron con coraza metálica.

Figura 24. Instalación de electroválvula, parcela San Miguel.



Fuente: Elaboración propia.

Instalación de sensores:

Posteriormente se procedió a la instalación en campo de los sensores de humedad del suelo, para el caso de la parcela demostrativa de San Miguel fueron dos. Los cables que van del sensor al microcontrolador fueron recubiertos también con coraza metálica como medida de protección.

Figura 25. Instalación de sensor, parcela San Miguel.



Fuente: Elaboración propia.

Instalación de unidad de control:

Finalmente, se instaló la unidad de control del prototipo de riego, la cual se montó en una caja de madera que contiene en su interior el microcontrolador (placa Arduino Uno), los circuitos de conexión de los módulos relé, Bluetooth y los sensores de humedad de suelo.

Figura 26. Instalación de unidad de control, parcela San Miguel.



Fuente: Elaboración propia.

4.2.3 Código de programa utilizado en prototipo de riego San Miguel.

```
//Sensor de humedad de suelo////  
  
///Librerías para CRT reloj////////////////////////////////////  
  
#include <Wire.h>  
  
#include "Sodaq_DS3231.h"  
  
char DiaSemana[][4]={"Dom", "Lun", "Mar", "Mie", "Jue", "Vie", "Sab"};  
  
//La línea abajo fija la fecha, hora y día de la semana, se debe suprimir la línea en  
la segunda carga  
  
//Ejemplo 2021 noviembre 17, 11:00:00 día 3-Miercoles  
(0=Dom,1=Lun,2=Mar,3=Mie,4=Jue,5=Vie,6=Sab)
```

```

//DateTime dt(2021,11,23,11,15,0,1);// (anno, mes, numero del dia, hora,
minuto,segundo, dia)

////////////////////////////////////

#include <EduIntro.h>
#include <TimeLib.h>
#include <SoftwareSerial.h>
#include <String.h>

///Parametros para Bluetooth
SoftwareSerial blue(2,3);//Crea conexiones al bluetooth - PIN 2 A TX y PIN 3 A RX
byte dato;

char NOMBRE[21] = "BLUE_UNAB"; //Nombre de 20 caracteres maximo
char BPS      = '4';// 1=1200, 2=2400, 3=4800, 4=9600, 5=19200, 6=38400,
7=57600, 8=115200

char PASS[5]  = "1234";

///Fin parametros Bluetooth////////////////////////////////////

int Humedad2;
int Humedad3;
int Relay = 6;
int SensorPin = A0;
int SensorPin2 = A1;
//int Sensor1=digitalRead(6);

const int Tr=10;
const int Ec=9;

//////////para promedio//////////

int numLecturas=10;//20
int total=0;
int total2=0;
int lecturas=0;

```

```

int promedio=0;
int e=0;
void setup() {
  Serial.begin(9600);
  ///Configura RTC reloj //////////////////////////////////////
  rtc.begin();
  Wire.begin();

  //La linea siguiente fija la fecha, hora, y dia de la semana, se debe suprimir la
  linea en la segunda carga.

  //rtc.setDateTime(dt);

  //////////////////////////////////////

  ///Codigo de configuracion de Bluetooth

  //////////////////////////////////////

  blue.begin(9600);//inicialmente la comunicacion es de 9600 baudios (velocidad
  de fabrica)

  //Ultrasonico/////
  pinMode(Ec,INPUT);
  pinMode(Tr,OUTPUT);
  digitalWrite(Tr,LOW);//se inicia el pin con 0
  delayMicroseconds(5);

  //////////////////////////////////////

  pinMode(13,OUTPUT);
  //digitalWrite(13,HIGH);

  //delay(4000);

  digitalWrite(13,LOW);
  delay (1000);
  digitalWrite(13,HIGH);
  digitalWrite(3,LOW);

```



```

blue.print("AT");//Inicializa comando AT
delay(1000);
blue.print("AT+NAME");//configura el nuevo nombre
blue.print(NOMBRE);
delay(1000);
blue.print("AT+BAUD");//Configura la nueva velocidad
blue.print(BPS);
delay(1000);
blue.print("AT+PIN");//Configura el nuevo PIN
blue.print(PASS);
delay(1000);
///Fin codigo de configuracion de Bluetooth*/
/*****/
pinMode(Relay, OUTPUT); //Configura 6 como la salida OUTPUT
}
void loop() {
////////////////////////////////////
DateTime now = rtc.now();
int hora = now.hour();
int minuto = now.minute();
int segundo = now.second();
////////////////////////////////////
///codigo para Bluetooth y manda datos a la APP
Humedad2 = analogRead(SensorPin);
Humedad3 = analogRead(SensorPin2);
int Humedad_app = map(Humedad2,0, 1023, 100, 0);
int Humedad_app2 = map(Humedad3,0, 1023, 100, 0);
////////////////////////////////////

```

```

blue.println();
if (hora < 10 || hora >= 16) { //inicio del if del rtc//
  blue.println ("HORARIO ADECUADO PARA REGAR ");
  if (lecturas <= numLecturas){
    blue.println("SENSOR 1");
    blue.print("Humedad de suelo: ");
    blue.println(Humedad2);
    blue.print("Porcentaje Humedad de suelo: ");
    blue.print(Humedad_app);
    blue.println("%");
    //////////////////////////////////////
    blue.println(" ");
    blue.println("SENSOR 2");
    blue.print("Humedad de suelo: ");
    blue.println(Humedad3);
    blue.print("Porcentaje Humedad de suelo: ");
    blue.print(Humedad_app2);
    blue.println("%");
    blue.println(" ");
    blue.print("Lecturas ");
    blue.print(lecturas);
    blue.print(" de ");
    blue.println(numLecturas);
    total = (Humedad2 + Humedad3)/2;
    total2 = total2 + total;
    lecturas++;
  }else {
    promedio= total2/numLecturas;
  }
}

```

```

if (promedio >= 650) {
    blue.print("SENSOR 1");
    blue.println();
    blue.print("Requiere regar: ");
    blue.println(Humedad2);
    blue.print("Humedad de suelo: ");
    blue.print(Humedad_app);
    blue.println("%");
    //////////////////////////////////////
    blue.println(" ");
    blue.print("SENSOR 2");
    blue.println();
    blue.print("Requiere regar: ");
    blue.println(Humedad3);
    blue.print("Humedad de suelo: ");
    blue.print(Humedad_app2);
    blue.println("%");
    blue.println(" ");
    digitalWrite(Relay, HIGH);
    blue.println("Relay encendido durante 25 minutos.");
    delay(1500000);//25 minutos
    lecturas=1;
    total=0;
    total2=0;
    promedio=0;
    digitalWrite(Relay, LOW);

} else {

```

```

blue.print("SENSOR 1");
blue.println();
blue.print("Humedad adecuada: ");
blue.println(Humedad2);
blue.print("Humedad de suelo: ");
blue.print(Humedad_app);
blue.println("%");
////////////////////////////////////
blue.println(" ");
blue.print("SENSOR 2");
blue.println();
blue.print("Humedad adecuada: ");
blue.println(Humedad3);
blue.print("Humedad de suelo: ");
blue.print(Humedad_app2);
blue.println("%");
blue.println(" ");
digitalWrite(Relay, LOW);
blue.println("Relay apagado, nuevo diagnostico en 10 minutos.");
for (int e=1; e<11 ;e++){
    delay(60000);///un minuto hace eso 10 veces////////
    blue.print("SENSOR 1");
    blue.println();
    blue.print("Humedad adecuada: ");
    blue.println(Humedad2);
    blue.print("Humedad de suelo: ");
    blue.print(Humedad_app);
    blue.println("%");
}

```

```

////////////////////////////////////
blue.println(" ");
blue.print("SENSOR 2");
blue.println();
blue.print("Humedad adecuada: ");
blue.println(Humedad3);
blue.print("Humedad de suelo: ");
blue.print(Humedad_app2);
blue.println("%");
blue.println(" ");
blue.println("Relay apagado, nuevo diagnostico en 10 minutos.");
blue.print("Minuto: ");
blue.println(e);
}
lecturas=1;
total=0;
total2=0;
promedio=0;
e=0;
}///Fin de lf 2///
}///Fin de lf///
}else {
blue.println ("HORARIO NO ADECUADO PARA REGAR ");
blue.println ("Entre las 10:00 a 16:00 horas");
blue.println("SENSOR 1");
blue.print("Humedad de suelo: ");
blue.println(Humedad2);
blue.print("Porcentaje Humedad de suelo: ");

```

```

    blue.print(Humedad_app);
    blue.println("%");
    //////////////////////////////////////
    blue.println(" ");
    blue.println("SENSOR 2");
    blue.print("Humedad de suelo: ");
    blue.println(Humedad3);
    blue.print("Porcentaje Humedad de suelo: ");
    blue.print(Humedad_app2);
    blue.println("%");
    blue.println(" ");
}////////////////////////////////fin del if RTC

blue.println();

////////////////////////////////////
////////////////////////////////////

digitalWrite(13, !digitalRead(13)); //Cuando termina de configurar el Bluetooth
queda encendido el LED 13

delay(1000);

///Fin codigo Bluetooth////////////////////////////////

delay(60000);//1 minuto

}

```

4.3 Resultados específicos del prototipo de riego automatizado Chalatenango

4.3.1 Resultados de pruebas del sensor de humedad del suelo.

Para la activación de un riego se realizaron pruebas de campo en terreno con el fin de definir una aceptable calibración de los sensores de humedad del suelo sobre lecturas seleccionadas. Estas fueron hechas mediante el entorno de desarrollo de Arduino en un periodo de tiempo de cinco días, a fin de que el sistema tome valores de señal de lectura con mayor precisión. Las pruebas se realizaron en tres escenarios diferentes:

- 1- Pruebas de lectura de ambiente (aire, tierra seca).
- 2- Bajo tierra con nivel de humedad al máximo con una profundidad de 50cm.
- 3- Tierra subhúmeda.

