

## Controlador de riego para un invernadero empleando el Internet de las cosas y Raspberry Pi B+

Reyes Ibarra Luis Alberto, Herrera Castillo Perla Alejandra, Ríos Chavarría Christian  
Universidad Politécnica de Durango  
Ing. Telemática

Carretera Durango-México Km 9.5 Tel. 618 1501300  
C.P. 34300 Durango, Dgo

luis.reyes@unipolidgo.edu.mx, perla.herrera@unipolidgo.edu.mx, Christian.rios@unipolidgo.edu.mx

### RESUMEN

Se presenta la implementación de un controlador de riego para el invernadero instalado en la Universidad empleando el Internet de las cosas (IoT), dicho término, se refiere a la conexión de objetos cotidianos con Internet, sin la necesidad de crear un servidor Web.

Una de las ventajas de este proyecto es la utilización de una Raspberry B+, el cual es un dispositivo que se puede enlistar a cualquier plataforma de IoT, además cuenta con múltiples pines GPIO para el control de actuadores, unidad serial para la recepción de datos, moduladores de ancho de pulso y puerto I2C.

También a través de IoT se pueden monitorear variables y conservar historiales lo que permitirá contar con registros para tomar decisiones con conocimiento de causa.

**Palabras clave:** Internet de las cosas, Raspberry, Riego

### ABSTRACT

It is presented the implementation of an irrigation controller for a greenhouse installed at the University using the Internet of Things (IoT), the term refers to the connection of everyday objects with Internet, without the need of a Web Server.

One of the advantages of this project is the use of a Raspberry B +, which is a device that can enlist any platform IoT, also has multiple GPIO pins to control actuators, a serial unit for receiving data, pulse width modulators and the I2C port.

Also through IoT the variables can be monitored and retain records that will allow to make decisions knowingly.

**Keywords:** Internet of things, Raspberry, Irrigation

### 1. INTRODUCCIÓN

Hoy en día los sistemas de monitoreo y control cuenta con determinada conectividad Web, lo que lo que hace que el costo aumente, algunos dependen todavía de crear servidores Web para su buen funcionamiento.

Una de las motivaciones por la que surgió este proyecto es no depender de un operador para encender ya sea el riego por goteo o el riego con nutrientes (fertiriego), por lo que se implemento un control de actuadores empleando el Internet de las cosas, con esto, permitirá encender cualquier bomba desde cualquier lugar donde se tenga conectividad a la Web.

De inicio el controlador de riego cuenta con la capacidad de encender 4 bombas diferentes y monitorear ya sea su propio encendido o el monitoreo de sensores de humedad de suelo de tipo On-Off.

Las pruebas de este controlador se realizaron en el Invernadero instalado en la Universidad (Figura 1), en este, se cuenta con sistema de riego por goteo además de un sistema de riego con nutrientes (fertiriego).



Figura 1. Invernadero tipo tunel con ventana cenital.

#### 1.1. Sistemas de riego avanzados

En la actualidad los sistemas de riego avanzados cuentan con diversas características como lo son el tipo de conectividad (GSM, Web), datalogger, son modulares (redes de sensores inalámbricos), manejo de actuadores (múltiples IO), etc. Un ejemplo es el modelo de la empresa iobridge (Figura 2) [1]



Figura 2. Sistema modular para el control de riego

Este sistema es modular ya que se pueden interconectar diversos módulos como el de la Figura 2 empleando módulos de RF, estos a su vez, se conectan a un modulo central (Figura 3) que servirá como enlace hacia el Internet de las cosas IoT.



Figura 3. Sistema iobridge para interconexión a IoT

Uno de los principales inconvenientes de estos sistemas es que al ser dispositivos avanzados muchas de las veces los productores no son capaces de configurarlos o manejarlos por lo que requieren de capacitación en su manejo que muchas de las veces no es posible o es muy costosa.

Es por lo anterior que el objetivo de este proyecto es crear un controlador de riego con Raspberry el cual sea capaz tanto de controlar un sistema de riego por goteo así como el fertiriego de un Invernadero, todo ello, considerando que no fuese modular para facilitar su manejo y configuración pero a la vez que cuente con la capacidad de múltiples GPIO, lo cual es posible empleando la Raspberry; con lo anterior es posible reducir costos sin quitar funcionalidad a este tipo de controladores de riego. Además una de las funcionalidades que se le pretende dar y que otros no la tienen es la configuración vía Bluetooth del sistema y la inserción de una pantalla táctil para el control de riego en forma manual.

