|  |  |  |
| --- | --- | --- |
|  | **UNIVERSIDAD DE LOS ANDES**  **FACULTAD DE INGENIERÍA**  **DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN**  **INFRAESTRUCTURA DE COMUNICACIONES**  **Profesor:**  **Yezid Donoso**  [**ydonoso@uniandes.edu.co**](mailto:ydonoso@uniandes.edu.co)  **Santiago Rojas**  [**s.rojas963@uniandes.edu.co**](mailto:s.rojas963@uniandes.edu.co) |  |

|  |  |  |
| --- | --- | --- |
| |  | | --- | | **LABORATORIO – ARDUINO GALILEO Y UBIDOTS** | |  | |

**1. OBJETIVO (S)**

Utilizar una tarjeta Arduino como mecanismo para recolectar información de diversos sensores análogos y digitales.

Utilizar una Arduino Galileo para transmitir información hacia la nube.

Realizar aplicaciones prácticas de medición utilizando sensores de Temperatura y detección de movimiento.

Conocer el funcionamiento de la plataforma Ubidots para recolectar y analizar grandes cantidades de datos.

**2. LECTURAS PREVIAS**

1. Arduino, Intel Galileo [10], Getting started with Galileo [12], Getting started, Ubidots [14]

**3. INFORMACIÓN BÁSICA**

A lo largo del laboratorio se observará la forma de trabajar con información de sensores, su obtención, transmisión, compilación, análisis y consulta. Se tomarán dos valores de temperatura simultáneos y un indicador de movimiento en la habitación. La información capturada será enviada a la plataforma Ubidots gracias a la fácil conectividad de las tarjetas Arduino a través de su puerto Ethernet.

**4. MARCO TEÓRICO**

Los sensores son dispositivos que reciben una señal o estimulo físico y lo convierten en una señal eléctrica. El estímulo es la cantidad, propiedad o condición que es censada y convertida en una señal eléctrica. La señal de la salida del sensor puede ser voltaje, corriente o carga.

Entre algunas de las variables físicas que puede medir un sensor se tienen: temperatura, humedad, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, movimiento, calidad del aire, calidad del agua, ruido, entre muchos otras.

Un sensor ideal es aquel instrumento que no altera las propiedades y valores de las medidas tomadas.

Ejemplo, “en una medición de temperatura el sensor ideal no debería aportar ni recibir calor, es decir debería tener una masa igual a cero o no debería estar en contacto con la masa a la que se le está midiendo la temperatura”.

**Características de un sensor**

* Rango de medida: dominio en la magnitud medida en el que puede aplicarse el sensor.
* Precisión: es el error de medida máximo esperado.
* Offset o desviación de cero: valor de la variable de salida cuando la variable de entrada es nula. Si el rango de medida no llega a valores nulos de la variable de entrada, habitualmente se establece otro punto de referencia para definir el offset.
* Linealidad o correlación lineal.
* Resolución: mínima variación de la magnitud de entrada que puede detectarse a la salida.
* Rapidez de respuesta: puede ser un tiempo fijo o depender de cuánto varíe la magnitud a medir. Depende de la capacidad del sistema para seguir las variaciones de la magnitud de entrada.
* Derivas: son otras magnitudes, aparte de la medida como magnitud de entrada, que influyen en la variable de salida. Por ejemplo, pueden ser condiciones ambientales, como la humedad, la temperatura u otras como el envejecimiento (oxidación, desgaste, etc.) del sensor.
* Repetitividad: error esperado al repetir varias veces la misma medida.

El valor de salida de un sensor es proporcional al estímulo recibido por el sensor en la entrada. Sin embargo hay sensores que se conocen como sensores ON/OFF, donde el valor salida es un cambio de estado generado cuándo la variable censada rebasa un cierto umbral. El cambio de estado puede ser encendido o apagado, o un estado lógico de uno o cero.

**Tipos de sensores**

* **Sensores Activos**, Sensores que generan señales representativas de las magnitudes a medir en forma autónoma, sin requerir de fuente alguna de energía.
* **Sensores Pasivos**, Son aquellos que generan señales representativas de las magnitudes a medir por intermedio de una fuente energía auxiliar.

Los sensores pueden entregar los datos de salidas, representándolos como una señal digital o una señal analógica:

* **Sensores Análogos**, Los sensores analógicos que miden la información continua en el tiempo, es decir para todo tiempo T, el sensor mide la entrada y entrega un valor de salida.
* **Sensores Digitales**, Los sensores digitales son aquellos que frente a un estímulo de entrada, su salida es representada por un cambia de estado lógico ya sea de cero a uno o de uno a cero o una secuencia de bits que representan el valor del estímulo. Utilizan valores discretos para representar la información para la entrada.

