



UNIVERSIDAD NACIONAL DE LA MATANZA

DEPTO. DE INGENIERIA E INVESTIGACIONES TECNOLOGICAS

Internet of Things

Sistemas Embebidos + Android

“SeedUp”

Asignatura: Sistemas Operativos Avanzados

Comisión: 1900

Integrantes:

Alvarez, Lautaro

DNI: 38.056.129

Amado, Brian

DNI: 32.071.824

Barros, Lucas

DNI: 30.651.158

Rizza, Nicolás

DNI: 30.340.422

Fecha	Versión	Devolución
19/06/17	1.0	

Primer Cuatrimestre 2017

<i>Índice</i>	
Objetivo.....	2
Descripción del entorno de desarrollo.....	3
Hardware.....	3
Software.....	3
Alcance del sistema (SE + Android).....	4
Sistema embebido.....	4
Sistema Android.....	4
Diseño.....	5
Implementación.....	8
Sensor de Luz.....	8
Sensor de Humedad.....	9
Capturas de sistema.....	10
Links de interés.....	12

<i>Objetivo</i>

El objetivo del trabajo práctico del cual es objeto el informe en el presente documento tiene como finalidad la implementación de un sistema de control para el crecimiento de una planta (seedUP) pudiendo mantener las propiedades necesarias para su correcto desarrollo. El sistema consta de un sistema embebido, desarrollado con código en Arduino, para el correcto funcionamiento de la placa base con los sensores y actuadores implementados, y una aplicación para dispositivos móviles desarrollada en Android para poder monitorear y controlar el sistema manualmente.

<i>Descripción del entorno de desarrollo</i>
--

Hardware:

- Sistema Embebido:
 - HP Pavilion dm4-3055dx
 - Placa Intel Galileo Gen 2
 - Protoboard de 400 puntos
 - Sensor analógico de humedad YL-69 (con módulo YL-38)
 - Sensor digital de luz BH1750
 - Módulo bluetooth HC-06
 - Módulo Relé de dos canales
 - Bomba de agua [BL-200]
 - Lámpara de luz
- Sistema Android:
 - Razer Blade Stealth 2016 (Intel i7 - 16GB RAM - Windows 10)
 - Samsung Galaxy S8 - Android 7.0

Software:

- Sistema Embebido:
 - Arduino IDE 1.8.2
- Sistema Android:
 - Android Studio 2.3.3 para Windows
 - Java 8
 - Android SDK 25
 - Apache Gradle

<i>Alcance del Sistema (SE + Android)</i>
--

Sistema Embebido:

1. El SE debe medir la luminosidad del entorno.
2. El SE debe medir la humedad de la tierra de la planta.
3. El SE debe implementar rangos de luminosidad para poder distinguir la luz en el ambiente entre alto y bajo.
4. El SE debe implementar rangos de humedad para poder distinguir la humedad de la tierra de la planta entre los niveles seco o húmedo.
5. El SE debe accionar la bomba de agua para el riego de la planta cuando el nivel de la humedad en la tierra se encuentra en estado seco.
6. El SE debe accionar la luz artificial (encender la lámpara de luz) cuando el nivel de luz del ambiente esté por debajo de 15 lux.
7. El SE debe enviar el estado de los sensores a través de bluetooth a la aplicación android para poder sincronizar y anunciar el estado del sistema.
8. El SE debe poder recibir a través de bluetooth las indicaciones de la aplicación android para poder iniciar algún actuador (control de forma manual).

