

Sistema de Control de una cámara de germinación hidropónica con IoT.

Marcelo Marinelli , Rubén R. Urquijo
Instituto de Biotecnología de Misiones "María Ebe Reca". Campus U.Na.M. Ruta
12 Km 7,5, Miguel Lanús, Posadas, Misiones, Argentina.
{marcelo, ruben}@gmail.com

Resumen. Se presenta el diseño de un sistema de control difuso y telemetría con tecnología IoT para una cámara de germinación de cultivos hidropónicos. El sistema fue desarrollado con software NODE-RED y hardware Raspberry Pi, toma mediciones de humedad y temperatura dentro de la cámara y, por intermedio de un controlador difuso tipo Sugeno, genera señales de control para un sistema de ventilación forzada. El programa envía datos de las variables intervinientes y de las reglas difusas que se activan por mail y Twitter permitiendo recibir el estado del sistema desde cualquier dispositivo con conexión a internet.

Palabras claves: Lógica Difusa, Hidropónia, Internet de las Cosas, Web Services, E-Mail.

1 Introducción

La cámara de germinación para cultivos hidropónicos consiste en un recipiente plástico con una solución de nutrientes sobre la cual flota una placa de poliestireno expandido con perforaciones rectangulares. La evolución de las tecnologías de Internet tienden a integrar servicios de la web, redes sociales, redes de sensores que pueden obtener información de variables ambientales o de cualquier otro tipo y además controlar actuadores, cámaras, etc.; la integración de estos dispositivos y servicios dan como resultado la IoT [2] [3]. Cuando se trabaja con un conjunto de sensores inalámbricos distribuidos espacialmente e interconectados por distintos concentradores, se denomina WSN (*wireless sensor network*) utilizado para monitoreo ambiental, sistemas interconectados de energía, etc., [4].

A la hora de desarrollar aplicaciones del tipo IoT, debemos tener en cuenta que se deben considerar la adquisición de datos proveniente de sensores, tomar decisiones para realizar acciones sobre actuadores, cámaras u otros dispositivos y en tiempo real interactuar con servicios en línea. Si tuviéramos que programar una aplicación IoT de este tipo, se necesitan conocer varios lenguajes y protocolos, así como desarrollar APIs específicas y vincularlas con todos los componentes del sistema [5]. Por esto, surgen varias aplicaciones que integran los componentes que permiten desarrollar aplicaciones IoT. Estas se componen de nodos con funciones específicas y se interconectan por conexiones que intercambian flujo de datos.

Para alojar cubos de espuma fenólica (Figura 1), en estos orificios se siembran las semillas para su germinación y desarrollo hasta su trasplante a otro sistema hidropónico que puede ser de raíz flotante o NFT (*Nutrient Film Technique*) [1]. La cámara posee iluminación artificial por medio de paneles led o luces de bajo consumo y un sistema de ventilación forzada cuyo flujo de aire se controla mediante un control PWM (*pulse-width modulation*), en su interior posee un sensor de humedad y temperatura DHT22 (Figura 2). Para mejorar el proceso de desarrollo de las plántulas, se diseñó e implementó un sistema basado en tecnología IoT (*Internet of Things* o Internet de las Cosas) en un sistema embebido, que consta de un control difuso que, en función de las variables humedad y temperatura, dentro de la cámara, proveen una salida de control de potencia del sistema de ventilación forzada [1].



Figura 1. Placa de poliestireno expandido con los cubos de espuma fenólica en los cuales se desarrollaron las plántulas.

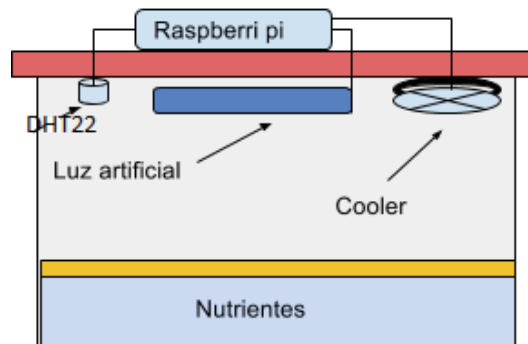


Figura 2. Diagrama de la cámara de germinación.