**ADVERTENCIA**: Resulta especialmente importante diferenciar los sensores análogos de los digitales para utilizarlos con una tarjeta Arduino. Esta cuenta con pines de lectura análogos y pines de lectura/escritura digitales. Si se conecta un sensor con una salida análoga a un pin digital, éste dañará el pin.

**Sensores de Temperatura**

Los sensores de temperatura son dispositivos capaces de transformar los cambios de temperatura de un medio en cambios en señales eléctricas que son procesados por un equipo eléctrico o electrónico. Existen diversos tipos de sensores de temperatura, para este laboratorio se utilizarán los sensores de tipo monolítico.

**Sensores de temperatura Monolíticos**

Las principales ventajas de estos dispositivos es que son económicos, pequeños y lineales. Su salida puede ser voltaje, corriente y datos digitales para los más complejos. La principal desventaja es que el rango de temperatura no supera los 200 °C.

Un ejemplo de este tipo de sensores son los LM35 (Sensor de Temperatura Celsius); es un dispositivo de tres terminales que produce un voltaje de salida de 10 mV/°C, de modo que el voltaje nominal de salida es de 250MV a 25°C y un 1V a 100 °C. La precisión del LM35 es de más o menos 1°C desde -55°C a +150°C.

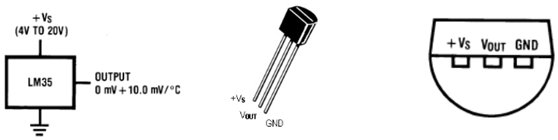


Ilustración 1. Sensor LM35

El LM35 no requiere de circuitos adicionales para calibrarlo externamente. La baja impedancia de salida, su salida lineal y su precisa calibración hace posible que sea instalado fácilmente en un circuito de control. Debido a su baja corriente de alimentación se produce un efecto de auto calentamiento muy reducido.

**Sensores de Movimiento**

Un sensor pasivo de infrarrojo (PIR por sus siglas en inglés) es un sensor electrónico que mide la luz infrarroja (IR) que irradia de los objetos en su campo de visión. Estos son los más utilizados para la detección de movimiento.

Todos los objetos con una temperatura por encima del cero absoluto emiten energía de calor en forma de radiación. Por lo general, esta radiación es invisible para el ojo humano, ya que irradia en longitudes de onda infrarrojas, pero puede ser detectado por dispositivos electrónicos diseñados para tal propósito. El término pasivo en este caso se refiere al hecho de que los dispositivos PIR no generan ni irradian energía para fines de detección. Trabajan en su totalidad por la detección de la energía emitida por otros objetos. Los sensores PIR no detectan o miden calor; en su lugar detectan la radiación infrarroja emitida o reflejada de un objeto.

**5. COMPONENTES REQUERIDOS**

Para el desarrollo completo de este laboratorio, se necesitarán:

1. Tarjeta Arduino Galileo (gen 1)
2. Protoboard
3. 11 Cables de cobre
4. 2 sensores LM35 de temperatura
5. 1 sensor PIR
6. Cable Ethernet
7. Cable USB – microUSB

**6. PROCEDIMIENTO**

Se debe montar el circuito de manera tal que el pin de 5V de la tarjeta alimente simultáneamente a los 3 sensores, a su vez, los tres estarán conectados al mismo polo a tierra. La salida de los sensores de temperatura estará unida a los pines A0 y A2 y por su parte el sensor de movimiento al pin D2. Es importante fijarse de conectar los sensores a los pines correctos, o podrá dañar a estos o a la tarjeta.