En ese sentido se puede decir que cuando el suelo está:

- **Húmedo:** el valor de salida analógico disminuye
- **Seco:** el valor de salida analógico aumenta

La salida también se puede canalizar a través de una señal digital (D0) la cual es más descriptiva, ya que solo se puede tipificar la humedad como “BAJA” o “ALTA”, dependiendo del contenido de agua; este valor se basa en el umbral que se ajusta a través de un potenciómetro asociado al sensor, en este sentido si la humedad del suelo excede el valor de umbral predefinido, los módulos emiten como salida el término BAJA, de lo contrario, emiten ALTA. Por el contrario, si se utiliza la salida analógica, el valor de la humedad del suelo se cuantifica y se obtendrá un valor entre 0 y 1023 (usa).

Pruebas de lectura de ambiente (aire, tierra seca).

Para el desarrollo de esta prueba se utilizaron dos sensores nuevos, uno ubicado en un área de la parcela de tierra totalmente seca y el otro a temperatura ambiente o al aire libre; este procedimiento de captura de datos fue evaluado desde el controlador de Arduino a través de la conexión analógica de cada sensor y su conexión al circuito principal. Los valores recibidos mostraron la variabilidad o

igualdad en diferente tiempo y fueron presentados en la pantalla, tuvieron por objetivo determinar la condición lógica de activación de la electroválvula para efectuar el riego de la tierra; la tabla 5 y la figura 27 muestran los resultados de esta prueba.

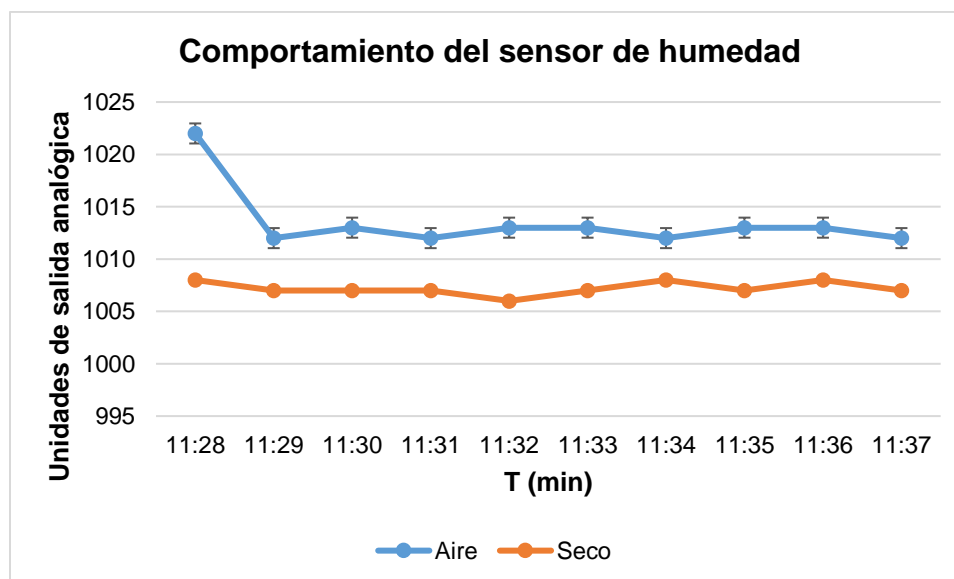
Tabla 5. Datos del monitoreo del sensor de humedad de suelo para calibración

Tiempo (min)	Valor en aire	Valor en suelo seco
11:28	1022	1008
11:29	1012	1007
11:30	1013	1007
11:31	1012	1007
11:32	1013	1006
11:33	1013	1007
11:34	1012	1008
11:35	1013	1007
11:36	1013	1008
11:37	1012	1007

Nota: Valores de sensor están expresados en unidades de salida analógicas (usa).

Fuente: Elaboración propia.

Figura 27. Resultados de prueba de sensor de humedad de suelo en diferente ambiente.



Fuente: Elaboración propia.

La figura 27 muestra el comportamiento del sensor y se observa una tendencia inicial en ambiente (al aire libre) que va de 1022 a 1012 usa; por otra parte, al colocar el sensor en tierra seca su variabilidad fue de 1008 a 1006 usa. Esta prueba se realizó con el fin de identificar su comportamiento de resistividad por lo que se sugiere tomar el valor máximo como dato lógico para la programación del sistema; dicho procedimiento fue llevado a cabo en un período de 10 minutos.

A continuación, en la figura 28, se presentan los resultados del monitoreo de la prueba en el software.

Figura 28. Datos de monitoreo prueba de sensores de humedad en software Arduino.

```

COM6
11:28:12.589 -> Dato de calibración expuesto al ambiente(aire)SH03: 1022 Nivel de humedad %:0.00
11:28:25.647 -> Dato de calibración expuesto al ambiente(tierra seca) SH02: 1008 Nivel de humedad %:0.00
11:28:27.667 -> 3 31.3 88.3 40.7 0.0 0.0
11:28:27.729 -> Dato de calibración expuesto al ambiente(aire)SH03: 1012 Nivel de humedad %:0.00
11:28:40.752 -> Dato de calibración expuesto al ambiente(tierra seca) SH02: 1007 Nivel de humedad %:0.00
11:28:42.814 -> 3 31.3 88.3 41.7 0.0 0.0
11:28:42.814 -> Dato de calibración expuesto al ambiente(aire)SH03: 1013 Nivel de humedad %:0.00
11:28:55.890 -> Dato de calibración expuesto al ambiente(tierra seca) SH02: 1007 Nivel de humedad %:0.00
11:28:57.935 -> 3 31.4 88.5 40.5 0.0 0.0
11:28:57.982 -> Dato de calibración expuesto al ambiente(aire)SH03: 1013 Nivel de humedad %:0.00
11:29:11.014 -> Dato de calibración expuesto al ambiente(tierra seca) SH02: 1007 Nivel de humedad %:0.00
11:29:13.065 -> 3 31.4 88.5 39.6 0.0 0.0
11:29:13.112 -> Dato de calibración expuesto al ambiente(aire)SH03: 1012 Nivel de humedad %:0.00
11:29:26.147 -> Dato de calibración expuesto al ambiente(tierra seca) SH02: 1008 Nivel de humedad %:0.00
11:29:28.195 -> 3 31.4 88.5 39.8 0.0 0.0
11:29:28.195 -> Dato de calibración expuesto al ambiente(aire)SH03: 1012 Nivel de humedad %:0.00
11:29:41.311 -> Dato de calibración expuesto al ambiente(tierra seca) SH02: 1007 Nivel de humedad %:0.00
11:29:43.321 -> 3 31.3 88.3 40.5 0.0 0.0
11:29:43.368 -> Dato de calibración expuesto al ambiente(aire)SH03: 1012 Nivel de humedad %:0.00
11:29:56.413 -> Dato de calibración expuesto al ambiente(tierra seca) SH02: 1007 Nivel de humedad %:0.00
11:29:58.462 -> 3 31.4 88.5 40.5 0.0 0.0
11:29:58.462 -> Dato de calibración expuesto al ambiente(aire)SH03: 1012 Nivel de humedad %:0.00
11:30:11.568 -> Dato de calibración expuesto al ambiente(tierra seca) SH02: 1006 Nivel de humedad %:1.00
11:30:13.573 -> 3 31.3 88.3 41.0 1.0 0.0
11:30:13.620 -> Dato de calibración expuesto al ambiente(aire)SH03: 1013 Nivel de humedad %:0.00
11:30:26.695 -> Dato de calibración expuesto al ambiente(tierra seca) SH02: 1007 Nivel de humedad %:0.00
11:30:28.721 -> 3 31.3 88.3 40.6 0.0 0.0
  
```

Fuente: Elaboración propia.

Prueba con sensores bajo tierra con nivel de humedad al máximo.

Haciendo uso de dos sensores de humedad, uno utilizado en varias ocasiones en tierra mojada y el otro en estado nuevo, se colocaron en un área de la parcela de tierra mojada al máximo; el procedimiento fue evaluado desde el controlador de Arduino a través de la conexión analógica del sensor y su ensamble al circuito principal. Mediante esta prueba se pudo comprobar la variabilidad e igualdad de los valores presentados en pantalla para determinar la condición lógica de inactivación

de la electroválvula de agua. En la tabla 6 y figura 29 se presenta un consolidado de estos resultados para un período de tiempo de diez minutos.

