Congr. Int. en Ing. Electrónica. Mem. ELECTRO, Vol. 39, pp. 199-204, Oct 2017, Chihuahua, Chih. México
http://electro.itchihuahua.edu.mx/memorias_electro/MemoriaElectro2017.zip
ISSN 1405-2172

DESARROLLO DE SISTEMA SCADA BASADO EN EL DISPOSITIVO PHOTON Y EL SOFTWARE MYOPENLAB.

Orlando Hernández Pérez[1, 2], Ernesto Jesús Rincón Martínez[1] , Juan Antonio Rojas Estrada[1], Jonam L Sánchez Cuevas[1], Enrique Mena Salgado[2].

[1] Tecnológico Nacional de México, Instituto Tecnológico de Nuevo León, División de Estudios de Posgrado e Investigación, Av. Eloy Cavazos No. 2001, Colonia Tolteca, C.P. 67170, Cd. Guadalupe, N.L., México.
[2] Tecnológico Nacional de México, Instituto Tecnológico de Iguala, Departamento de Sistemas y Computación, Carretera Nacional Iguala-Taxco esq. Periférico Norte Colonia Adolfo López Mateos, Infonavit, C.P. 40030, Iguala, Gro., México.
Tel. +52 (733) 106-01-23
orlando.hernandez@itiguala.edu.mx

RESUMEN

Este trabajo presenta el desarrollo de un sistema para obtener datos mediante sensores de corriente, temperatura y humedad conectados a un dispositivo relacionado con el internet de las cosas, llamado Photon, así como su interacción con la plataforma y entorno de desarrollo para diseño de sistemas de código abierto llamada MyOpenLab. El proyecto está enfocado a monitorear el funcionamiento de aires acondicionados en casas habitación o negocios mediante un panel de control creado con MyOpenLab mostrando gráficamente la información obtenida y a la par con el servicio web ThingSpeak visualizarla vía internet. El objetivo principal del proyecto está enfocado a ser la base para la creación de un sistema SCADA que pueda ser instalado en áreas comerciales o residenciales. Destacando que la parte más importante de este proyecto es la interfaz, mediante software, entre el dispositivo Photon y el software MyOpenLab ya que no existían publicaciones técnicas al respecto.

Palabras clave: Photon, MyOpenLab, Servicios Web, Internet de las Cosas, SCADA.

ABSTRACT

This paper presents the development of a system to obtain data through current, temperature and humidity sensors connected to a device related to the internet of things (IoT), called Photon and its interaction with the platform and development environment for designing open source systems called MyOpenLab, the project is focused on monitoring the performance of air conditioners in homes or business through a dashboard created with MyOpenLab showing the behavior of the data side by side with the ThingSpeak web service that shows the information by internet. The main objective of the project is the creation of a monitoring system SCADA that can be installed in commercial or residential areas. However, the most important part of this project is the interface, through software, between the Photon device and the software MyOpenLab due to there weren't technical publications about the topic.

Keywords: Photon, MyOpenLab, Web Services, IoT, SCADA.

1. INTRODUCCIÓN

Uno de las erogaciones más grande que se generan en la casa o una empresa es el pago de energía eléctrica y es uno de los servicios que su precio aumenta regularmente, aun cuando la empresa que actualmente la suministra, tiene tarifas diversas para cada usuario dependiendo del giro o utilización y cantidad de energía, así como en los horarios de consumo y varios otros factores diversos[1]. Uno de los aparatos o dispositivos que más consume energía es el aire acondicionado, ya sea de solo frio o de frio/calor, aun cuando los avances en investigaciones han logrado que se reduzca el consumo mencionado derivado de las nuevas tecnologías utilizadas en estos aparatos como lo es la tecnología Inverter[2]. Por lo anterior se planteó un proyecto que nos permitiera monitorear y llevar una estadística de forma visual del consumo de energía en aires acondicionados al cual le agregamos la obtención de parámetros como humedad y temperatura en el ambiente. Al investigar nos encontramos que existían un número significativo de proyectos similares utilizando la tarjeta de Arduino, que contiene un microcontrolador ATmega328P[3], y el software LabVIEW de la compañía National Instruments[4], este último domina el mercado en el diseño y control de sistemas de prueba para las empresas, ya que cuenta una plataforma y entorno de desarrollo para diseñar sistemas, con un lenguaje de programación visual gráfico, sin embargo tiene costo y éste es elevado, por lo que se pretende implementar nuestro sistema de monitoreo en casas primero, y después en empresas. Durante la búsqueda de información encontramos un dispositivo llamado Photon de la compañía Particle, el cual tiene entre otras características un microcontrolador ARM Cortex M3 y Wi-Fi que es compatible con Arduino y su lenguaje de programación basado en C y además un software de código abierto para pruebas y diseño de sistemas llamado MyOpenLab, el cual ya contiene la interfaz para conectarse con Arduino. Sin embargo lo que consideramos más importante en el trabajo que presentamos es que se logró la interfaz entre MyOpenLab y el dispositivo Photon ya que no existía y tampoco había bibliografía disponible. Por lo que una vez obtenido este logro con MyOpenLab y Photon se decidió realizar el sistema de Monitoreo de aires acondicionados utilizándolos para darle difusión al hardware

