

Design and implementation of an automated greenhouse irrigation controller, made with recycled materials, focused on precision agriculture.

Universidad Tecnologica La Salle, Member number #95666642, D. Centeno, Estudiante de ingeniería cibernetica electrónica, IEEE, dcentenogonza@gmail.com

Abstract – Rapid climate change, agricultural exploitation and the reduction of arable land, demand new methods for sustainable agriculture. Greenhouse farming is a viable solution to this problem, and although it presents many challenges in terms of maintenance and management, technologies such as: smart sensors, communication devices and microcontrollers can be used, which help to minimize the possible inconveniences that occur presented in the implementation of these greenhouses. With these devices and sensors we will be able to monitor the climate, soil humidity, etc. Thus being able to control some variables of the local environment and even in outdoor conditions. This document presents the design with recycled materials, development and implementation of an automated irrigation controller for greenhouses, showing the hardware and software used and the purpose of each one, also showing the operation of the controller and the results obtained.

Keywords - *Arduino, Automated, Bluetooth, Hardware, Greenhouse, Controller, Sensors, design, recycled materials.*

I. Introducción

El aumento de la temperatura media anual y la disminución de la precipitación que se prevén para el año 2030, debido a los efectos del cambio climático, tendrá impactos significativos en la agricultura de todo el mundo[1].

Debido al incremento de la temperatura, causado por el cambio climático, se dificulta el crecimiento óptimo de los cultivos, por ende su exportación, provocando una pérdida de la seguridad alimentaria de algunos productos.

Debido a la degradación de las tierras y las dificultades de mantener un control del riego óptimo para los cultivos, los sistemas de riegos automáticos ofrecen a los usuarios una gran flexibilidad de regular y controlar las operaciones de riego, ya que los sistemas de riegos funcionan con poca intervención para su funcionamiento, con esto se logra reducir el manejo inadecuado de los recursos hídricos, se mejora la calidad de

los cultivos y el ahorro del coste de agua al realizarse los períodos de riego de forma automática [2].

Hoy en día los avances en sensores, actuadores y microcontroladores nos permiten tener un gran nivel de hardware y software [3]. En este documento se explica el diseño, desarrollo e implementación de un controlador para invernadero totalmente automatizado, mostrando los datos recolectados del entorno (temperatura relativa, humedad relativa, índice de calor, humedad del suelo y luminosidad ambiental). Una vez recolectados los datos, se determinó el momento idóneo para el riego de las plantas, además con estos se puede hacer un pequeño análisis del clima y cómo ha sido el comportamiento en todo el día, de tal manera, que se pueda identificar si ha sido un día bueno, regular o malo para los cultivos.

II. Objetivos

Objetivo general

- Implementar un controlador de riego automatizado para invernaderos comunitarios o familiares, como solución para mantener un ambiente óptimo para el cultivo, esto a partir de materiales reciclados.

Objetivos específicos

- Aplicar métodos novedosos para el diseño de un controlador de riego en invernaderos utilizando materiales reciclados.
- Desarrollar un programa para un Arduino Nano con las instrucciones y variables a monitorear del controlador.
- Poner en práctica el controlador de riego para un invernadero de 6 x 4 metros en instalaciones de la Universidad tecnológica La Salle.

III. Metodología

El proyecto se desarrolló en un invernadero de las instalaciones de la Universidad Tecnológica La Salle (ULSA). Los pasos que se presentan a continuación fueron los necesarios para el diseño de un controlador para invernaderos comunitarios o familiares a partir de materiales reciclados y reutilizados, enfocándonos en la economía circular y la agricultura de precisión. De igual forma se presenta el desarrollo de cada fase del controlador con los componentes utilizados. El sistema fue monitoreado por dos semanas y mediante la observación se fueron recolectando los datos medioambientales del entorno, para realizar un análisis del óptimo funcionamiento en su implementación.

IV. Diseño del controlador de riego

El controlador está diseñado de tal manera que el usuario no tenga que realizar ninguna acción en el funcionamiento, el usuario podrá realizar una solicitud mediante conexión bluetooth, la cual le proporcionará los datos recolectados. Todo el sistema está desarrollado con la placa de desarrollo arduino nano [4].

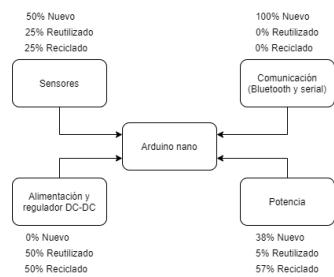


Fig. 1: Diseño y desarrollo del controlador - (Autoría propia).

En la figura 1 se puede observar además del diseño del controlador, los porcentajes de reciclaje, reutilización y componentes nuevos que se utilizaron en cada fase.