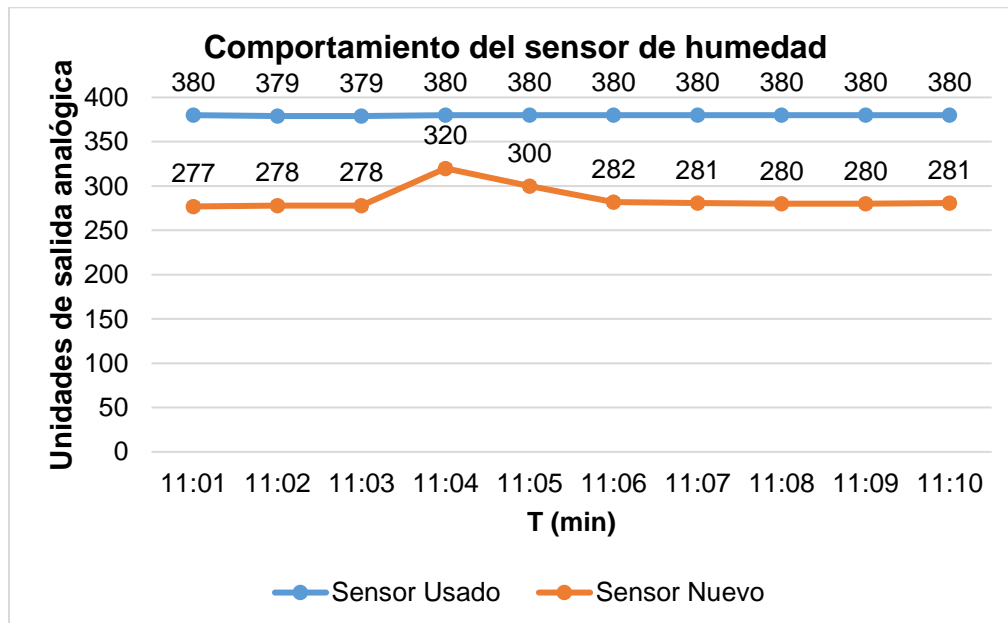
Tabla 6. Datos muestrales de monitoreo de dos sensores de humedad en suelo húmedo.

Tiempo (min)	Sensor Usado	Sensor Nuevo
11:01	380	277
11:02	379	278
11:03	379	278
11:04	380	320
11:05	380	300
11:06	380	282
11:07	380	281
11:08	380	280
11:09	380	280
11:10	380	281

Nota: Valores de sensor están expresados en Unidades de salida analógicas. (usa).

Fuente: Elaboración propia.

Figura 29. Resultados de prueba de dos sensores de humedad de suelo en tierra húmeda.



Fuente: Elaboración propia.

La figura 29 muestra la tendencia de los valores de dos sensores de humedad de suelo ubicados en diferente zona de la parcela; ambos en tierra mojada, a una profundidad de 50 cm, en la cual se aprecia una diferencia en las unidades de salida analógica entre ambos sensores; para el primer sensor que ya había sido usado varias ocasiones en tierra mojada alcanzó un rango de 379 a 380 usa, mientras que el sensor nuevo alcanzó valores de 277 a 320 usa, valores menores que los registrados por el sensor usado. La diferencia podría atribuirse al hecho que el sensor de humedad usado tenía corrosión en sus dos láminas metálicas, por lo que se sugiere tomar el valor mínimo del sensor nuevo como dato lógico para la programación del sistema; este procedimiento fue realizado en un período estimado de 10 minutos. En la figura 30 se puede visualizar los resultados del monitoreo de la prueba en el software Arduino.

Figura 30. Datos de monitoreo de comportamiento de dos sensores de humedad del suelo en suelo húmedo.

```
COM6
11:01:26.085 -> Dato de calibracion bajo tierra SH02: 278 Nivel de humedad %:100.00
11:01:28.118 -> 3 30.6 87.1 42.9 100.0 86.0
11:01:28.118 -> Dato de calibracion bajo tierra SH03: 380 Nivel de humedad %:86.00
11:01:41.215 -> Dato de calibracion bajo tierra SH02: 277 Nivel de humedad %:100.00
11:01:43.207 -> 3 30.7 87.3 43.2 100.0 86.0
11:01:43.207 -> Dato de calibracion bajo tierra SH03: 379 Nivel de humedad %:86.00
11:01:56.263 -> Dato de calibracion bajo tierra SH02: 278 Nivel de humedad %:100.00
11:01:58.294 -> 3 30.7 87.3 43.2 100.0 86.0
11:01:58.341 -> Dato de calibracion bajo tierra SH03: 380 Nivel de humedad %:86.00
11:02:11.372 -> Dato de calibracion bajo tierra SH02: 278 Nivel de humedad %:100.00
11:02:13.399 -> 3 30.7 87.3 43.3 100.0 86.0
11:02:13.399 -> Dato de calibracion bajo tierra SH03: 379 Nivel de humedad %:86.00
11:02:26.489 -> Dato de calibracion bajo tierra SH02: 278 Nivel de humedad %:100.00
11:02:28.476 -> 3 30.7 87.3 43.7 100.0 86.0
11:02:28.523 -> Dato de calibracion bajo tierra SH03: 379 Nivel de humedad %:86.00
11:02:41.545 -> Dato de calibracion bajo tierra SH02: 277 Nivel de humedad %:100.00
11:02:43.595 -> 3 30.7 87.3 44.1 100.0 86.0
11:02:43.595 -> Dato de calibracion bajo tierra SH03: 379 Nivel de humedad %:86.00
11:02:56.660 -> Dato de calibracion bajo tierra SH02: 277 Nivel de humedad %:100.00
11:02:58.660 -> 3 30.8 87.4 44.6 100.0 86.0
11:02:58.707 -> Dato de calibracion bajo tierra SH03: 379 Nivel de humedad %:86.00
11:03:11.733 -> Dato de calibracion bajo tierra SH02: 278 Nivel de humedad %:100.00
11:03:13.749 -> 3 30.8 87.4 43.6 100.0 86.0
11:03:13.796 -> Dato de calibracion bajo tierra SH03: 379 Nivel de humedad %:86.00
11:03:26.848 -> Dato de calibracion bajo tierra SH02: 278 Nivel de humedad %:100.00
11:03:28.844 -> 3 30.8 87.4 43.9 100.0 86.0
11:03:28.890 -> Dato de calibracion bajo tierra SH03: 380 Nivel de humedad %:86.00
11:03:41.943 -> Dato de calibracion bajo tierra SH02: 278 Nivel de humedad %:100.00
```

Fuente: Elaboración propia.

Prueba con sensores en tierra subhúmeda.

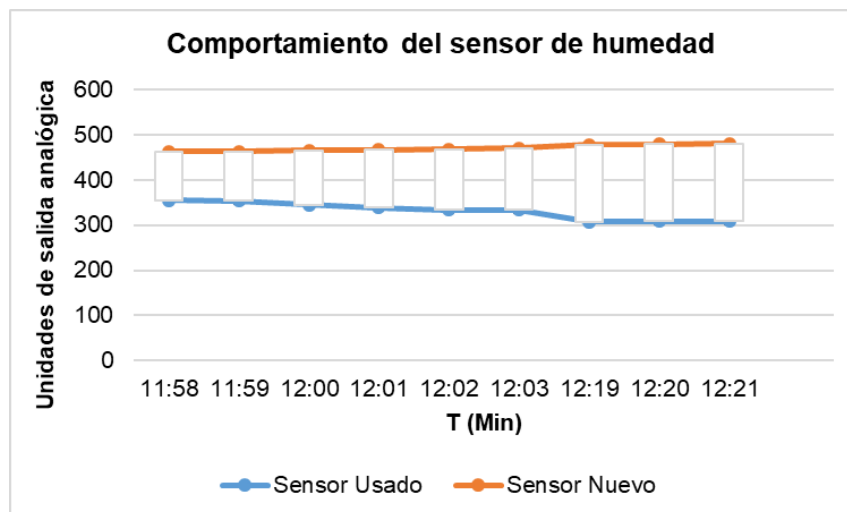
Similar al caso anterior en esta prueba se hizo con dos sensores de humedad, pero en tierra subhúmeda; este procedimiento también fue evaluado desde el controlador de Arduino a través de la conexión analógica del sensor y su ensamble al circuito principal. La tabla 7 y figura 31 muestran los resultados, siempre a un intervalo de medición de diez minutos.

Tabla 7. Datos del monitoreo de dos sensores de humedad en suelo subhúmedo.

Tiempo (min)	Sensor Usado	Sensor Nuevo
11:58	355	462
11:59	354	463
12:00	345	465
12:01	339	466
12:02	334	468
12:03	333	470
12:19	307	478
12:20	309	479
12:21	309	480

Nota: Valores de sensor están expresados en Unidades de salida analógicas. (usa).
Fuente: Elaboración propia.

Figura 31. Resultado de prueba de dos sensores de humedad de suelo en tierra subhúmeda.



Fuente: Elaboración propia.

En la figura 31 se visualizan los valores registrados por los dos sensores colocados en tierra subhúmeda; sin embargo, se volvió a identificar una diferencia de unidades de salida analógica entre ambos sensores; para el caso, el sensor usado presentó una variabilidad aleatoria de los datos; mientras que el sensor nuevo presentó un incremento ascendente de los valores a medida que la humedad va desapareciendo de la tierra hasta llegar al máximo que corresponde a tierra seca. Para efecto de constatar esta situación en la figura 32 se muestran los valores registrados en Arduino de dicha prueba.