## 1.2 Importancia del manejo del riego en el Invernadero

Actualmente se cultiva dentro del invernadero de la Universidad una variedad de tomate, la cual requiere determinado ciclo de riego por goteo además de la inyección de nutrientes a través del mismo.

El manejo del agua en el cultivo de tomate es una de las acciones más críticas del proceso de producción. El riego adecuado permite compensar las extracciones de agua y nutrimentos de la planta, controla la acumulación de sales y mantiene un adecuado nivel de oxígeno en el ambiente radical. Además, el riego afecta positiva o negativamente el crecimiento, la producción, la condición fitosanitaria de la planta, el uso efectivo de fertilizantes y el consumo de energía. El riego permite manipular en cierto grado el crecimiento y desarrollo del cultivo. [2]

## 1.3 Introducción a la plataforma Raspberry Pi B+

En la Figura 4 se muestra la Raspberry, esta, cuenta con un chip integrado Broadcom BCM2835, que contiene un procesador ARM11 con varias frecuencias de funcionamiento y la posibilidad de subirla (haciendo overclocking) hasta 1 GHz, un procesador gráfico VideoCore IV, y 512MB de

memoria RAM. La última Raspberry Pi 2 B+ cuenta con 1GB de memoria RAM. [3]



Figura 4. Raspberry Pi B+

### 1.3.1 Hardware de la Raspberry Pi B+

Algunas de las características más relevantes de la Raspberry se muestran en la Tabla 1.

Tabla 1. Características de la Raspberry B+

Conector	Característica
Ethernet	10/100 BaseT Ethernet socket
Salida Video	HDMI (rev 1.3 & 1.4) Composite RCA (PAL and NTSC)
Salida Audio	3.5mm jack, HDMI
USB	4 x USB 2.0 Connector
Conector GPIO	40-pin 2.54 mm (100 mil) 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines
Conector Cámara	15-pin MIPI Interfaz serial para cámara (CSI-2)
Conector Display	Interfaz Serial para Display (DSI) 15
Lector de tarjetas MicroSD	Soporta MicroSD de 4G o más

### 1.3.2 Pines GPIO

El modelo B+ cuenta con un conector de 40 pines para el manejo de GPIO (Pines de entrada y salida de propósito general).

La tensión de trabajo del puerto es de 3,3v para uno. Además, la corriente máxima que puede suministrar es de 16 mA.

Todos los pines GPIO se pueden gestionar directamente a través de código, ya sea para poner un valor o leer un valor de un elemento externo.

Además, se pueden configurar interfaces complejos de entrada/salida por lo que algunos pines pasan a ser gestionados directamente por el chip del procesador.[4]

### 1.3.3 Configuración inicial de la Raspberry

Existen diferentes versiones de sistema operativo que soporta la Raspberry, una de las versiones más usadas es la Raspbian que es la utilizada en este proyecto.

El sistema operativo se debe copiar en una MicroSD de 4G o más, una vez que se tiene esto, se procede a la configuración inicial de la Raspberry donde los puntos más importantes a configurar son:

- Región
- Idioma del teclado
- Modo de arranque: Gráfico, Terminal
- Expansión del espacio de memoria de la tarjeta SD.

Una vez que arrancamos la Raspberry si seleccionamos el modo de arranque gráfico nos aparece una interfaz parecida a la de Windows (Figura 5), enseguida si se cuenta con un módulo WiFi se procede a configurar una SSID para contar con acceso a la red y con ello instalar diversos componentes y actualizaciones que requiere la Raspberry.

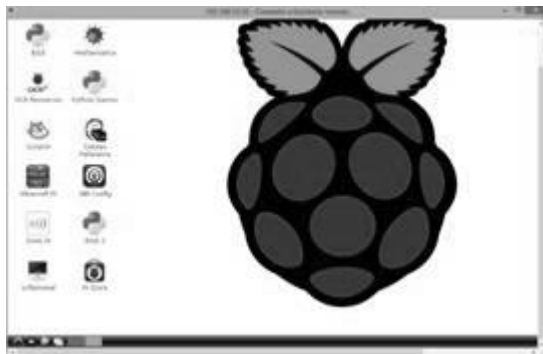


Figura 5. Modo gráfico

Otro punto importante de configuración es la instalación de un escritorio remoto para el caso de trabajar con la Raspberry sin la necesidad de un monitor HDMI, para ello existen componentes como XRDP, TightVNCServer, etc.