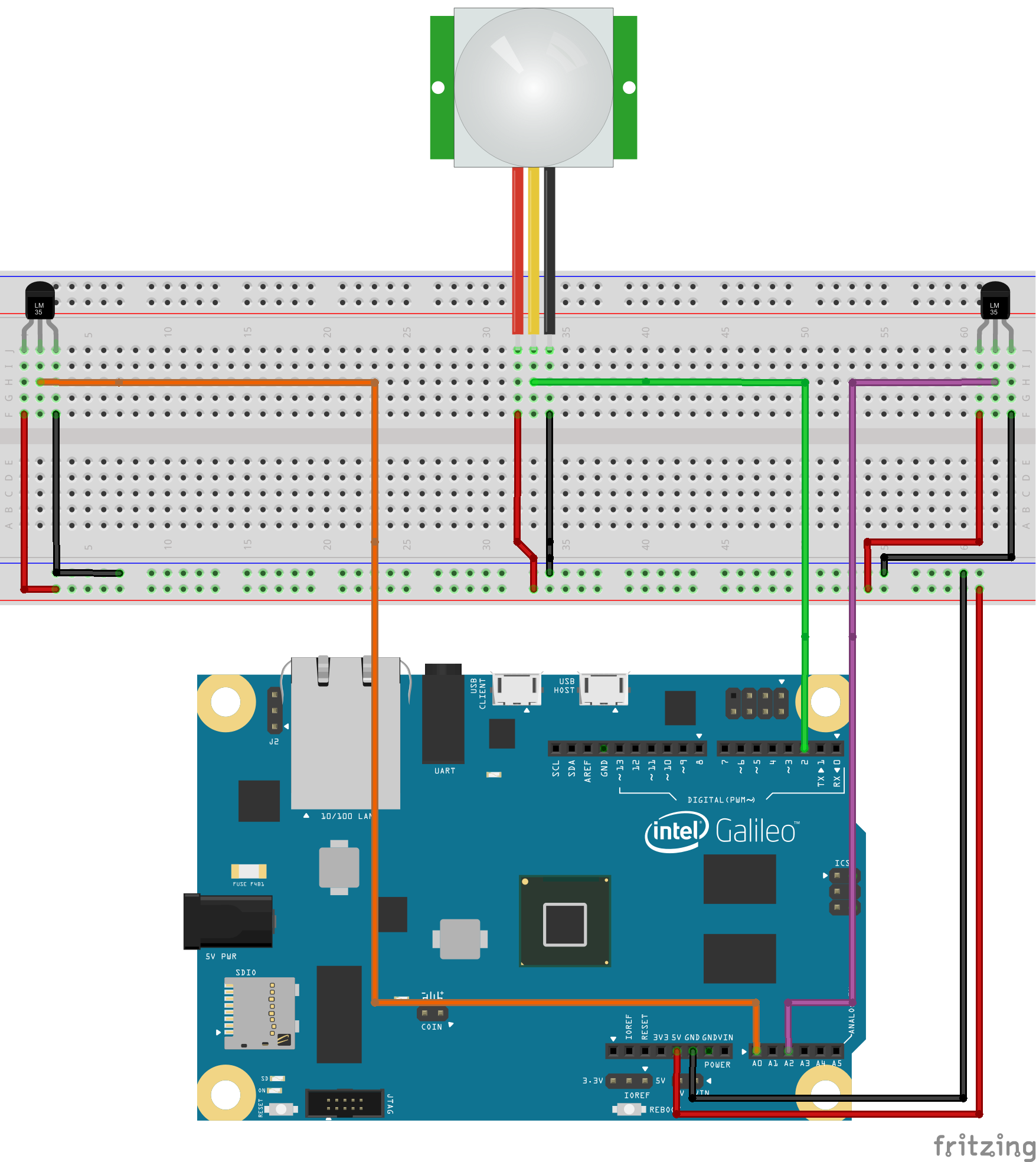


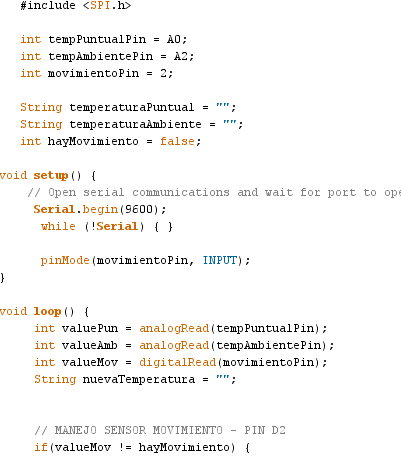
Ilustración 2 Arquitectura del circuito

**6.1. LECTURA DE UN SENSOR DE TEMPERATURA LM35**

En esta parte usted utilizará dos sensores LM35 como medidores de temperatura que entrega una medida de voltaje en un puerto analógico de la placa Arduino Galileo. Este valor censado está dado en unidades de voltaje por lo que es necesario realizar una conversión a temperatura y finalmente presentar los datos censados a través del monitor serial de la placa.

Abra el IDE de Arduino. Conecte la placa Arduino Galileo a su equipo, verifique cuál es el puerto serial asociado a este y configúrelo en el entorno de desarrollo de ser necesario. Se recomienda utilizar la versión 1.5.3 del IDE dada por Intel para conectarse correctamente a la tarjeta. Escriba el siguiente código para el sketch:

Se utilizan funciones auxiliares para calcular el valor en grados centígrados dado el voltaje de entrada.