Figura 32. Datos de monitoreo de comportamiento de sensores en ambiente semihúmedo.

```

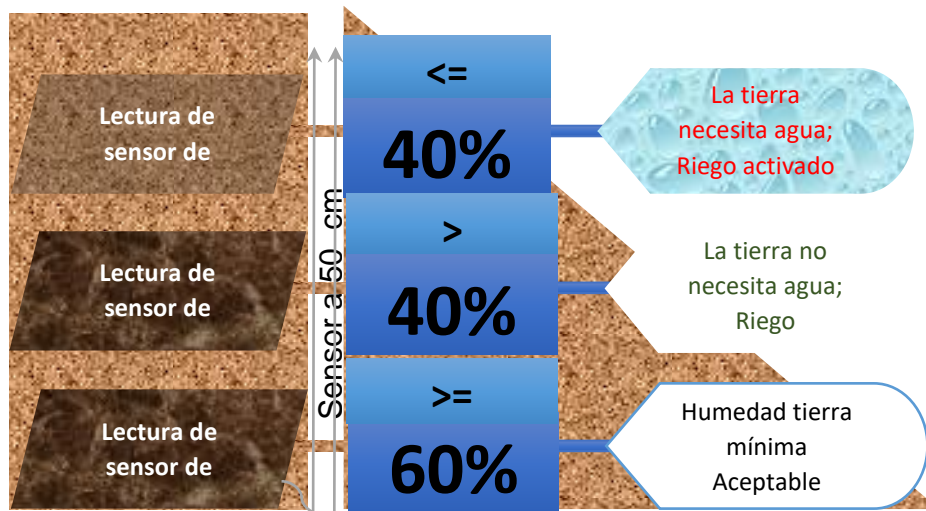
COM6
11:58:44.075 -> Dato de calibración expuesto al ambiente(tierra subhúmedas) SH02: 463 Nivel de humedad %:75.00
11:58:46.104 -> 3 32.1 89.8 37.6 75.0 90.0
11:58:46.151 -> Dato de calibración expuesto al ambiente(tierra subhúmedas)SH03: 355 Nivel de humedad %:90.00
11:58:59.196 -> Dato de calibración expuesto al ambiente(tierra subhúmedas) SH02: 462 Nivel de humedad %:75.00
11:59:01.258 -> 3 32.1 89.8 39.3 75.0 90.0
11:59:01.305 -> Dato de calibración expuesto al ambiente(tierra subhúmedas)SH03: 354 Nivel de humedad %:90.00
11:59:14.379 -> Dato de calibración expuesto al ambiente(tierra subhúmedas) SH02: 463 Nivel de humedad %:75.00
11:59:16.423 -> 3 32.1 89.8 39.8 75.0 90.0
11:59:16.423 -> Dato de calibración expuesto al ambiente(tierra subhúmedas)SH03: 351 Nivel de humedad %:90.00
11:59:29.525 -> Dato de calibración expuesto al ambiente(tierra subhúmedas) SH02: 464 Nivel de humedad %:75.00
11:59:31.576 -> 3 32.2 90.0 38.3 75.0 90.0
11:59:31.576 -> Dato de calibración expuesto al ambiente(tierra subhúmedas)SH03: 349 Nivel de humedad %:90.00
11:59:44.656 -> Dato de calibración expuesto al ambiente(tierra subhúmedas) SH02: 464 Nivel de humedad %:75.00
11:59:46.708 -> 3 32.2 90.0 37.2 75.0 91.0
11:59:46.747 -> Dato de calibración expuesto al ambiente(tierra subhúmedas)SH03: 347 Nivel de humedad %:91.00
11:59:59.832 -> Dato de calibración expuesto al ambiente(tierra subhúmedas) SH02: 464 Nivel de humedad %:75.00
12:00:01.886 -> 3 32.1 89.8 36.9 75.0 91.0
12:00:01.886 -> Dato de calibración expuesto al ambiente(tierra subhúmedas)SH03: 345 Nivel de humedad %:91.00
12:00:14.998 -> Dato de calibración expuesto al ambiente(tierra subhúmedas) SH02: 465 Nivel de humedad %:75.00
12:00:17.003 -> 3 32.1 89.8 37.5 75.0 91.0
12:00:17.050 -> Dato de calibración expuesto al ambiente(tierra subhúmedas)SH03: 344 Nivel de humedad %:91.00
12:00:30.150 -> Dato de calibración expuesto al ambiente(tierra subhúmedas) SH02: 464 Nivel de humedad %:75.00
12:00:32.189 -> 3 32.1 89.8 38.3 75.0 91.0
12:00:32.189 -> Dato de calibración expuesto al ambiente(tierra subhúmedas)SH03: 342 Nivel de humedad %:91.00
12:00:45.286 -> Dato de calibración expuesto al ambiente(tierra subhúmedas) SH02: 466 Nivel de humedad %:74.00
12:00:47.336 -> 3 32.1 89.8 38.2 74.0 92.0
12:00:47.336 -> Dato de calibración expuesto al ambiente(tierra subhúmedas)SH03: 340 Nivel de humedad %:92.00

```

Fuente: Elaboración propia.

El producto final de todas las pruebas fue la calibración de los sensores de humedad, definiéndose los parámetros de activación y desactivación del riego de la tierra, entendiendo que el porcentaje de humedad tiene una equivalencia en el valor de unidades de salida analógica del sensor a través de la función “Map” del lenguaje Arduino. En la figura 33, se presenta un resumen de la calibración de los sensores con su respectivo porcentaje calculado.

Figura 33. Porcentaje de humedad del suelo para activación de riego.



Fuente: Elaboración propia.

4.3.2 Implementación del prototipo de riego

En la parcela de Chalatenango con dimensiones de 10 m de largo por 10 m de ancho (100 m²) se realizó la instalación *in situ* de los elementos físicos del prototipo del sistema de riego.

Instalación de tubería de riego:

El mismo procedimiento seguido en la parcela de San Miguel se aplicó para la instalación de las ramificaciones de irrigación, que consta de 12 tuberías conectadas a la tubería madre de PVC de media pulgada. A cada tubo se le hicieron perforaciones distanciadas 60 centímetros entre sí.

Instalación de Electroválvula:

Para el funcionamiento de este prototipo también se instaló una electroválvula con conexión a la tubería madre, con el fin de abastecer de agua cada uno de los 12 ramales de riego de la parcela.

Figura 34. Instalación de tubería en parcela de Chalatenango.



Fuente: Elaboración propia.

Figura 35. Instalación de electroválvula en parcela de Chalatenango.



Fuente: Elaboración propia.

Instalación de sensores:

Al igual que el prototipo de San Miguel, se instalaron dos sensores de humedad de suelo en diferentes puntos de la parcela con los cuales se hicieron las pruebas. Considerando que esta parcela tiene dimensiones mayores se dejó lista la conexión para dos sensores más, en caso de ser requeridos en un futuro.

Figura 36. Instalación de sensor en parcela de Chalatenango.

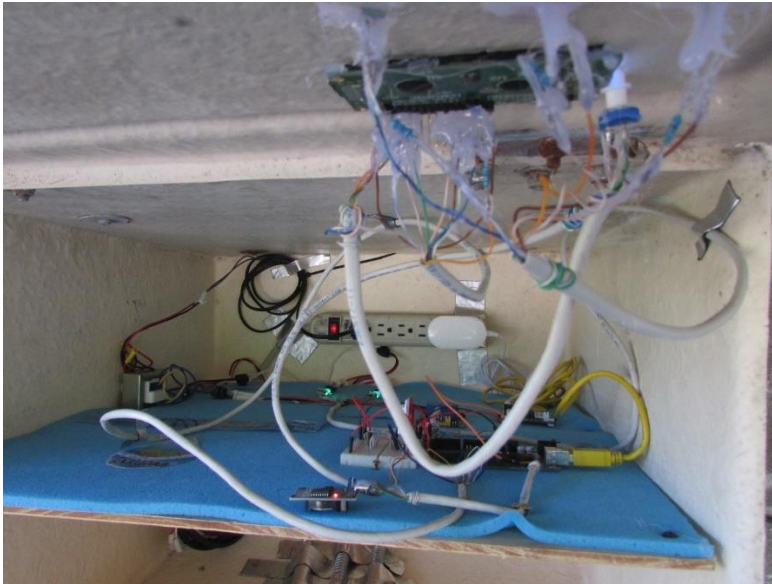


Fuente: Elaboración propia.

Instalación de unidad de control:

La unidad de control del prototipo de riego fue el último elemento instalado, en este caso se empotró en la pared una caja con todos los elementos electrónicos y sus respectivos cables que conforman los circuitos de conexión en su interior.

Figura 37. Instalación de unidad de control en parcela de Chalatenango.



Fuente: Elaboración propia.

Figura 38. Caja de unidad de control en parcela de Chalatenango.



Fuente: Elaboración propia.