y software de código abierto. De esta forma, obtener un sistema de bajo costo que pueda ser utilizado por personas o compañías sin que esto genere una erogación alta para los usuarios del sistema. Nuestro sistema lleva, por medio de gráficas generadas con MyOpenLab, las estadísticas de consumo de energía por aparato de aire acondicionado usando el sensor no invasivo ECS1030-L72 y muestra las variaciones de temperatura y humedad usando los sensores LM35 y DHT11 respectivamente. A la par estos mismos datos son enviados a un servicio web llamado ThingSpeak que genera gráficamente estadísticas, previa programación y éstas pueden ser visualizadas en cualquier parte del mundo vía internet, además brinda un determinado número de transacciones gratuitas por año.

En la actualidad existen sistemas de supervisión, control y adquisición de datos conocidos como SCADA por sus siglas en inglés[5], estos sistemas se realizan de manera específica para la empresa que los requiere y pueden ser tan robustos como se desee, sin embargo el precio de estos sistemas es relativamente elevado y conlleva por lo regular una renta anual, ya que dichos sistemas son realizados por grandes compañías que hacen de su producto un software propietario o privativo. En este proyecto se busca realizar la supervisión y adquisición datos de la forma más económica posible que se pueda utilizando para ello software libre o de código abierto como se ha buscado hacer en otros proyectos [6] o [7].

Por otro lado, el internet de las cosas (IoT, por sus siglas en inglés) juega un papel muy importante en este proyecto ya que se están usando sensores, microcontroladores, equipos de cómputo y software que tienen interacción entre sí y nos dan información vía los datos que se obtienen y nosotros los seres humanos podemos tomar decisiones en base a éstos[8].

2. COMPONENTES PRINCIPALES.

2.1. Hardware

El sistema de monitoreo que se presenta está formado por sensores modelo ECS1030-L72[9] y es manufacturado por ECHUN Electronic[10]. Se seleccionó este sensor ya que es no invasivo, es decir, no tenemos que cortar ningún cable conductor de energía y existe menos riesgo al trabajar con él. El ECS1030-L72 trabaja con una relación de corriente de 30A/15mA lo que nos permite trabajar con el dispositivo Photon.

Fig. 1 Sensor de Corriente No-Invasivo-30A.

El sensor manda la señal obtenida en mili volts, sin embargo, esta señal fluctúa de negativo a positivo ya que la corriente que estamos leyendo en corriente alterna, para asegurarnos de que el voltaje que llegue a nuestro dispositivo Photon podemos usar un amplificador operacional o un divisor de voltaje con un condensador y esto además nos ayuda para filtrar la señal y obtenerla con menos ruido.

Otro Sensor que se utiliza es el LM35 que es un sensor de temperatura de Texas Instruments muy confiable y su rango de lectura es de -55 °C a 150 °C y la relación de salida de 10mV/°C[11].

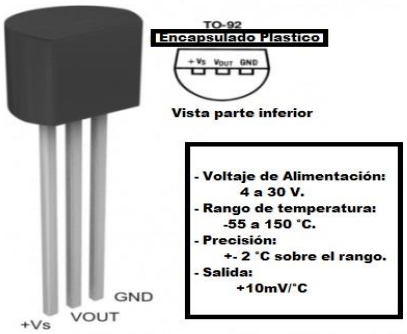

Fig. 2 Sensor de Temperatura LM35.