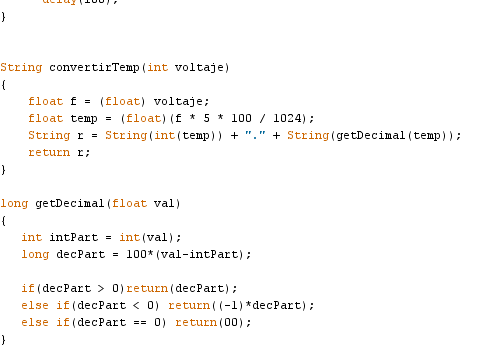


Ilustración 3. Código fuente actividad 1

Como puede observar en el código, el pin análogo seleccionado es el pin analógico cero (0). En este pin cero, se conectó la salida del sensor de temperatura LM35. Por medio de la función analogRead se lee el valor de voltaje asociado entregado en el pin análogo por el sensor. Esta medida de voltaje es un valor representado en la escala de cuantización del conversor análogo digital de la placa.

Es de recordar que el sensor análogo digital de Arduino, tiene una resolución de 10 bits, por lo tanto para una medida de voltaje de 5 voltios, el sistema representara el valor como 1024 unidades; un valor de cero voltios lo representara como 0 unidades y para un valor de 2.5 voltios se tendrá una representación de 512 unidades.

La expresión **temperatura = (5 \* temperatura \* 100) / 1024**; permite convertir la representación del valor de temperatura en unidades a temperatura en grados centígrados. Recordando que el sensor LM35 entrega un voltaje de 1 voltio a una temperatura de 100 °C y un voltaje de 250 mili-voltios a 25 °C podemos validar la expresión de la siguiente forma:

Primer paso convertir la expresión en voltios a unidades cuantizadas por el conversor análogo digital.

**6.2. Usando el API de Ubidots**

Cree una cuenta gratuita dentro de la plataforma, a continuación, cree un token para acceder al API desde Arduino.

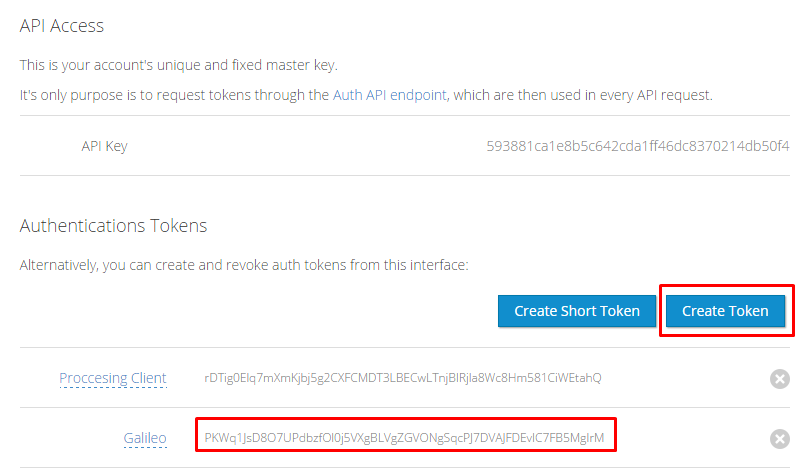


Ilustración 4 Crear un token de acceso a Ubidots

Cree una fuente de datos, que en este caso es nuestra tarjeta Galileo:

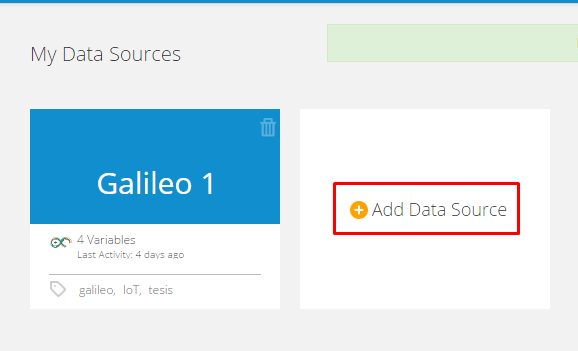
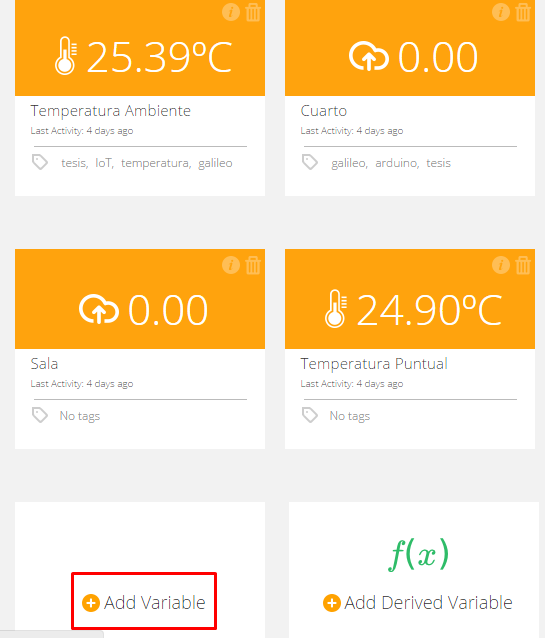
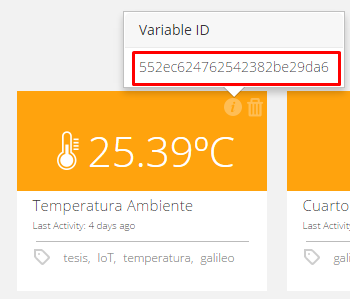