Otros comandos importantes son los comandos de actualización de repositorios, programas y Kernel de la Raspberry, que de igual forma se instalan desde la terminal de la Raspberry.

```
sudo apt-get update
sudo apt-get upgrade
sudo rpi-update
```

## 1.4 Internet de las cosas IoT

La IoT es lo que obtenemos cuando conectamos cosas que no son operadas por los seres humanos a Internet.

Este visión toma en cuenta que si los acontecimientos que suceden en el mundo real son recabados por el ser humano se tiende a tener errores además de ser muy lentos [5].

### 1.4.1 Plataformas IoT

En la actualidad existen diferentes proveedores que ofrecen servicios de IoT, algunos de ellos solo se pueden crear cuentas con algún costo y en algunas otras se pueden crear cuentas para desarrolladores. Algunos ejemplos de estos proveedores son:

- Smartliving
- Xively
- ThingSpeak

-Dataplicity

### 1.4.2 Conexión de dispositivos a IoT

Para todos los proveedores que se mencionaron con anterioridad se coincide mucho en la forma de agregar un dispositivo a IoT, enseguida se resumen los pasos a seguir:

1) Crear una cuenta gratuita, para esto, se debe de ir a la sección de desarrolladores, en donde los datos a pedir son usuario, e-mail, contraseña, motivo de proyecto, nombre completo, país, código postal, etc.

Una vez la cuenta ha sido creada se debe de confirmar mediante un link enviado a correo electrónico e iniciar sesión, donde se nos mostrara la información necesaria para agregar dispositivos, en nuestro caso el dispositivo con el que se trabajara será Raspberry.

2) Una vez que los dispositivos son agregados se entrega información del dispositivo privado, como Product ID, Product Secret, Feed ID, Feed URL, etc. Estos datos son insertados en el programa de la tarjeta de desarrollo que se tenga.

## 2. DESARROLLO

El desarrollo del proyecto abarca diferentes etapas para su implementación (Figura 6), como los son la programación de la Raspberry para soportar IoT, creación de una cuenta IoT, etapa de potencia y manejo de sensores y actuadores.



Figura 6. Esquema general del prototipo.

### 2.1 Programación de GPIO

Una de las características más importante de la Raspberry es el manejo de los puertos GPIO, para el proyecto se contemplan entradas y salidas digitales, cabe mencionar que antes de implementar IoT en la Raspberry primero se manipularon tanto entradas digitales como salidas digitales, todo ello, empleando Python.

### 2.1.1 Entradas y salidas digitales

Como se mencionó anteriormente los pines de entrada de la Raspberry solo soportan hasta 3.3 V, también se requiere definir si el pin de entrada va estar en “pull up” es decir, si va a detectar que este en nivel alto o en bajo.

Para utilizar los GPIO se requiere importar en Python la siguiente librería.

```
import RPi.GPIO as GPIO
```

Enseguida se muestra el código en Python para controlar el estado de una salida digital, la cual, depende de si esta pulsado o no un boton momentáneo.

```
#Entradas y salidas digitales
import RPi.GPIO as GPIO
import time
boton=24

GPIO.setmode(GPIO.BCM)
GPIO.setup(23,GPIO.OUT)
GPIO.setup(boton,GPIO.IN,GPIO.PUD_UP)

while True:
    if GPIO.input(boton)==False:
        GPIO.output(23,True)
        print "boton presionado"
        time.sleep(5)
    else:
        GPIO.output(23,False)
        print "boton sin presionar"
        time.sleep(1)
GPIO.cleanup()
```

En la Figura 7 se muestra el esquema de conexión básico de cuatro salidas digitales monitoreadas con Leds y cuatro entradas digitales con push-button, este esquema de conexión fue el que se empleo tanto para comprobar el funcionamiento de los pines GPIO como la implementación de IoT en la Raspberry.

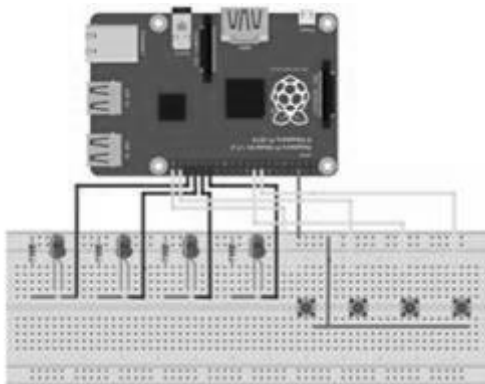


Figura 7. Esquema de conexión de entradas y salidas digitales.