Con reciclaje nos referimos a los materiales utilizados que fueron extraídos de componentes dañados como UPS, los componentes reutilizables se trajeron de proyectos antiguos y los componentes nuevos se compraron completamente nuevos ya sea, porque no disponíamos de ellos o por la necesidad de dichos materiales.

A. Funcionalidad del controlador

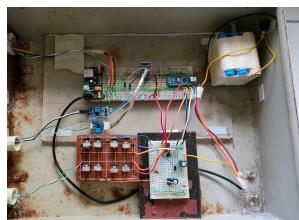


Fig. 2: Controlador de riego automatizado con materiales reciclados - (Autoría propia).

- El controlador ofrece una gran flexibilidad y confiabilidad para determinar el momento idóneo para realizar el riego mediante las variables del entorno.
- El controlador recolecta y almacena las variables de los últimos 2 días.
- Proporciona comunicación bluetooth y serial para la transferencia de los datos recolectados.
- Controlador de bajo coste y eficaz, construido con materiales reciclados.

B. Sensores

Para detectar los valores de “Temperatura relativa”, “Índice de calor” y “Humedad relativa” se utilizó un sensor HT11 [7].



Fig. 4: Protección y ubicación del sensor DHT11 - (Autoría propia).

Se utilizó un pequeño LDR para calcular la luminosidad ambiental medidas en porcentajes de 0% a 100% [8].



Fig. 5: Protección y ubicación del sensor LDR - (Autoría propia).

Se utilizaron dos sensores “Raindrop sensor analyser” para la medición de la humedad del suelo en porcentajes de 0 a 100%. Cabe recalcar que mientras más sensores se coloquen en el terreno mejor serán las lecturas, ya que una vez recolectada la lectura de cada sensor se saca un promedio de las mediciones, de tal manera sabremos qué tan húmedo o seco está el terreno en general [9].

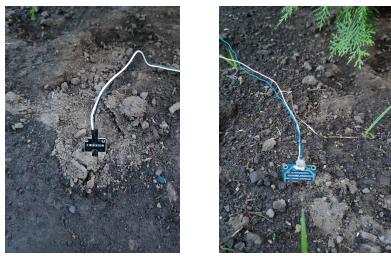


Fig. 6: Ubicación de los sensores de humedad del suelo - (Autoría propia).

C. Comunicación (Bluetooth y USB serial). a. Comunicación Bluetooth

Para la comunicación bluetooth entre el arduino nano y otro dispositivo, se utilizó el módulo bluetooth “HC-06”, para establecer una conexión bluetooth con este módulo y mandar una solicitud con las letras claves, en este caso “d” o “D”, el controlador nos proporcionará todos los datos recolectados de las últimas horas mediante el módulo “HC-06” [10].

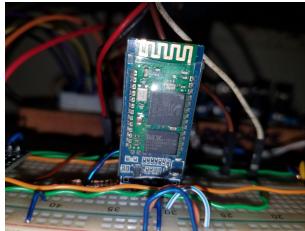


Fig. 7: Módulo Bluetooth “HC-06” - (Autoría propia).

b. Comunicación USB serial.

Si no se cuenta con un computador o dispositivo con bluetooth tiene la posibilidad de visualizar los datos mediante la comunicación USB serial, de tal manera que si el operario conecta el cable del arduino nano a su computador en el monitor serial del IDE de arduino aparecerán todos los datos recolectados de las últimas 48 horas.

D. Alimentación y regulador DC-DC. a. Alimentación.

Para la alimentación energética del controlador se elaboró un arreglo de baterías 18650 estas baterías fueron recicladas de laptops dañadas. Se utilizaron un total de 9 baterías de las cuales se elaboraron 3 paquetes en serie y posteriormente fueron conectadas en paralelo, de esta manera tenemos un arreglo de 12V y 10,800mAh. Este arreglo de baterías nos da una autonomía de 15 horas con el sistema funcionando continuamente.

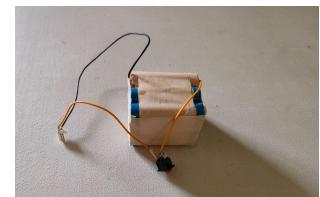


Fig. 8: Arreglo de baterías de 12V y 10,800mAh - (Autoría propia).

E. Regulador de voltaje DC-DC.

Dado que muchos de los dispositivos utilizados funcionan con 5V, como los sensores o, 3.3V que requiere el módulo de comunicación bluetooth, se necesitó un regulador de voltaje para alimentar estos dispositivos [11].



Fig. 9: Regulador de voltaje DC-DC - (Autoría propia).

F. Potencia

a. Controlador “D640DVE”

Debido al alto consumo de corriente del motor encargado de realizar la apertura del grifo, se diseñó un circuito integrado con mosfet IRF640 de canal N. Que fueron reciclados de un sistema de alimentación ininterrumpida “Back-UPS ES 750”. El integrado al cual se le nombró como “D640DVE” es muy similar a un integrado L293D es decir es un puente H de tipo cuadruple [12].