4.3.3 Código de programa

```
#include <RTClib.h> // importación de librería para el manejo del módulo RTC
```

```
#include <Wire.h> // importación de librería para interfaz I2C
```

```
#include <LiquidCrystal.h> // importación de librería para el manejo LCD 16*02
```

```

#include "DHT.h"//importación de librería para el sensor DHT22
#include <DHT.h>

RTC_DS3231 rtc;          // definición de objeto del tipo RTC_DS3231
LiquidCrystal lcd(8,7,6,5,4,3,2); //definición de objeto LCD con los pines que se
van a utilizar

int ledValvulaActiva = 30;    // definición de variable de LED verde conectada
al pin 30

int ledValvulaApagada = 31;   // definición de variable de LED rojo conectada
al pin 31

int relayAelectrovalvula = 13; // definición de variable del relay conectada al pin
13

int dht22Pin = 10;           // definición de variable del sensor dht22 conectada al
pin 10

#define DHTTYPE DHT22

DHT dht(dht22Pin, DHTTYPE);

float humedad = 0;          // definición de variables del sensor dht22

float celsius = 0;

float fahrenheit = 0;

const int YSensorPin1S2 = A0; //definición del sensor de humedad SH01
conectada al pin Analogico cero

float ValorMoisFinalS2 = 0;

const int YSensorPin2S3 = A1; //definición del sensor de humedad SH02
conectada al pin Analogico uno

float ValorMoisFinalS3 = 0;

void setup() {

pinMode(ledValvulaActiva, OUTPUT); //definición del modo de salida de led verde

pinMode(ledValvulaApagada, OUTPUT); //definición del modo de salida de led rojo

Serial.begin(9600); // Abrir el puerto serie a la velocidad de 9600bps para
transmisión de datos.

```

```

// Configuración

pinMode(relayAelectrovalvula, OUTPUT); // Definición del pin como salida para
el relay

pinMode(YSensorPin1S2, INPUT); // Definición de entrada sensor humedad de
tierra SH01

pinMode(YSensorPin2S3, INPUT); // Definición de entrada sensor humedad de
tierra SH02

dht.begin(); // INICIALIZAR EL DHT

if (! rtc.begin()) { // si falla la inicialización del modulo

Serial.println("Modulo RTC no encontrado!"); // muestra mensaje de error

while (1); // bucle infinito que detiene ejecución del programa

// rtc.adjust(DateTime(2021, 11, 22, 8, 45, 00)); // función que permite establecer
fecha y horario

//rtc.adjust(DateTime(__DATE__, __TIME__)); // forma de indicar date y time
desde la computadora de forma automática por primera vez, cárgalo la primera
vez y luego en la segunda vez a cargar se debe comentar

}

lcd.begin (16, 2); // inicializar la pantalla con la cantidad de fila y columna que esta
tiene que son 16*2

lcd.print("Sistema de riego"); // Impresión de mensaje de salida por pantalla LCD

delay(1000);

lcd.setCursor(0,1); //columna 1, fila 0

lcd.print(" v.1");

delay(3000);

lcd.clear(); //limpia la pantalla

delay(1000);

}

void loop() {

DateTime fecha = rtc.now(); // función que devuelve fecha y horario en formato

lcd.setCursor(0, 0); //COLOCAR EN POSICION

```

```

lcd.print(fecha.day());    // función que obtiene el día de la fecha completa
lcd.print("/");           // carácter barra como separador
lcd.print(fecha.month()); // función que obtiene el mes de la fecha completa
lcd.print("/");           // carácter barra como separador
lcd.print(fecha.year());  // función que obtiene el año de la fecha completa
lcd.print(" ");           // carácter espacio en blanco como separador
lcd.print(fecha.hour());  // función que obtiene la hora de la fecha completa
lcd.print(":");           // carácter dos puntos como separador
lcd.print(fecha.minute()); // función que obtiene los minutos de la fecha
completa
lcd.print(":");           // carácter dos puntos como separador
lcd.println(fecha.second()); // función que obtiene los segundos de la fecha
completa
delay(1000);              // demora de 1 segundo
float Humd = dht.readHumidity(); // lectura de humedad
float TemC = dht.readTemperature(); //lectura de temperatura C
float TemF = dht.readTemperature(true); // lectura de temperatura F
if(isnan(Humd) || isnan(TemC) || isnan(TemF)) { //condición lógica en donde sí; en
cualquiera de las variables el valor que devuelve no es numero
    lcd.setCursor(0, 1); //COLOCAR EN POSICION
    lcd.print("Hay un error al leer el sensor"); //imprimir en pantalla el mensaje
    lcd.setCursor(0, 1); //COLOCAR EN POSICION
    lcd.print("Revisar conexion");           //imprimir en pantalla el mensaje
}
else{
    lcd.setCursor(0, 1); //COLOCAR EN POSICION
    lcd.print("Tem: "+ String(TemC,1) + "C " + String(TemF,1) +"F ");
    delay(6000);
    lcd.setCursor(0, 1); //COLOCAR EN POSICION

```

```

    lcd.print("Hum:" + String(Humd,1) + "% ");
}
delay(2000); // 2 mil milisegundos entre lectura y lectura() esperar 2 segundos
////////////////////////////////////
int Ndata = 3;
//creacion de string con el formato Ndata
String str_Payload;
str_Payload += Ndata;
str_Payload += " " + String(TemC,1);
str_Payload += " " + String(TemF,1);
str_Payload += " " + String(Humd,1);

const int valorAire = 1022;//valor de calibración sensor SH01 máximo obtenido al
ambiente es decir al aire

const int valorAgua = 277;//valor de calibración mínimo SH01 obtenido en
ambiente sumergido en agua con tierra

int ValorSensorhumedadPinS2 = analogRead(YSensorPin1S2);

if (ValorSensorhumedadPinS2 >= valorAgua && ValorSensorhumedadPinS2 <=
valorAire) //condición lógica para comprobar que las dos lecturas se cumplan en
donde el valor de agua sea mayor que 277 y el valor de aire menor o igual 1022

    ValorMoisFinalS2 = map(ValorSensorhumedadPinS2, valorAire, valorAgua, 0,
100);//mapear el resultado del pin analógico, entre mas cercano se este valor del
aire sera 0, y si más cerca se este del valor de agua será el 100%

    str_Payload += " " + String(ValorMoisFinalS2,1);
}
else
{
    if (ValorSensorhumedadPinS2 > valorAire )
    {
        ValorMoisFinalS2 = 0;
str_Payload += " " + String(ValorMoisFinalS2,1);

```

```

}
else if(ValorSensorhumedadPinS2 < valorAgua)
    ValorMoisFinalS2 = 100;
str_Payload += " " + String(ValorMoisFinalS2,1);
}
lcd.setCursor(0,0);
lcd.print("SensorHY-1:"); //sensor humedad
lcd.print(ValorMoisFinalS2,1);
lcd.print("% ");
delay(2000);
const int valorAireY2 = 1022;//valor máximo SH02 expuesto al ambiente aire
const int valorAguaY2 = 277;//valor mínimo SH02 cuando el sensor está
sumergido en agua más tierra
int ValorSensorhumedadPinS3 = analogRead(YSensorPin2S3);
if (ValorSensorhumedadPinS3 >= valorAguaY2 && ValorSensorhumedadPinS3 <=
valorAireY2)
{
    ValorMoisFinalS3 = map(ValorSensorhumedadPinS3, valorAireY2, valorAguaY2,
0, 100);//mapear el resultado del pin analógico, entre más cercano se este valor
del aire sera 0, y si más cerca se este del valor de agua será el 100%
    str_Payload += " " + String(ValorMoisFinalS3,1);
}
else
{
    if (ValorSensorhumedadPinS3 > valorAireY2 )
    {
        ValorMoisFinalS3 = 0;
        str_Payload += " " + String(ValorMoisFinalS3,1); //imprimir valor de humedad de
tierra
    }
}
}

```

```

}
else if(ValorSensorhumedadPinS3 < valorAguaY2)
    ValorMoisFinalS3 = 100;
    str_Payload += " " + String(ValorMoisFinalS3,1); //imprimir valor de humedad de
tierra
}
str_Payload += " " ;
str_Payload += "\n";
Serial.print(str_Payload);
lcd.setCursor(0,0);
lcd.print("SensorHY-2:"); //sensor humedad y 01
lcd.print(ValorMoisFinalS3,1);
//Serial.print(ValorMoisFinal);
lcd.print("% ");
//Serial.println("%");
delay(1000);
//////////////////////////////////////*****//////////////////////////////////////

//La siguiente condición lógica corresponde a la configuración de activación y
desactivación de riego según datos de lectura por los sensores de humedad
//El sistema de riego se activa cuando el valor de lectura del sensor de humedad
es menor o igual a 40% y se desactivara cuando exista un valor mayor a 40%
//El sistema de riego muestra en pantalla humedad de tierra mínima aceptable si el
valor de humedad es mayor 0 igual al 60%
//El Sistema de riego muestra en pantalla humedad de tierra máxima si el valor de
humedad es mayor o igual al 80%
//////////////////////////////////////*****//////////////////////////////////////

if((ValorMoisFinalS2<=40)&&(ValorMoisFinalS3<=40))
{
// Código principal donde ocurren en loop