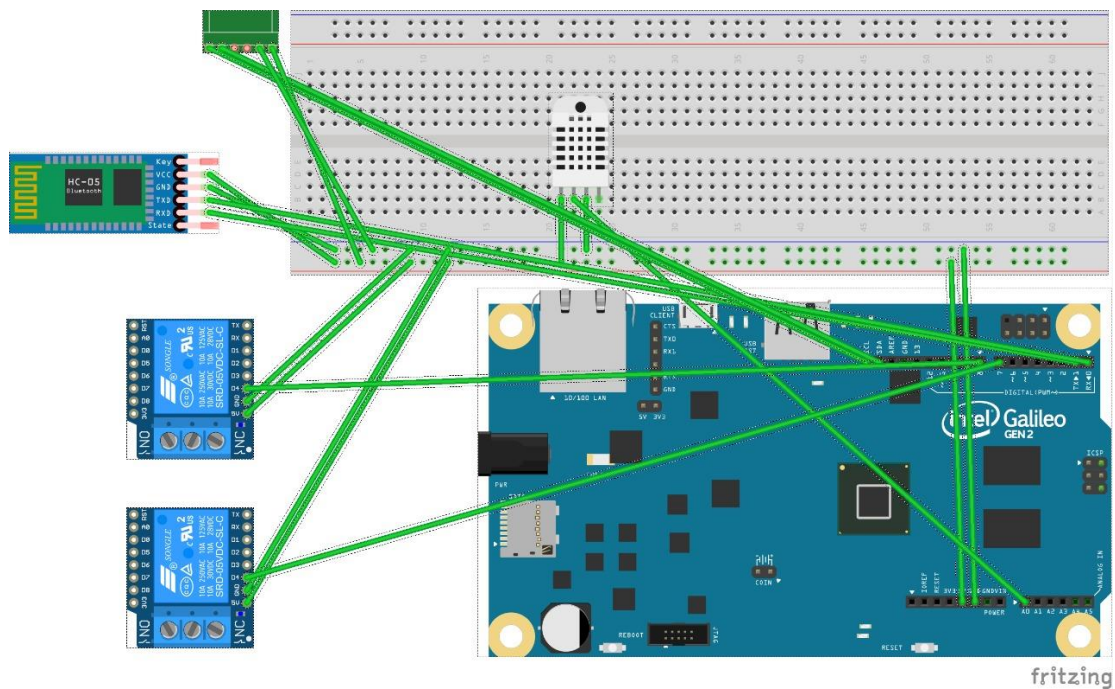
Sistema Android:

1. La aplicación Android (AA) debe poder conectarse por bluetooth al SE (por el módulo HC-06).
2. La AA debe consultar a través de bluetooth el estado del sensor de humedad para poder mostrarlos en la interfaz.
3. La AA debe consultar a través de bluetooth el estado del sensor de Luz para poder mostrarlos en la interfaz.
4. La AA debe poder ejecutar el actuador de luz a través del modo manual.
5. La AA debe poder ejecutar el actuador de riego a través del modo manual.

Diseño

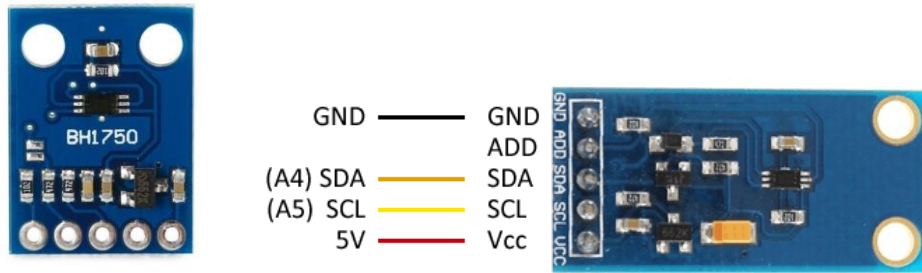
Para poder realizar todas las interconexiones de los diferentes módulos de sensores, y los respectivos actuadores que forman parte del sistema, se realizó brevemente una lectura de cada uno de los componentes para entender las necesidades técnicas y circuitales necesarias para una correcta ejecución, principalmente de la placa Intel Galileo, base para el prototipo.

A continuación podemos ver un diagrama el cual hace referencia a las conexiones entre dispositivos:



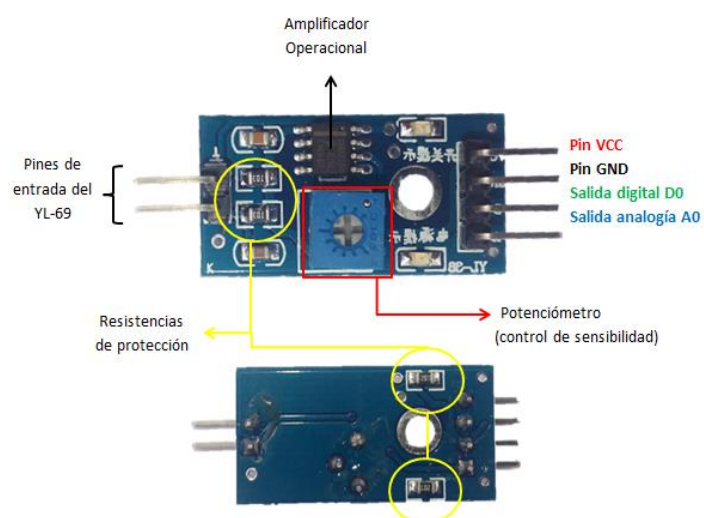
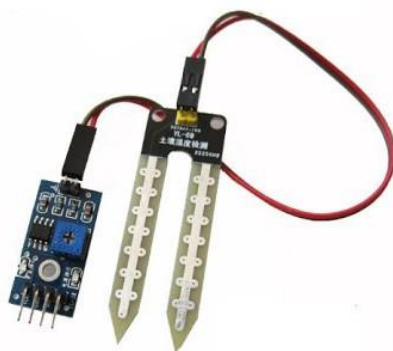
El sistema tiene dos importantes sensores, el de luminosidad y el de humedad. Aquel encargado de controlar la iluminación del ambiente, es el modelo BH1750, el cual es un sensor digital de intensidad de luz ambiente, con un conversor ADC de 16bits interno. Esta es una versión mejorada del típico sensor de luz a base de un LDR, el cual simplemente entrega un valor analógico. El BH1750, nos entrega automáticamente el valor En Lux (desde 1 lx hasta 65535 lx), y se comunica por I2C, con la posibilidad de seleccionar 2 Address. Este sensor nos permite medir la cantidad de luz por metro cuadrado que tenemos en algún ambiente.

Informe IoT - Sistema "SeedUp"



El otro sensor, es el encargado de la medición de la humedad de la tierra donde crece la planta, el YL-69. Aplicando una pequeña tensión entre los terminales del módulo YL-69 hace pasar una corriente que depende básicamente de la resistencia que se genera en el suelo y ésta depende mucho de la humedad. Por lo tanto al aumentar la humedad la corriente crece y al bajar la corriente disminuye.

Consiste en una sonda YL-69 con dos terminales separados adecuadamente y un módulo YL-38 que contiene un circuito comparador LM393 SMD (de soldado superficial) muy estable, un led de encendido y otro de activación de salida digital. Este último presenta 2 pines de conexión hacia el módulo YL-69, 2 pines para la alimentación y 2 pines de datos. VCC, GND, D0, A0.



Informe IoT - Sistema "SeedUp"

Como actuadores, tenemos una bomba de agua, instalada para el riego y generación de la humedad necesaria para el crecimiento de la planta y por otro lado una lámpara de luz, necesaria para el correcto proceso de fotosíntesis. Ambos actuadores reciben 220W de corriente por lo que están conectados a un relé de dos canales el cual permite el paso de corriente de manera conveniente de acuerdo a la configuración del sistema.



Luego, la bomba de agua, actuador de riego, es un modelo que encontramos originalmente implementado en peceras. El modelo exacto es PUMP BL 200.



El otro actuador es una bombilla de luz de bajo consumo (con poca generación de calor para no modificar el entorno) conectada con un porta lámpara a 220W.



Implementación

Sensor de luz:

El BH1750 es un sensor de iluminación ambiental. Se comunica de forma digital con el microcontrolador, con el bus I2C. Su tiempo de respuesta es bastante bajo, menos de 200 ms en las circunstancias más desfavorables.

Desde el punto de vista electrónico su implementación es sencilla, sólo tuvimos que conectar la alimentación y el bus I2C. Opcionalmente puede cambiarse la dirección del bus para conectar dos dispositivos en el mismo bus (con ADDR a nivel bajo es 0B0100011 o 0x23 y con ADDR a nivel alto es 0B1011100 o 0x5C).

El BH1750 tiene dos modos de lectura, continua e individual, que corresponden con dos estados, activo y de bajo consumo o reposo. Mientras que si se utiliza el modo de lectura continua el BH1750 sigue activo tras el muestreo, después de realizar una medición individual entra automáticamente en el modo de reposo y bajo consumo.

Para medir la iluminación ambiental con el BH1750 utilizamos la librería Wire que gestiona las comunicaciones con el bus I2C. Como decidimos utilizar un modo de lectura continua con resolución de 1 lux, fue necesario usar el `Wire.beginTransaction()` y el `Wire.endTransmission()` entre el `Wire.read()`. De haber utilizado un modo de lectura individual esto no hubiera sido necesario, pero decidimos utilizar el modo de lectura continua ya que íbamos a estar constantemente consultando este valor. Además, si se utiliza el modo de lecturas individuales el BH1750 entra en modo de reposo luego de una lectura y para volver al modo activo tendríamos que enviar una configuración (mismo modo de lectura que veníamos usando o uno nuevo) o el código de encendido (0x01).