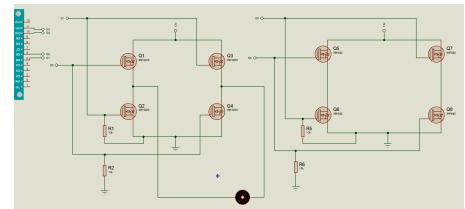


Fig. 10: Esquema eléctrico del integrado D640DVE y conexión al arduino nano - (Autoría propia).

b. Duplicador de Voltaje.

Por las características de los mosfet, el voltaje de saturación en Gate-source (V_{GS}) tiene que ser mayor a 15V esto varía entre cada mosfet, en el caso de los mosfet IRF640 se necesita un voltaje de saturación mayor a 20V para que el voltaje de Drain-source (V_{DSS}) sea igual a V_{CC} . Para que los mosfet IRF640 llegaran a su punto de saturación se desarrolló

un duplicador de voltaje con el integrado 555 el cual está en una configuración astable dando los pulsos para la carga y descarga de los capacitores, y de esta manera duplicar el voltaje a 24V que son más que suficientes para saturar los mosfet. Para dar los pulsos de 24V a la entrada Gate-source (V_{GS}) del integrado “D640DVE” se utilizó un integrado L293D que va conectado al arduino nano, esté integrado funciona también como una protección ante cualquier corriente inversa del duplicador hacia el arduino nano [13].

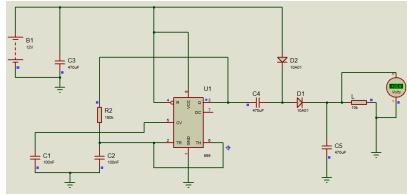


Fig. 11: Esquema eléctrico del duplicador de voltaje - (Autoría propia).

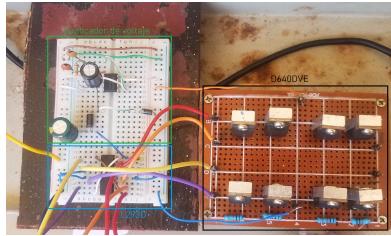


Fig. 12: Integrado de potencia “D640DVE”, duplicador de voltaje e integrado L293D - (Autoría propia).

V. Diseño y desarrollo del programa

El programa está desarrollado en el lenguaje que utiliza arduino que está basado en C y C++. El programa está construido de tal forma que, cada proceso es independiente uno del otro, es decir ningún proceso afecta el funcionamiento del otro. Esto quiere decir que se consiguió un entorno de multiprogramación [5].

A. Condición de riego

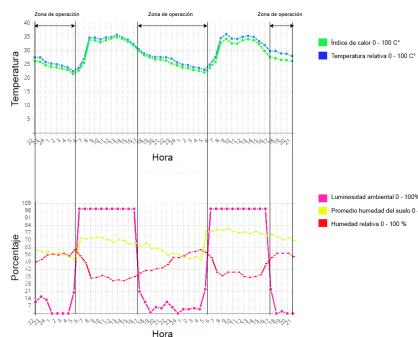


Fig. 13: Gráfica de los datos recolectados y zonas de operación del controlador - (Autoría propia).

Como se muestra en la Figura 13, se utilizaron los valores de temperatura relativa, promedio de la humedad del suelo y luminosidad ambiental recolectados por los sensores del controlador para evaluar los mejores momentos para realizar el riego. La condición a cumplir es cuando la temperatura relativa es menor o igual a 32 grados, el promedio de la humedad del suelo es menor al 50% y la luminosidad ambiental es menor o igual al 30%. Cuando las 3 condiciones se cumplan se abrirá el grifo durante 60 minutos ya que es un riego por goteo de 2 galones por hora [6].

B. Sistema de multiprogramación

Gracias a la función millis incorporada en la librería de arduino se logró hacer un sistema con planificador de tareas lo cual nos permite la ejecución de cada proceso sin interrupciones entre ellos.

VI. Monitoreo del sistema

El controlador pasa monitoreando y regando de forma automatizada durante dos semanas con el fin de confirmar el correcto funcionamiento del sistema a la hora de recolectar y guardar todos los datos. Fueron monitoreados todos los sensores y almacenado los datos recolectados, para evaluar los momentos óptimos en que se realiza el riego. Debido a las limitaciones del hardware, en la programación solo se registraron los datos de las últimas 48 horas.

VII. Evaluación del funcionamiento y posibles mejoras

A. Evaluación del funcionamiento

El controlador funciona de forma eficiente en cada tarea que se le programó, como se observa en la figura 13 están las zonas de operación del sistema cada 11 horas, notándose en la gráfica de color amarrillo que el porcentaje de humedad del suelo obtenido es del 80% en cada ciclo de riego por goteo.