```

```

digitalWrite(relayAelectrovalvula, HIGH); // envía señal alta al relay
//Serial.println("EL riego automático se ha activado");
//delay(300000);
lcd.setCursor(0,1); //columna 1, fila 0
lcd.print("La tierra necesita");
delay(2000);
lcd.setCursor(0,1); //columna 1, fila 0
lcd.print("Agua; Riego activado");
digitalWrite(ledValvulaActiva,HIGH);
digitalWrite(ledValvulaApagada, LOW);//riego inactivo
delay(1000);
}
else if ((ValorMoisFinalS2>=60)&&(ValorMoisFinalS3>=60)){
digitalWrite(relayAelectrovalvula, LOW); // envía señal baja al relay
digitalWrite(ledValvulaActiva,LOW);
digitalWrite(ledValvulaApagada, HIGH);//riego inactivo
lcd.setCursor(0,1); //columna 1, fila 0
lcd.print("Humedad. de tierra ");
delay(1000);
lcd.setCursor(0,1); //columna 1, fila 0
lcd.print("Minima Aceptable");
delay(1000);
} else if ((ValorMoisFinalS2>=80)&&(ValorMoisFinalS3>=80)) {
digitalWrite(relayAelectrovalvula, LOW); // enviar señal baja al relay
digitalWrite(ledValvulaActiva,LOW);
digitalWrite(ledValvulaApagada, HIGH);//riego inactivo
lcd.setCursor(0,1); //columna 1, fila 0
lcd.print("Humedad. de tierra ");

```



```
    delay(2000);
    lcd.setCursor(0,1); //columna 1, fila 0
    lcd.print("    Máxima");
    delay(1000);
}
else {
    digitalWrite(relayAelectrovalvula, LOW); // envía señal baja al relay
    digitalWrite(ledValvulaActiva,LOW);
    digitalWrite(ledValvulaApagada, HIGH);//riego inactivo
    lcd.setCursor(0,1); //columna 1, fila 0
    lcd.print("Riego desactivado");
    delay(2000);
    lcd.setCursor(0,1); //columna 1, fila 0
    lcd.print("No necesita agua");
    delay(1000);
}
}
```

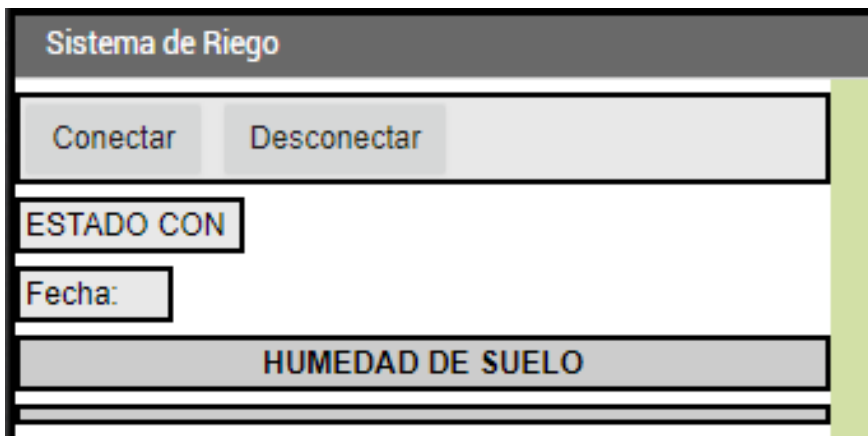
4.4 Aplicación móvil del prototipo de Riego

Como resultado final se desarrolló una aplicación móvil que tiene dos propósitos: 1) Monitorear el comportamiento de los sensores de humedad del suelo, 2) Servir como canal de comunicación entre el prototipo de riego y el usuario.

4.4.1 Diseño de Aplicación Móvil (Interfaz del usuario)

En la figura 39 se presenta el diseño de la pantalla de la aplicación móvil, que se divide en tres partes: en la superior se muestra la conexión a través de Bluetooth a la placa Arduino; al medio se despliega el detalle del estado de conectividad para garantizar al usuario haber realizado ese proceso de forma correcta y la fecha de monitoreo de datos, lo que permite generar un registro histórico; en la parte inferior aparece el detalle de los datos capturados en los sensores de humedad de suelo.

Figura 39. Diseño de pantalla de aplicación móvil.

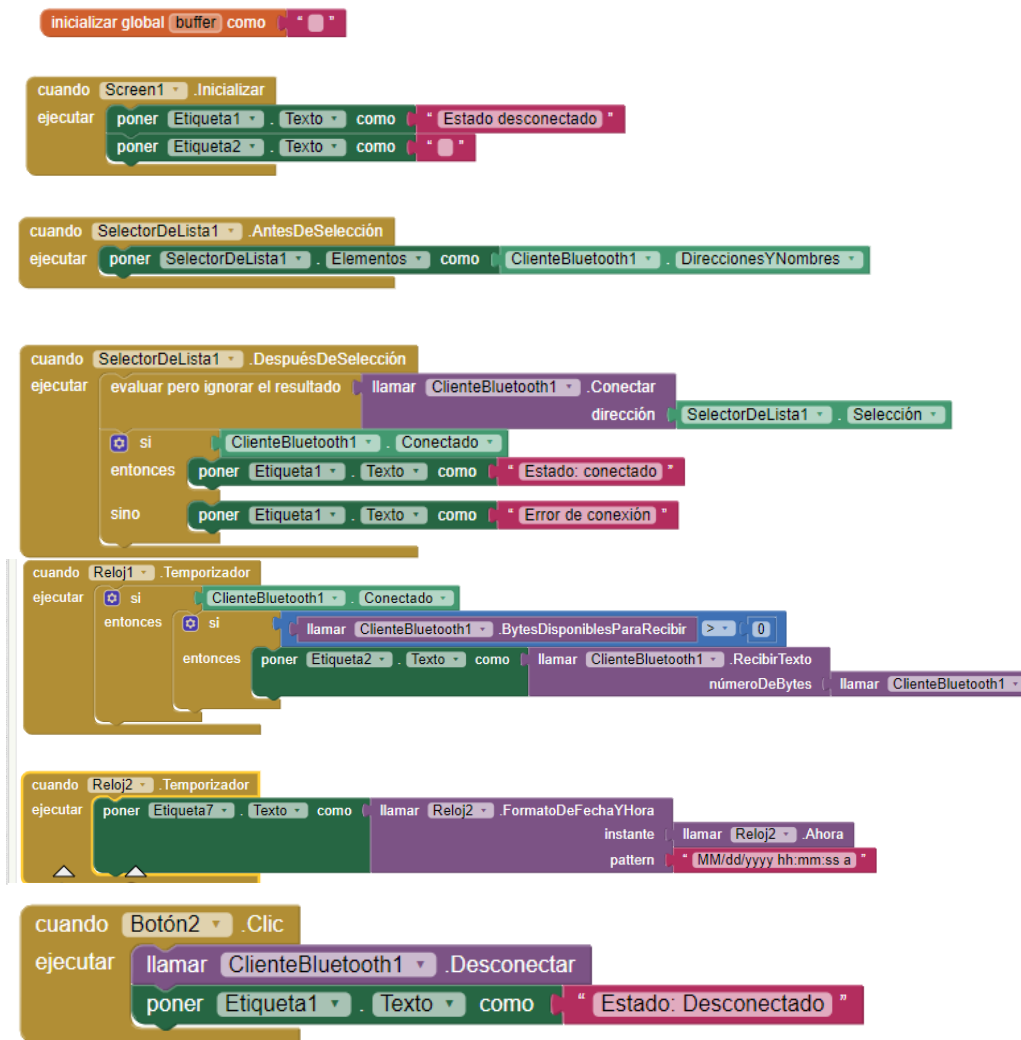


Fuente: Elaboración propia.

4.4.2 Algoritmo y programación de la aplicación móvil

Para pasar del diseño de la pantalla a la programación de la aplicación móvil se hizo uso del entorno de desarrollo App Inventor. En la figura 40 se describe la secuenciación de instrucciones para lograr la funcionalidad de aplicación diseñada.

Figura 40. Codificación de aplicación móvil.



Fuente: Elaboración propia.

Descripción del funcionamiento de la aplicación.

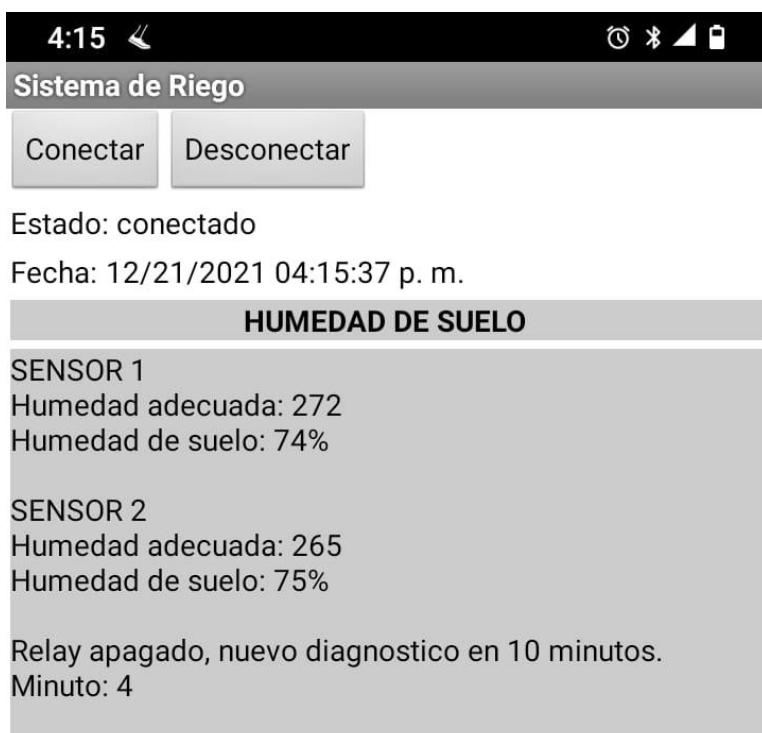
La figura 41 muestra la pantalla o interfaz final de la aplicación, en la parte superior aparece el título “**Sistema de Riego**”, luego tiene dos botones “**Conectar**” y “**Desconectar**”, precisamente sirven para establecer conexión con el microcontrolador del prototipo y, la conectividad se visualiza en la parte de “**Estado**”.