### 2.2 Conexión de Raspberry a Smartliving

Una vez que se ha creado una cuenta en Smartliving se procede a agregar un dispositivo. En esta plataforma se pueden agregar dispositivos de hardware Arduino y Raspberry.

Después de agregar el dispositivo nos muestra la pantalla que resume la configuración en tres breves pasos (Figura 8).



Figura 8. Configuración del dispositivo en IoT.

El paso uno indica que hay que descargar un archivo .rar que contiene las librerías y algunos ejemplos para hacer conexión con la interfaz de SmartLiving. Después de descargar el archivo, se tiene que pasar a la tarjeta SD de la Raspberry Pi por cualquier método.

En la terminal de la tarjeta Raspberry Pi es necesario situarse en el directorio donde se encuentra el .rar descargado, después se descomprime el archivo y se procede a modificar un archivo de Python llamado Shield\_Demo.py el cual contiene una serie de instrucciones para realizar la conexión con Smartliving y tomar las lecturas de GPIO.

En el paso dos se proporciona el DeviceID, ClientID y el ClientKey los cuales deberán ser agregados al archivo .py de Python en la Raspberry.

Por último en el paso 3 se encuentra la documentación de IoT para el dispositivo configurado.

### 2.3 Descripción de la librería IoT en Python

En el archivo .py que se ejecuta en la Raspberry para la implementación de IoT se deben insertar el DeviceID, ClientID y el ClientKey además de configurar los canales que se crearan en la interfaz de Smartliving, en la Tabla 2 se muestra la configuración de estos canales.

Tabla 2. Configuración de canales en Python

Tipo de canal IoT	Configuración
Entrada digital	IOT.addAsset(sensorPin, sensorName, "Push button", False, "bool")
Salida digital	IOT.addAsset(actuatorPin, actuatorName, "LED", True, "bool")

### 2.4 Funcionamiento del controlador de riego.

El controlador de riego que se tiene es de cuatro canales, es decir, se pueden controlar hasta cuatro electroválvulas, además se tienen cuatro sensores de humedad de suelo que envían señales On-Off dependiendo de si sobrepasa cierto porcentaje de humedad. Cabe señalar que por el momento el control es solo On-Off incluso no tiene programación de



temporizaciones las cuales son unas mejoras que se tienen como trabajo futuro.

Como etapa de potencia para el encendido de electroválvulas se tienen una etapa de MOC y TRIAC empleadas en el *Diseño e implementación de un controlador de riego con reloj de tiempo real para un invernadero* (Figura 9)[6]

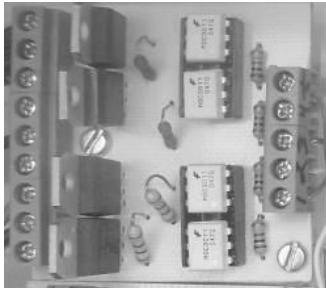


Figura 9. Etapa de potencia

### 3. RESULTADOS

Como se ha observado una de las ventajas que se tiene en IoT es que no se requiere crear una página y servidor Web, como se hacía con anterioridad pero esto trae como consecuencia el que no se puedan personalizar las interfaces. En la Figura 10 se muestra un canal digital de salida creado en Smarliving, aquí, el botón se muestra habilitado.

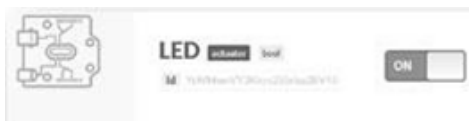


Figura 10. Canal digital de salida

En la Figura 11 se muestra un canal digital de entrada, el botón se muestra deshabilitado y solamente muestra un True o False dependiendo de su estado.

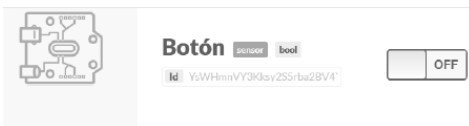


Figura 11. Canal digital de entrada.

Otra de las ventajas de Smarliving es que cuenta con aplicación para dispositivos Android, donde básicamente es lo mismo que una computadora de escritorio, solo que con la aplicación se tiene acceso directo a Smarliving donde los diseños de interfaz son muy similares.