Se observó que el arreglo de baterías diseñado en el apartado de alimentación DC-DC no fue suficiente para mantener encendido el sistema por 24 horas ya que su capacidad máxima de funcionamiento es de 15 horas; por lo que fue necesario utilizar fuentes de alimentación externas.

Se utilizó un motor paso a paso reutilizado de una impresora para la apertura y cierre del grifo siendo necesario desarrollar el integrado “D640DVE”, pero en el transcurso del proyecto se hizo el cambio a un motor de vidrios para autos DC. Este tipo de motores funcionan con un puente H sencillo y no es necesario un puente H de tipo cuadruple como el que se utilizaría con el motor paso a paso.

B. Posibles mejoras

Debido a la capacidad máxima de la batería del controlador mencionada en el apartado anterior, para mejorar la eficiencia del sistema se puede desconectar módulos innecesarios para luego con uso de relés conectarlos antes de su funcionamiento. Esto nos ayudará mucho en el rendimiento

energético de las baterías.

El controlador es muy flexible y fácil de utilizar por el operario cuando se realiza la solicitud de los datos que ha recolectado el controlador. Sin embargo se podría desarrollar una aplicación móvil propia, que se conecte vía bluetooth al arduino para darle más funcionalidades, como la opción de abrir y cerrar la válvula de forma manual o hacer un riego con temporizador, de tal forma que el operario le indique cuánto tiempo quiere que riegue sin importar si se cumplen o no las condiciones de riego.

Para desarrollar un sistema completamente ininterrumpido y ecológico sería conveniente la inclusión de un panel solar de 12V al controlador. De esta manera en el día funciona con el panel solar mientras se cargan las baterías y en la noche trabajará al 100% con las baterías.

VIII. Conclusiones

Actualmente el controlador está funcionando continuamente en un pequeño invernadero en las instalaciones de la universidad donde se está monitoreando y llegando cada 2 días para la recarga de las baterías.

Se logró con éxito el diseño con materiales reciclados y reutilizados en el desarrollo del controlador, logrando un entorno de multiprogramación y un sistema totalmente automatizado, implementando el controlador en un invernadero el cual fue monitoreado por dos semanas donde se obtuvo con éxito todos los objetivos de recolección de los datos y riego automático.

Teniendo en cuenta las limitaciones antes mencionadas en el apartado de “Evaluación del funcionamiento y posibles mejoras”, el controlador brinda una gran flexibilidad y funcionalidad en el momento de recolectar, almacenar y evaluar los momentos idóneos para realizar el riego de forma automatizada.

Referencias

- [1]. C. Bourocle, P. Imbach, P. Laderach, B. Rodriguez, C. Medellín, E. Fung, "La agricultura en Nicaragua y el cambio climático", Centro Internacional de Agricultura Tropical (CIAT), 2014.
- [2]. D.S Pavithra, and M.S Srinath, "GSM based Automatic Irrigation Control System for Efficient Use of Resources and Crop Planning by Using an Android Mobile", IOSR, 2014.
- [3]. A.Salah and A. Zneid, "Evolution of microcontroller-based remote monitoring system applications", IJECE, 2019.
- [4]. E. Suarez, "RISC-V AN OPEN SOURCE ISA", From Universidad Cantabria, 2019.
- [5]. B. Evans, "Beginning Arduino Programming", New York: Spring Street, 6th Floor, 2011.

[6]. D. Centeno. (2022, march, 03), Program for controller for automatic greenhouse (Versión 8.4). [Online]. Available: <https://github.com/DCenten0/Program-for-controller-for-automatic-greenhouse.git>

[7]. D-Robotics, DHT11 Humidity & Temperature Sensor datasheet, July 2010.

[8]. RS components, Light Dependent Resistor datasheet, march 1997.

[9]. Research Design Lab, Soil Moisture Sensor datasheet.

[10]. Guangzhou HC Information Technology Co, HC-06 datasheet, september 2006.

[11]. CircuitAttic, YwRobot Breadboard Power Supply MB-V2 datasheet, september 2013.

[12]. SGS-Thomson microelectronics, L293D datasheet, june 1996.

[13]. Vishay Siliconix, IRF640, SiHF640, 91000, (2022, Jan. 1)



Daniel J. Centeno González nació en León, Nicaragua el 18 de abril de 2001.

Se recibió de bachiller en ciencias y letras del instituto La Asunción en León en el 2018 y actualmente cursando su cuarto año de la carrera de ingeniería cibernética electrónica en la Universidad Tecnológica La Salle.

Actual miembro de la rama IEEE ULSA 2022 y Tesorero de la misma.