El sensor es capaz de medir a intervalos (resolución) de 4 lux, 1 lux y 0,5 lux. El BH1750 recomienda en su hoja de datos utilizar la resolución de 1 lux, que permite distinguir iluminaciones por debajo de los 10 lux (que corresponde con la luz crepuscular) y que es más inmune al ruido que podría afectar a la medida. Las resoluciones de 1 lux y 4 lux utilizan los 16 bits de datos para representar la parte entera por lo que se puede alcanzar una medida máxima de 65535 lux (día soleado sin luz directa). El modo de 0,5 lux usa el bit menos significativo para la parte decimal (mide de 0,5 lux en 0,5 lux) por lo que con los 15 bits restante se puede conseguir representar un valor máximo de 32767 lux (exterior sin luz directa).

Considerando que hay que realizar dos lecturas I2C de un byte para obtener el valor de 16 bits, hay que cargar los bits del byte más significativo (los primeros que se leen), rotarlos 8 bits a la izquierda y luego concatenarlos a los del segundo byte. La lectura se hace de la siguiente forma:

Informe IoT - Sistema "SeedUp"

1. Obtenemos el primer byte. Supongamos que es 11100111.
2. Obtenemos el segundo byte. Supongamos que es 00011000.
3. Concatenamos ambos bytes en un único int:
 - a. Hacemos un shift del primer byte en 8 lugares. Nos quedaría
1110011100000000.
 - b. Hacemos un OR lógico entre ese y el segundo byte. Nos quedaría
1110011100011000.
4. Al número obtenido anteriormente lo dividimos por 1.2 (factor de precisión)

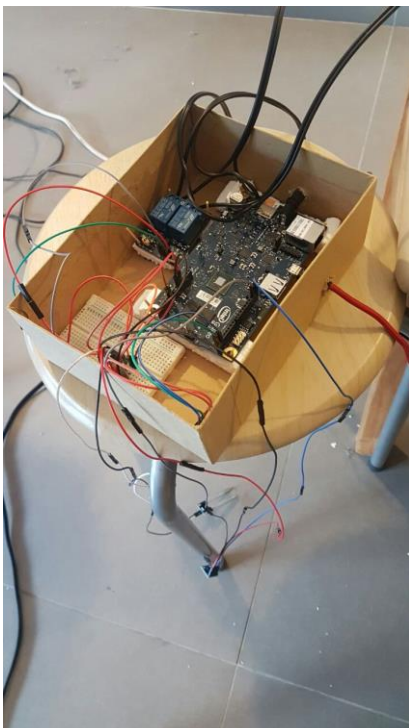
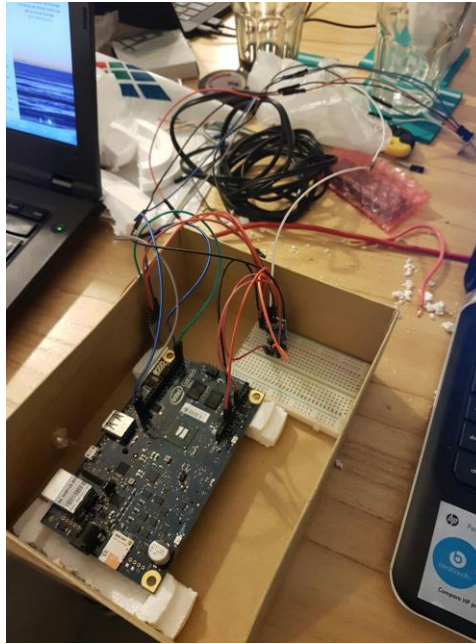
De esta forma obtenemos el número final en lux.