Como se comento con anterioridad el control que se tiene es simplemente On-Off, pero cabe resaltar que ha resultado de gran ayuda para los operadores el poder encender el riego o fertiriego desde su casa.

Como trabajo futuro se pretende agregar la lectura de sensores analógicos para esto se empleara un convertidor analógico-digital con interfaz SPI, el modelo es de la familia Microchip MCP3008 (Figura 12).

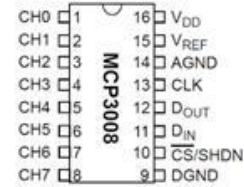


Figura 12. Convertidor analógico-digital SPI.

Con ello la funcionalidad del controlador de riego se expandiría ya que no solo controlaría el riego si no también serviría como una pequeña estación meteorológica para el invernadero.

En las Figuras 13 y 14 se despliega las pantallas de visualización de canales analógicos en Smarliving, como se puede observar se pueden conservar históricos para tomar decisiones con conocimiento de causa, es decir al contar con gráficas de temperatura y humedad por ciclo de cultivo, por semana, por mes o incluso por año, se tendrá información que permite tomar decisiones en el manejo del invernadero.



Figura 13. Canal analógico.

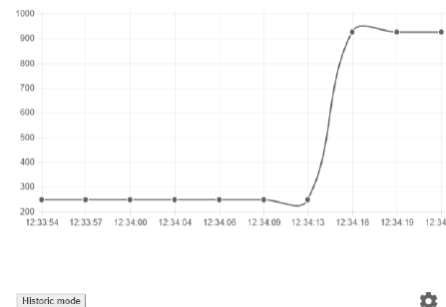


Figura 14. Canal analógico con historial.

En la Tabla 3 se muestran las pruebas realizadas con el controlador de riego, como se puede observar se hicieron pruebas hasta con 4 canales IoT para sensor y 4 canales para actuador. Se observó que el funcionamiento depende mucho del tipo de conexión a Internet, si se tiene muy poco ancho de banda los canales tardan en responder tanto para lectura y escritura de datos.

Como trabajo futuro se pretende realizar un diseño de circuito impreso (PCB) para conectar directamente a la tarjeta Raspberry sensores y actuadores. Además se pretende que cuente con una pantalla táctil para implementar un modo manual de riego.

Tabla 3. Pruebas realizadas al controlador de riego

<b>Canales IoT p/sensor</b>	<b>Canales IoT p/actuador</b>	<b>Tipo de sensor</b>	<b>Actuador/Cantidad</b>	<b>Conectividad y velocidad de la conexión</b>	<b>Funcionamiento</b>
1	1	Humedad de suelo/1	Electroválvula/1	Ethernet/5Mb	-Poco estable -Retardo en la respuesta del canal
2	2	Humedad de suelo/2	Electroválvula/2	WiFi/10Mb	-Estable -Sin retardo en la respuesta del canal
4	4	Humedad de suelo/2 Encendido/2	Electroválvula/2 Motoreductores/2	WiFi/10Mb	Estable -Ligero retardo en la respuesta del canal

#### 4. CONCLUSIONES

El objetivo primordial se alcanzó al implementar un controlador de riego con conectividad IoT que fuera fácil su manejo e instalación por parte de un productor de hortalizas bajo un ambiente protegido.

El controlador permite controlar hasta cuatro diferentes electroválvulas y conocer el estado de cuatro sensores de humedad o de activación, cabe resaltar que esta configuración según las pruebas realizadas es estable y cumple con los requerimientos del Invernadero instalado en la Universidad; actualmente se realizan pruebas para aumentar tanto el número de sensores como de actuadores.

Además el desarrollo de este prototipo permitió conocer el funcionamiento y configuración de diversas plataformas IoT. También con la realización de este proyecto se generan múltiples ideas para la automatización de un invernadero empleando IoT.

#### 5. REFERENCIAS

- [1]<http://connect.iobridge.com/docs/>
- [2] Manual de producción de tomate en invernadero Javier Z. Castellanos, Intagri, 2009.
- [3]<https://www.broadcom.com/products/BCM2835>
- [4]<https://www.raspberrypi.org/documentation/usage/gpio/>
- [5] Learning Internet of Things/Peter Waher/Enero 2015/ISBN 978-1-78355-353-2
- [6] Diseño e implementación de un controlador de riego con reloj de tiempo real para un invernadero/Perla Alejandra Herrera Castillo/2012