2 Hardware del sistema

En este trabajo se desarrolló un sistema de control de una cámara de germinación con un computador de placa simple o SBC (*Single board computer*) Raspberry Pi (<https://www.raspberrypi.org/>) que tiene la característica de ser un dispositivo de bajo costo y soporta sistemas operativos GNU/Linux ARM (Debian, Fedora, Arch Linux), RISC OS2.

Tiene un tamaño reducido (85 mm de longitud por 56 mm de ancho) y posee una salida HDMI, una ethernet y 4 USB con lo que, si se conecta un monitor, teclado y ratón, se convierte en un ordenador. Tiene una CPU ARM1176JZF-S (armv6k) a 700 MHz³, GPU Broadcom VideoCore IV3, memoria 512 MiB, Capacidad de almacenamiento Tarjeta SD o SDHC (Figura 3).

Debido a su reducido tamaño se puede montar sobre la cámara de germinación y con sus puertos de propósito general GPIO (*General Purpose Input/Output*) se pueden controlar sensores y actuadores para desarrollar un sistema de control de la misma.

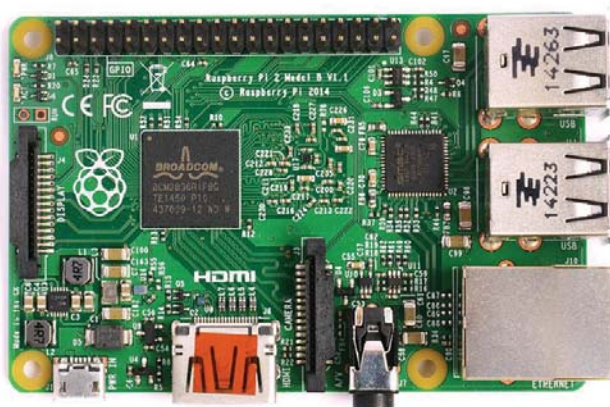


Figura 3. Raspberry Pi 2 modelo B.

2.1 Puertos GPIO

Los puertos GPIO son, como su nombre lo indica, puertos de entrada/salida de propósito general, es decir, un conjunto de conexiones que pueden utilizarse como entradas o salidas para diversos usos. Estos puertos representan la interfaz entre la Raspberry Pi y el medio exterior [6].

Posee un total de 17 puertos configurables como entrada o salida. Por defecto están todos configurados como entradas excepto los GPIO 14 y 15 que operan como salidas (Figura 4).

En este caso se utilizó el puerto GPIO 5 como entrada digital para obtener los datos provenientes del sensor de humedad y temperatura.



Figura 4. Salidas GPIO de Raspberry Pi.

2.2 Sensor de temperatura DHT22

Para obtener los valores de las variables humedad relativa y temperatura, se utilizaron sensores DHT22 (Figura 5 y Tabla 1) que miden estos parámetros. Estos sensores poseen un rango de operación de 0 a 100 % HR y de -40 a 80 °C y contando con una precisión de humedad de 2% RH y de temperatura 0,5%. Por otra parte, tiene la característica de trabajar con protocolo serial enviando los datos por una sola vía, por el pin de datos (Figura 5).

Tabla 1. Pines de salida de DHT22.

Pin	Función
1	5 v
2	Datos
3	NC
4	GND

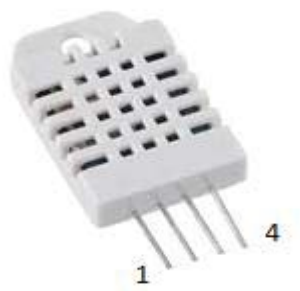


Figura 5. Sensor DHT22.