En la siguiente sección “**HUMEDAD DE SUELO**”, se monitorean los valores de cada uno de sensores del prototipo instalados en la parcela; los valores se presentan de

dos formas: como humedad del suelo expresada en unidades de salida analógica (usa) en la opción **Humedad adecuada** o como valor porcentual que es la medida usual de esta variable en la opción **Humedad del suelo**. Para calcular este valor equivalente se hace uso de la función Map de Arduino.

En la parte inferior se muestra el estado del relé; cuando está apagado indica que el valor de humedad registrado no ha activado la electroválvula; cuando está encendido ya se está irrigando la parcela. Se visualiza también el intervalo de tiempo en el cual se vuelve a captar lectura de los sensores de humedad para determinar si se continúa regando o se cierra la electroválvula; además se despliega un contador de minutos, que es importante para reiniciar el diagnóstico, es decir tomar una nueva lectura de los sensores.

Figura 41. Pantalla final de aplicación móvil.



Fuente: Elaboración propia.

5. Discusión

El resultado principal de este estudio fue la implementación de un prototipo de sistema de riego inteligente en dos parcelas demostrativas, ubicadas en los campus de los centros regionales universitarios de San Miguel y Chalatenango. La característica de riego inteligente fue posible alcanzarla mediante el uso de microcontroladores que interactúan con sensores y actuadores, cumpliendo con la premisa que todo sistema domótico está basado en la captación de datos de una variable física o química (humedad del suelo para este estudio) a través de un sensor que convierte el valor captado en energía que se transfiere a un microcontrolador y éste la canaliza como señal a un actuador, en este caso el relé, como lo definen Merino & Guzmán (2015).

Un elemento de mucha importancia para el prototipo desarrollado fue el método de irrigación, el sistema de riego por goteo el seleccionado para instalarse en ambas parcelas, variando únicamente la cantidad de ramales instalados de acuerdo a la extensión del terreno irrigado. Otro aspecto importante es que la fuente de agua para abastecer todos los ramales de riego fue similar en las dos parcelas, teniendo la presión y caudal suficiente para regar todos los surcos de cultivo, por lo que toma menos de un minuto para activar el riego en su totalidad.

La razón principal de la elección del sistema de riego por goteo obedece a las múltiples ventajas que tiene respecto a otros sistemas, tal como se plantea en el sitio web oficial de Ambientum.com, para el caso: 1) El riego se aplica de manera focalizada cerca de cada planta, humedeciendo solo las zonas de interés, 2) Es aplicable a diferentes tipos de cultivo, característica importante en caso de hacer rotación del cultivar en la parcela, 3) Es adaptable a terrenos de diferentes dimensiones. Las tres características se demostraron con éxito en las parcelas descritas en el presente estudio.

La humedad del suelo fue el factor determinante para lograr la automatización del sistema de irrigación de las parcelas, es decir activar y desactivar el riego; si bien, esto coincidió con lo planteado en el estudio de Simón (2018), también hay

diferencias con dicho estudio, específicamente con respecto al tipo de cultivo para ambos prototipos, ya que en ese el riego se aplicó a parcela sembrada con pasto y en el presente caso con maíz.

El prototipo implementado utilizó sensores de humedad de suelo (higrómetro YL69), cuantificada por medio de una salida analógica con que cuenta dicho sensor, esto facilitó la comunicación constante con el microcontrolador Arduino, lo que permitió realizar un monitoreo de la variable de activación para un periodo establecido y activar el riego con mayor precisión. Una debilidad de estos sensores (higrómetro YL69) es que sus terminales se corroen con facilidad por estar construidas con estaño, lo que acorta su vida útil; esto se comprobó con los resultados de las pruebas efectuadas que mostraron disparidad entre las medidas obtenidas por un sensor nuevo comparado con otro que tenga meses de uso. Esta desventaja se solventó sustituyendo los terminales del sensor por otros elaborados con dos barras de cobre y la conexión de ambas barras con alambre Phelps Dodge 12 AWG. La efectividad de la solución se demostró por medio de la similitud de los resultados obtenidos tanto con sensores refaccionados como nuevos.

En cuanto a los actuadores, se utilizó una electroválvula de 110 voltios conectada directamente a la fuente de agua; para su funcionamiento correcto se utilizó un relé como elemento de apertura y cierre de dicho dispositivo, además de un transformador de voltaje, previendo la enorme diferencia con la que trabaja la electroválvula y el relé conectado al microcontrolador.

Todo prototipo de riego inteligente debe poseer una interfaz de comunicación entre el microcontrolador o componente electrónico, el componente hidráulico conformado por la serie de tuberías interconectadas y el usuario para volver operativo todo el sistema. En este estudio se tuvo dos interfaces que utilizaron un módulo Bluetooth como canal de comunicación para el envío de datos a un teléfono móvil y a una pantalla led como dispositivos de salida.

Los prototipos del sistema de riego desarrollados en las parcelas de San Miguel y de Chalatenango tiene similitud en cuanto a: ambos utilizan el mismo modelo de

sensor de humedad, tipo de electroválvula, tipo de tubería, fuente de agua y lenguaje de programación. La disimilitud entre ambos está referida a: el área de cultivo, la cuantificación del parámetro de humedad de suelo para activar el riego, en un caso se utilizaron unidades de salida analógica (usa) y se estableció la activación de riego cuando el promedio de los valores de los sensores alcanza un valor de 750 (San Miguel); en el otro, se hizo una conversión del valor cuantificable a porcentaje de humedad como parámetro en el software y la activación de la irrigación se estableció cuando el promedio de los valores de los sensores fuera menor o igual al 40% (Chalatenango). Esa conversión fue posible mediante el uso de la función "Map" de Arduino. La otra diferencia entre prototipos fue la interfaz de comunicación, en la parcela de San Miguel se utilizó un teléfono móvil y en el caso de Chalatenango se usó una pantalla LED y se monitoreó con una computadora portátil.

Finalmente, se presentan algunas fortalezas y limitaciones del prototipo, que deben considerarse para mejorar la funcionalidad del sistema. Como fortalezas se identificaron:

- Alta adaptabilidad del prototipo considerando al buen funcionamiento verificado en las dos parcelas demostrativas con diferentes condiciones.
- Se cuenta con un algoritmo base en lenguaje Arduino, que puede ser adaptado a diferentes condiciones con solo modificar en el código el parámetro humedad de suelo para cambiar el comportamiento del sistema de riego.
- Fácil manejo de interfaz del usuario ya que la aplicación es interactiva e intuitiva.
- Reducción en el consumo agua, puesto que el riego es aplicado solo en la zona cultivada de las parcelas.

Como limitantes se identificaron:

- La Plataforma de Arduino es funcional para el manejo de sistemas de riego en áreas pequeñas, para parcelas de mayor extensión se debe utilizar otro tipo de tecnología.
- Debió considerarse el estudio del tipo de suelo de las parcelas. Con los resultados obtenidos se podría afinar el parámetro de humedad del suelo para activar el proceso de riego, así como caracterizar mejor el tipo de cultivo a sembrar, de acuerdo a la composición química del suelo.
- Debió considerarse el manejo de plagas de insectos y de otras especies que depredan el cultivo, con el fin de mantenerlo en buenas condiciones durante todas sus etapas fenológicas.

6. Conclusiones

1- Los prototipos de riego inteligente implementados en las parcelas demostrativas de San Miguel y Chalatenango funcionaron satisfactoriamente con base a lo diseñado.

2- La automatización del proceso de riego, a través del control de la variable de humedad de suelo, permitió aplicar riego focalizado en la zona aledaña a la planta, lo cual garantiza el uso óptimo del recurso hídrico.

3- La documentación del algoritmo y del código de programación Arduino permiten la facilidad de modificación de estos, garantizando con ello el buen funcionamiento del prototipo de riego, incluso en casos que haya cambios en el entorno, sean por cambio de cultivo o por el tipo de suelo; de manera que cada prototipo es una herramienta con alto grado de flexibilidad en el diseño y adaptabilidad en la implementación.

4- La aplicación móvil diseñada que interactúa con el prototipo es funcional, y permite al usuario realizar un monitoreo constante del comportamiento de la variable de humedad del suelo, de manera que el riego se active a la hora apropiada y durante el tiempo de riego requerido en cada parcela.

5- El monitoreo continuo durante la fase de prueba del prototipo de riego fue importante para la calibración de los sensores de humedad, dependiendo de diferentes condiciones de suelo (húmedo, semihúmedo y seco). Ese proceso permitió hacer los ajustes necesarios para la puesta en marcha todo el sistema de riego.