Sensor de humedad:

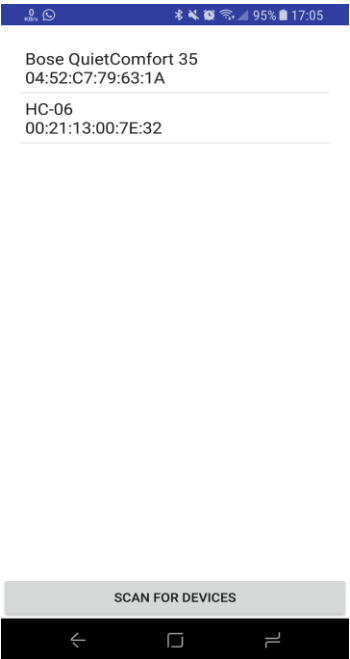
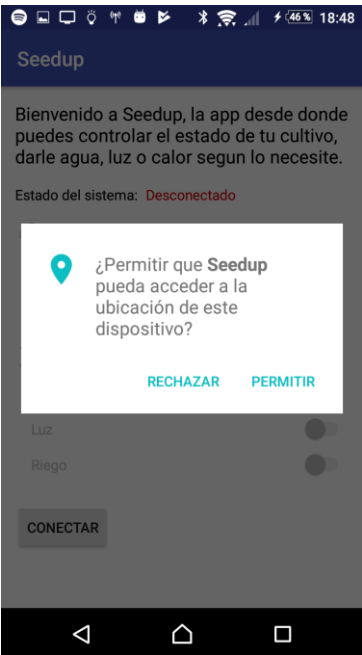
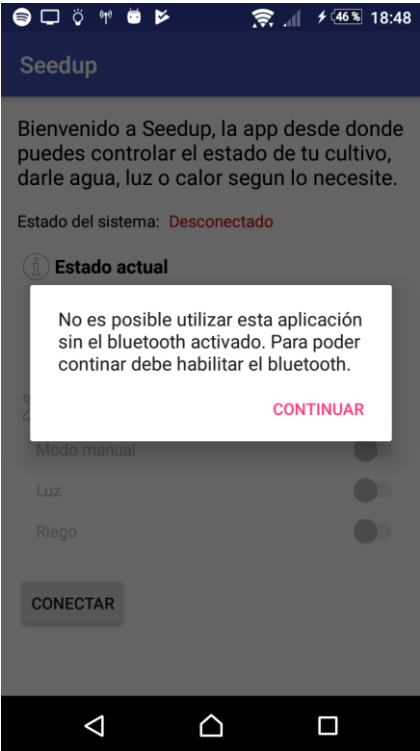
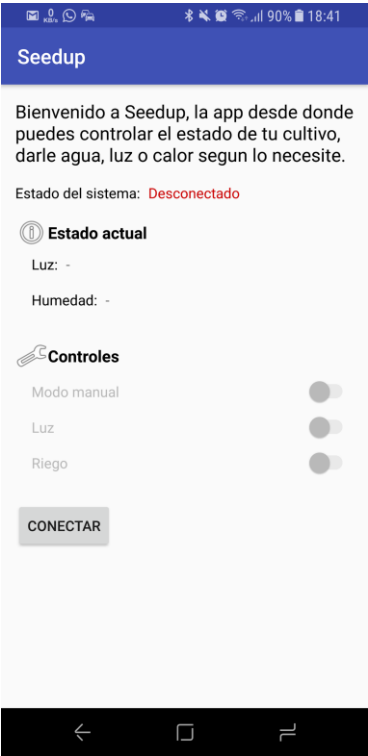
El sensor de humedad utilizado fue el YL-69 en su modo analógico. Los valores entregados por este sensor se encuentran en el rango de 0-1023, siendo 0 cuando no hay diferencia de potencial entre los terminales (es decir, el suelo está tan mojado que es como si ambos terminales estuvieran conectados entre si) y 1023 cuando la diferencia de potencial es máxima (el suelo está completamente seco y no hay paso de corriente entre los terminales).

Teniendo eso en cuenta necesitábamos establecer una relación entre 0 y 1023, con el 100% y el 0% de humedad. Para ello utilizamos la función `map()` de Arduino que lo que haces es justamente remapear el número de un determinado rango al que le correspondería en otro.

Capturas del Sistema



Informe IoT - Sistema “SeedUp”



<i>Links de interés</i>

Foro de cátedra SOA UNLAM Experiencias IoT:

<http://so-unlam.com.ar/moodle/mod/forum/view.php?id=52>

Proyecto en GIT de seedUP:

<https://github.com/lukasbloom/SOA-Seedup>

Android Studio:

<https://developer.android.com/studio/index.html>

Java:

<https://www.java.com/es/download/>

Linux Yocto:

<https://www.yoctoproject.org/>

IDE Arduino:

<https://www.arduino.cc/en/main/software>