De acuerdo con la Tabla 1, el pin 1 corresponde a la alimentación, el 4 a masa (GND) y el pin 2 tiene una resistencia de *pull up* de 10 K por donde son enviados los datos de humedad y temperatura en forma serial hacia el puerto GPIO correspondiente de la Raspberry Pi (Figura 6).

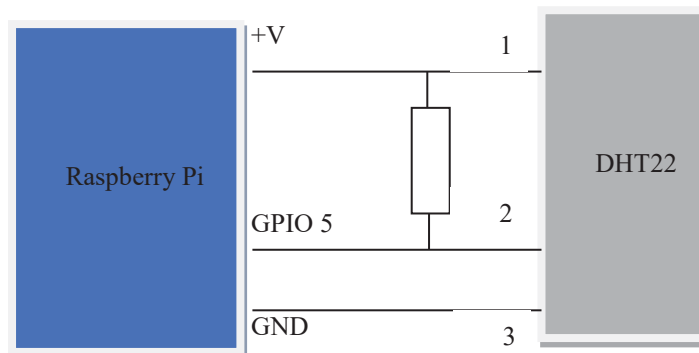


Figura 6. Conexiones del sensor DHT22 con el puerto GPIO.

2.3 Control del sistema de ventilación forzada

La salida PWM del puerto GPIO (GPIO 17) se conecta a un amplificador que permite controlar los niveles de potencia del sistema del *cooler*, esta salida entrega un nivel de potencia calculado por el controlador difuso en función de las variables de entrada y de la base de conocimientos del mismo.

3 NODE-RED

Es un editor basado en web para generar aplicaciones de IoT. De código abierto, creado por el equipo de *IBM Emerging Technology*. Posee una interfaz amigable que, mediante la función de arrastrar y soltar, permite colocar en el área de trabajo nodos que representan APIs de web, servicios en línea o dispositivos de hardware. Estos nodos se pueden unir con un cableado que representan el flujo de datos [7], también se pueden programar funciones en *JavaScript*; de esta manera es posible programar nodos con determinadas funcionalidades. El motor de tiempo real se basa en Node.js que es un entorno para ejecutar *JavaScript* realizado con el motor V8 de Chrome. Dado que Node.js trabaja con arquitectura basada en eventos no bloqueantes que se ejecutan del lado del servidor, esto hace que consuma poco costo de hardware lo que lo hace ideal para correr en placas Arduino (www.arduino.cc) o Raspberry Pi [8].

3.1 Controlador difuso

Se optó por un sistema de control difuso debido a la mayor interpretabilidad que posee para ser utilizado por expertos de otras disciplinas, como en este caso en que la base de conocimientos del sistema difuso se basó en información provista por expertos en Biología Vegetal [1].

Para la codificación del controlador difuso se seleccionó el tipo Sugeno [9], ya que al no poseer etapa de defusificación es posible programarlo con menor costo de procesamiento. Para la implementación se utilizó un nodo del tipo “función” y se codificó en JavaScript.

A la salida de este nodo se obtiene un flujo de datos que contiene la información de las reglas que se activan y el valor z que corresponde al porcentaje de potencia que se aplicará al nodo siguiente con salida PWM conectado al *cooler*.