Ilustración 5 Crear una nueva fuente de datos

Y agregue suficientes variables a medir: dos de temperaturas y un movimiento, asignando nombres y unidades de medición en ºC para el caso de la temperatura. No es necesario definir una unidad para la variable de movimiento.



Después de esto, debemos anotar el id de variable asignado por Ubidots, este, en conjunto con el token será utilizado para enviar la información a la plataforma.



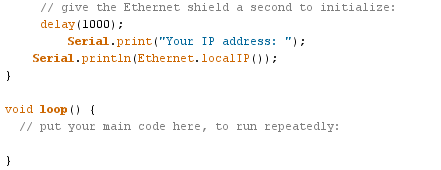
**6.3. Enviando valores desde Galileo a Ubidots**

Conecte la tarjeta a un puerto LAN con salida a internet. Tenga en cuenta que las restricciones dentro de la universidad posiblemente no permitirán que Galileo se conecte a internet, por lo que se recomienda utilizar un router intermedio en función de NAT. Conecte el cable de Ethernet antes de conectar el cable de poder de la tarjeta.

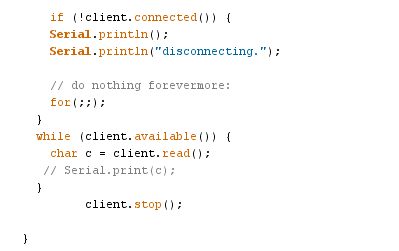
Defina como variables el token y los id obtenidos en el paso anterior, también declare una mac para asignar a la tarjeta e incluya las librerías necesarias:











Una vez enviado un valor exitosamente, este se podrá ver en tiempo real en el dashboard de Ubidots.



**7. ENTREGABLES**

Actividades:

* Integre el proceso de censado de temperatura y movimiento visto en la actividad 1, con el proceso de envío de la actividad 2.
* Cree un cliente en otra plataforma, que consuma los servicios REST de Ubidots de consulta para obtener los últimos valores de las variables enviadas y simularlos gráficamente. Se recomienda utilizar el lenguaje processing, dado su similitud a la manera de usar Arduino y su sencillez. Refiérase a la documentación del API de Ubidots para tal proceso.
* Informe de laboratorio donde documente el proceso para desarrollar los dos puntos anteriores.

**8. REFERENCIAS**

Basada en la guía de laboratorio “Laboratorio 2 - Trabajando con Sensores” de Jesse Padilla.

[1] Arduino Cookbook. O'Reilly. 2nd Edition.

[2] Arduino in Action. Manning. 1st Edition.

[3] Arduino, a Quick Star Guide. The Pragmatic Programmers. 1st Edition.

[4] Getting Started with Arduino. O'Reilly. 2nd Edition.

[5] Make: Getting Started with Sensors. Maker Media. 1st Edition.

**9. REFERENCIAS WEB**

[10] Intel Galileo - <http://arduino.cc/en/ArduinoCertified/IntelGalileo>

[11] Intel Galileo Software - <https://communities.intel.com/docs/DOC-22226>

[12] Galileo Getting Started Guide - <https://learn.sparkfun.com/tutorials/galileo-getting-started-guide>

[13] Passive Infrared Sensor - <http://en.wikipedia.org/wiki/Passive_infrared_sensor>

[14] Getting Started, Ubidots - <http://ubidots.com/docs/get_started/index.html>

**HISTORIAL DE REVISIONES**

|  |  |  |
| --- | --- | --- |
| **FECHA** | **AUTOR** | **OBSERVACIONES** |
| 23/05/2015 | **Santiago Rojas Ricaurte**  s.rojas963@uniandes.edu.co | Versión inicial del documento. |