Además de los dos sensores anteriores utilizamos el DHT11 para medir la humedad del ambiente que maneja una histéresis de ± 0.3%[12][13].

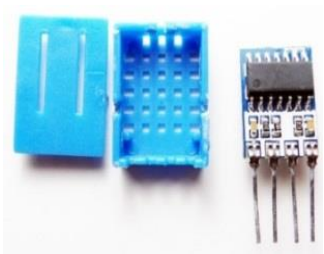

Fig. 3 Sensor de Humedad y Temperatura DHT11.

Todos los sensores anteriores envían la señal en un rango de 0 a 3.3vcd a las entradas analógicas o digitales del dispositivo utilizado en el internet de las Cosas llamado Photon cuyas características principales son[14]:

- Módulo Wi-Fi Particle PØ
  - Broadcom BCM43362 Wi-Fi chip
  - 802.11b/g/n Wi-Fi
  - STM32F205RGY6 120Mhz ARM Cortex M3
  - 1MB flash, 128KB RAM
- RGB status LED
- 18 GPIO de señal mixta.
- Diseño de código abierto.
- Sistema Operativo de tiempo real (FreeRTOS)

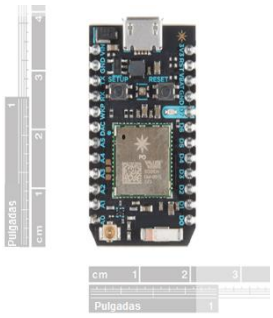

Fig. 4 Dispositivo Photon.

Como se puede observar en la figura 4 las dimensiones del Photon (con Wi-Fi incluido) son más pequeñas que una tarjeta Arduino UNO, sin embargo podría ser del tamaño de una tarjeta Arduino Nano, sin embrago el Arduino Nano no cuenta con conectividad Wi-Fi por lo que se tendría que utilizar un como accesorio un dispositivo ESP8266 que es un módulo que permite la comunicación Wi-Fi.

Cabe hacer la aclaración que en el Photon las entradas analógicas pueden leer valores de 0 a 3.3vcd con una resolución de 4096 (12 bits) a diferencia de Arduino que se caracteriza por leer valores de tensión de 0 a 5vcd con una resolución de 1024 (10 bits). Por lo que se deben hacer los cálculos correctos a la hora de hacer las conversiones de datos.

El voltaje de trabajo que utiliza el Photon puede ser suministrado a través de un conector micro USB o directamente en el puerto VIN, si el voltaje es suministrado vía VIN debe regularse entre 3.6vcd and 5.5vcd y si es suministrado vía USB el puerto VIN tendrá un voltaje de 4.8VCD debido a un diodo de protección de polaridad invertida de la serie schottky entre V+ del USB y VIN la máxima salida de corriente del puerto VIN es de 1Amper. En el puerto 3V3 que maneja un voltaje de 3.3vcd tiene un límite de corriente de salida de 100mA.

Además la compañía Particle cuenta con una plataforma en la nube para sus productos donde quedan guardados los programas que se compilan y se cargan en los microcontroladores, se puede ver los valores de las variables y si las funciones están recibiendo o enviando datos, así como una sección de integración con otras plataformas para realizar el envío de datos vía internet.

2.2. Software

MyOpenLab es un entorno orientado a la simulación y modelado de sistemas físicos, electrónicos y de control [15] que está siendo utilizado en Europa, es de código abierto y por esa razón en el aspecto económico es mejor que LabVIEW para este proyecto, claro sin el soporte, mercadotecnia e infraestructura que éste último y ahí es realmente donde radica el trabajo de investigación e implementación realizado en este proyecto. MyOpenLab al momento de empezar a utilizarlo, solo se tenían desarrolladas interfaces para interactuar con él de las tarjetas K8055, PSC10, Arduino, ModBus RTU, MSE-PCIO-4E4S, Raspberry Pi, RS232 y firmata. Esta última es la base de la interface realizada con el Photon vía un puerto serial para poder recibir y mandar datos desde MyOpenLab a Photon y Viceversa.