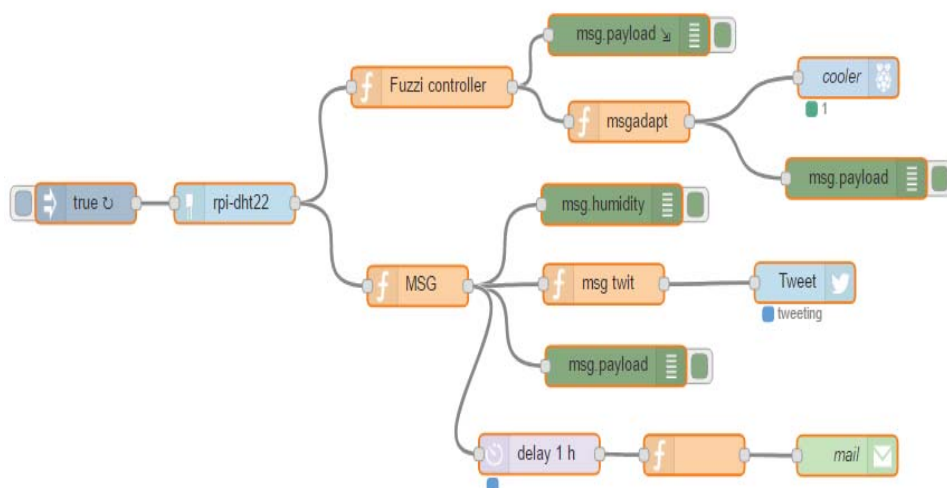


Figura 8. Diagrama de nodos del sistema de control de la cámara de germinación.

4 Telemetría social

La implementación de mensajería en redes sociales se hace a través de Twitter cada 10 minutos y por mail cada 60 minutos.

Estos mensajes son configurables y permiten tener información a distancia en un dispositivo móvil para estar al tanto de los valores de las variables intervinientes en la cámara de germinación y, en caso de algún valor anómalo, poder tomar una acción correctiva.

5 Resultados

El sistema de control diseñado combina tecnologías de IoT utilizando como soporte de hardware placas del tipo SBC (Raspberry Pi) y software Node-Red en el cual se implementó un sistema de control basado en lógica difusa.

La telemetría social permite informar los estados de las variables de la cámara por mail y por Twitter, de esta forma se tiene actualizado el estado de las variables intervinientes y poder tomar acción correctiva en caso de ser necesario.

6 Referencias

1. Marinelli, Marcelo J, Otegui, Mónica B, Zapata, Pedro D, & Acosta, Nelson H. (2015). Control Difuso de una Cámara de Germinación para Hidroponía. *Revista de Ciencia y Tecnología*, (24), 42-47. Recuperado en 25 de abril de 2017, de http://www.scielo.org.ar/scielo.php?script=sci_arttext&pid=S1851-75872015000200007&lng=es&tlng=es.
2. Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7), 1645-1660.
3. Kelly, S. D. T., Suryadevara, N. K., & Mukhopadhyay, S. C. (2013). Towards the implementation of IoT for environmental condition monitoring in homes. *IEEE Sensors Journal*, 13(10), 3846-3853.
4. Mainetti, L., Patrono, L., & Vilei, A. (2011, September). Evolution of wireless sensor networks towards the internet of things: A survey. In *Software, Telecommunications and Computer Networks (SoftCOM)*, 2011. 19th International Conference on (pp. 1-6). IEEE.
5. Blackstock, M., & Lea, R. (2014, October). Toward a distributed data flow platform for the web of things (distributed node-red). In *Proceedings of the 5th International Workshop on Web of Things* (pp. 34-39). ACM.
6. Jain, S., Vaibhav, A., & Goyal, L. (2014, February). Raspberry Pi based interactive home automation system through E-mail. In *Optimization, Reliability, and Information Technology (ICROIT)*, 2014 International Conference on (pp. 277-280). IEEE.
7. Kleinfeld, R., Steglich, S., Radziwonowicz, L., & Doukas, C. (2014, October). glue. things: a Mashup Platform for wiring the Internet of Things with the Internet of Services. In *Proceedings of the 5th International Workshop on Web of Things* (pp. 16-21). ACM.
8. Salihbegovic, A., Eterovic, T., Kaljic, E., & Ribic, S. (2015, May). Design of a domain specific language and IDE for Internet of things applications. In *Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2015 38th International Convention on (pp. 996-1001). IEEE.
9. Takagi, T., & Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE transactions on systems, man, and cybernetics*, (1), 116-132.