6- La integración del prototipo de riego inteligente con la aplicación móvil diseñada evidenció el cumplimiento de los requerimientos de información para el monitoreo del estado de la variable humedad de suelo en tiempo real; por ello la transmisión continua de estos datos y su visualización en un dispositivo móvil permitió la posibilidad de tomar decisiones sobre todo respecto al oportuno mantenimiento preventivo o correctivo de los elementos electrónicos del prototipo.

7. Referencias

- Alises, F. (s.f.). TecnologíaFelixAlises. Tema 3. Sistemas de control. Consultado el 15 de junio de 2021. <https://sites.google.com/site/tecnologiafelixalises/tema-3>
- Almacenes Vidri. (2021). Electroválvula Nelson para jardín 1" 9002. Consultado el 3 de julio de 2021. <https://www.vidri.com.sv/producto/49328/electrovalvula-nelson-para-jardin-1-9002.html>
- Ambientum.com - El portal profesional del medio ambiente. (s.f.). Sistemas de riego. https://www.ambientum.com/enciclopedia_medioambiental/suelos/sistemas_de_riego.asp
- Arduino. (2021). Arduino Mega 2560 Rev3. Consultado el 17 de agosto de 2021. <https://store.arduino.cc/products/arduino-mega-2560-rev3>
- Arduino. (2021). Arduino Uno Rev3. Consultado el 17 de agosto de 2021. <https://store-usa.arduino.cc/collections/boards/products/arduino-uno-rev3>
- Arduino. (2021). Guía de Referencia de Arduino. <https://www.arduino.cc/reference/es/>
- Asociación Española de Domótica e Inmótica (s.f.). Qué es Domótica. <http://www.cedom.es/sobre-domotica/que-es-domotica>.
- Asociación Programo Ergo Sum. (2021). Introducción a la programación con AppInventor. <https://www.programoergosum.es/tutoriales/introduccion-a-la-programacion-con-appinventor/>
- Barahona, E.M., Ramírez, U.H., & Vásquez, Á.A. (2018). Caracterización morfoagronómica de cinco variedades de maíz criollo (*Zea mays*) en la zona de San Luis Talpa bajo un manejo orgánico. AGROCIENCIA Cultivando el conocimiento para un mejor futuro, 2. (7), 06-16.
- Blog de Tecnologías. (2021). Arduino. Consultado el 14 de junio de 2021. <https://www3.gobiernodecanarias.org/medusa/ecoblog/rsuagued/arduino/>
- Bricogeek. (s.f.). Módulo Bluetooth HC-06. Consultado el 20 de agosto de 2021. <https://tienda.bricogeek.com/modulos-bluetooth/1351-modulo-bluetooth-hc-06.html>
- C&D TechNología. (2021). Sensor de temperatura y humedad DHT22. Consultado el 22 de agosto de 2021. <https://cdtecnologia.net/sensores/61-sensor-de-temperatura-y-humedad-dht22.html>

- C&D TeCHNologia. (2021). Válvula electromagnética solenoide de metal 1" 110V NC. Consultado el 22 de agosto de 2021.
<https://cdtecnologia.net/conectores-interruptores-y-botones/111-valvula-electromagnetica-solenoide-de-metal-1-110v-nc.html>
- Centro Nacional de Tecnología Agropecuaria y Forestal "Enrique Álvarez Córdova". (s.f.). Guía Técnica: El cultivo del maíz.
- Christoph Thiele, J., Pinto Rodríguez, J. (2014). Operaciones auxiliares de riego en cultivos agrícolas. AGAX0208. IC Editorial.
- Cortes Cadavid, V., Vargas García, M. F. (2020). Diseño e implementación de un sistema de riego automatizado y Monitoreo de variables ambientales mediante IOT en los cultivos urbanos de la Fundación Mujeres Empresarias Marie Poussepin. [tesis de grado, Universidad Católica de Colombia], Repositorio Institucional de la Universidad Católica de Colombia.
<https://repository.ucatolica.edu.co/bitstream/10983/25546/1/Tesis%20Fabian%20Vargas-%20Valeria%20Cortes.pdf>
- DEKALB. (s.f.) MAÍZ DK-357. Consultado el 26 de noviembre de 2021.
<https://www.dekalb.com.mx/es-mx/productos-dekalb/maiz/product-detail-template.html/dk-357-noreste.html>
- del Valle Hernández, L. (2021). Programa fácil. Texto en movimiento en un LCD con Arduino. Consultado el 23 de agosto de 2021.
<https://programarfácil.com/blog/arduino-blog/texto-en-movimiento-en-un-lcd-con-arduino/>
- Electrontools. (2021). Arduino MEGA 2560 características. Consultado el 17 de agosto de 2021. <https://www.electrontools.com/Home/WP/arduino-mega-2560-caracteristicas/>
- Guzmán Navarro, F., Merino Córdoba, S. (2015). Domótica - Gestión de la energía y gestión técnica de edificios. RA-MA.
- Herramientas Tecnológica Profesionales (HETPRO). (2021). LCD 16x2. Consultado el 23 de agosto de 2021. <https://hetpro-store.com/lcd-16x2-blog/>
- It- Electronics Components. (2021). Consultado el 30 de agosto de 2021. DS3231 Real Time Clock Module - AT24C32 RTC With Battery.
- Matías Maestro, I. R., Gabilondo Areta, A., Fernández Valdivielso, C., Ruiz Zamarreño, C., Falcone Lanas, F. J., Del Villar Fernández I., y Milltino Anchorena, J. (2015). Domótica e Inmótica. Instalaciones de telecomunicaciones para edificaciones. (1a. Ed.) Alfaomega Marcombo.

- MCI Electronics. (s.f.). ¿Qué es Arduino?. Consultado el 11 de junio de 2021.
<https://arduino.cl/que-es-arduino/>
- MCI Electronics. (s.f.). Sensor Digital de Temperatura y Módulo AM2302 DHT2. Consultado el 14 de julio de 2021.
<https://www.mcielectronics.cl/shop/product/sensor-digital-de-temperatura-y-modulo-am2302-dht2-29682>
- Mecabot. Laboratorio de electrónica y robótica. (s.f.). Práctica 17. Controlar una electroválvula (110 V) para flujo de agua. Consultado el 20 de agosto de 2021.
<http://mecabot-ula.org/tutoriales/arduino/practica-17-controlar-una-electrovalvula-110-v-para-flujo-de-agua/>
- Méndez Girón, A. M. (2013). Diseño de algoritmos y su programación en C. Alfaomega.
- Mora Zamora, A. (2005). La investigación dirigida.
<https://www.cientec.or.cr/exploraciones/ponenciaspdf/ArabelaMora.pdf>
- Naylamp Mechatronics SAC. (2021). Módulo relay 2CH 5VDC. Consultado el 9 de septiembre de 2021. <https://naylampmechatronics.com/drivers/31-modulo-relay-2-canales-5vdc.html>
- Naylamp Mechatronics SAC. (2021). Módulo RTC DS3231 + EEPROM AT24C32 (I2C). Consultado el 9 de septiembre de 2021.
<https://naylampmechatronics.com/sensores/107-modulo-rtc-ds3231-eprom-at24c32-i2c.html>.
- Niple Blog. (2016). Sensor de humedad de suelo YL-69. Consultado el 9 de septiembre de 2021 <http://www.niplesoft.com/blog/2016/01/19/sensor-de-humedad-de-suelo-yl-69/>
- Pérez López, R. (2021). Programación estructurada. IES Doñana. Consultado el 23 de julio de 2021 <https://pro.iesdonana.org/apuntes/programacion-estructurada-apuntes.pdf>.
- Rechner Sensors. (s.f.). El sensor de temperatura. Consultado el 17 de septiembre de 2021. <https://www.rechner-sensors.com/es/documentacion/knowledge/el-sensor-de-temperatura>
- Riego.org.(2021). Glosario de riego. <https://www.riego.org/glosario/humedad-del-suelo-fao/>.

- Rojas Cala, R. E. y Areiza Guzmán, L. A. (2015). Sistema Domótico. [tesis de postgrado, Universidad Santo Tomás]. Repositorio Institucional de la Universidad Santo Tomás.
<https://repository.usta.edu.co/bitstream/handle/11634/2359/Areizaluis2015.pdf?sequence=1&isAllowed=y>
- Sznajdleder, P. A., (2012). Algoritmos a fondo con implementaciones en C y Java (1ª. Ed.). Alfaomega.
- Simón Mori, G. A. (2018). Implementación, control y monitoreo de un sistema de riego por goteo subterráneo con microcontroladores. [tesis de grado, Universidad Nacional Agraria La Molina]. Repositorio Institucional de la Universidad Nacional Agraria La Molina.
<https://repositorio.lamolina.edu.pe/handle/20.500.12996/3610>
- TodoMicro.(2021). Modulo bluetooth HC-06. Consultado el 27 de agosto de 2021.
<https://www.todomicro.com.ar/arduino/24-modulo-bluetooth-hc-06.html>.
- Torrente Artero, O., (2013). Arduino. Curso práctico de formación (1ª. Ed.). Alfaomega.
- Unit Electronics. (2021). Relevador 5V DC SRD-5VDC-SL-C. Consultado el 23 de septiembre de 2021. <https://uelectronics.com/producto/relevador-5v-dc-srd-5vdc-sl-c/>