3. DESARROLLO

Se realizaron las conexiones de los sensores a la tarjeta Photon mandando la señal obtenida de dos sensores de corriente ECS1030-L72 a las entradas analógicas A2 y A3 (cabe mencionar que se hicieron pruebas usando a la par otro modelo de sensor no invasivo, el SCT-013-30 como el azul que se aprecia en la figura 5, pero se tomó la decisión de utilizar el ECS1030-L72 por su tamaño).

Fig. 5 Conexión de sensores no invasivos en centro de carga.

Sin embargo la señal obtenida de cada sensor ECS1030-L72 es tanto positiva como negativa, ya que lee corriente alterna, estos cambio de fase dañarían el Photon.

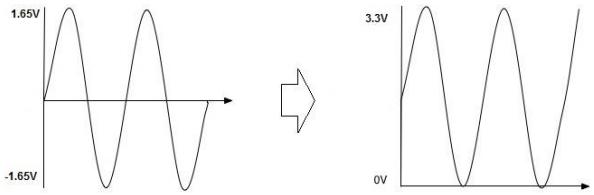

Fig. 6 Onda inicial que se obtiene en los sensores y onda final que buscamos para no afectar el Photon.

Por lo que se utilizó un divisor de voltaje con 2 resistencias de 10KΩ para obtener esos 3.3V y añadimos una resistencia de carga con un valor de 18Ω entre la entrada y salida del sensor para convertir la intensidad a voltaje y poderlo recibir en la entrada analógica del Photon. También se agregó un condensador de 10uF para suavizar la señal disminuyendo el posible ruido (variaciones de lectura) ya que como se utilizó la librería EmonLib se necesita calibrar el valor que recibe esta librería en el método current para cada puerto analógico y se hace un promedio de 2000 lecturas para obtener el Irms

(valor cuadrático medio de la Corriente) y en este caso fue de 111.1 ya que se dividió 2000/18ohms, una vez este valor se pasa como parámetro de entrada, como se muestra en la figura 9 en las líneas 827 y 828 a continuación.

```
827  emon1.current(A2, 111.1);          // Current: input pin, calibration.(2000/18ohms)
828  emon2.current(A3, 111.1);          // Current: input pin, calibration.(2000/18ohms)
829
830  dht.begin();
831
832  Particle.variable("analog",&reading,INT);
833  Particle.variable("volts",&volts,DOUBLE);
834  Particle.variable("voltsc1",&volts,DOUBLE);
835  Particle.variable("voltsc2",&volts,DOUBLE);
836  Particle.variable("tempC",&tempC, DOUBLE);
837  Particle.variable("tempF",&tempF, DOUBLE);
838  Particle.variable("humedad",&hum, DOUBLE);
839  Particle.variable("corriente",&corriente, DOUBLE);
840  Particle.variable("corriente2",&corriente2, DOUBLE);
```

Fig. 9 Segmento de código 1.

Además la señal del sensor LM35 se envió a la entrada analógica A0 y la señal del sensor DHT11 fue enviada a la entrada digital D0, cabe hacer mención que este sensor necesita una resistencia de carga de 10KΩ como se muestra en la figura 8.

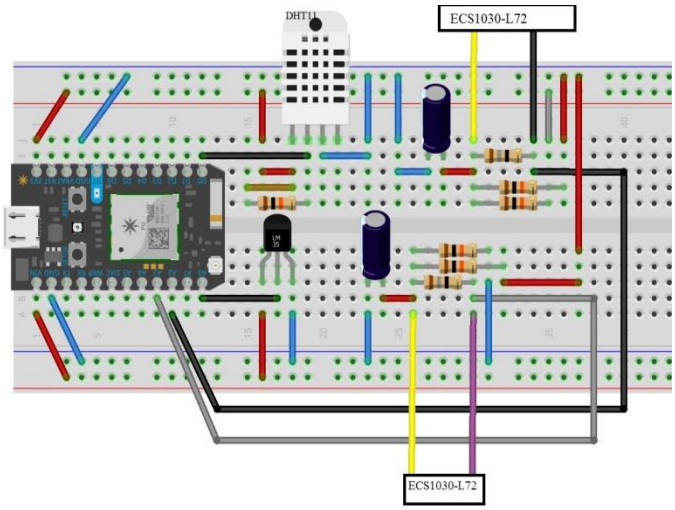

Fig. 8 Circuito montado en Protoboard realizado en Fritzing.

En la plataforma de Particle, una vez hecha la programación correspondiente del sketch (nombre que se le da al código), se compila y se carga en el microcontrolador vía inalámbrica (a diferencia de otras tarjetas con microcontroladores que tienen que estar conectadas mediante un cable), cabe hacer mención que el sketch queda cargado en la nube en la plataforma de Particle en la cuenta del usuario, por lo que si se necesita utilizar ese sketch u otro nuevo y compilarlo e insertarlo a en el dispositivo Photon solo se necesita una conexión de internet. Dicho sketch cuando se carga al Photon se convierte en un Firmware para el Microcontrolador, el cual se ejecutará en tiempo real. El Photon además cuenta con 2 claves de seguridad, un Access Token, clave numérica que está asociada a la cuenta del usuario, y un device ID, clave alfanumérica asociada al dispositivo Photon en particular, ambas claves le dan mayor protección a tus programas o sketches.

En el Sketch que se utilizó se insertó parte del código del protocolo Firmata que conecta computadoras de manera más fácil con microcontroladores[16] y además es de código abierto. Se utilizaron las librerías Adafruit_DHT y EmonLib las cuales están enfocadas a trabajar con Arduino. Así se logró conectar el Photon y MyOpenLab para que puedan recibir y enviar información entre sí.

Por otro lado dentro de las opciones de integración en la plataforma de Particle existen varias opciones para interactuar con los datos obtenidos, en este proyecto se utilizó Webhook. En la figura 9 se muestran las diversas opciones de integración de la plataforma de Particle para sus dispositivos.

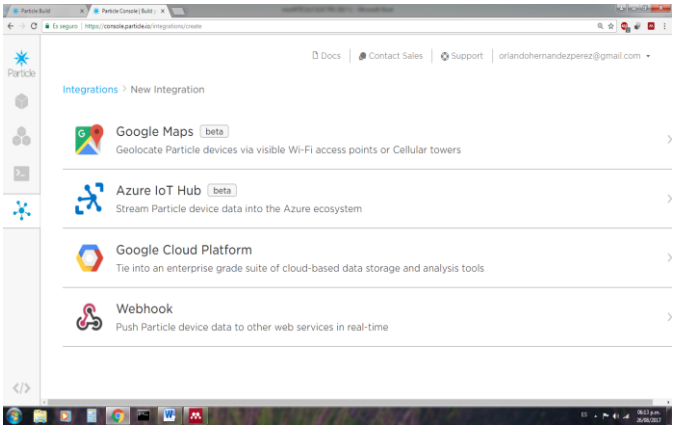

Fig. 9 Submenú de integraciones de Particle.

Para que los datos sean recibidos se debe crear una cuenta en la plataforma ThingSpeak y codificar un canal para mostrar los resultados de los datos enviados por el dispositivo Photon, los cuales fueron obtenidos por los sensores, es en ese momento que se genera la clave que puede utilizar en el sketch que se desee utilizar. En la figura 10 muestra la vista parcial de la página web ThingSpeak donde se visualiza la generación de una clave para escritura y otra para lectura.

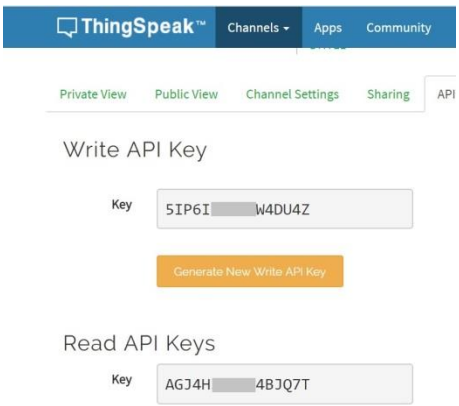

Fig. 10 Sección de la página web de ThingSpeak.

En la figura 11 se muestra parte del código necesario para enviar o publicar los datos a la página que brinda los servicios web, en la línea 941 se muestra "xxxxxxxxx" ya que en esa parte va la clave que nos proporciona ThingSpeak para darle seguridad a nuestros datos, ya que se envían encriptados.

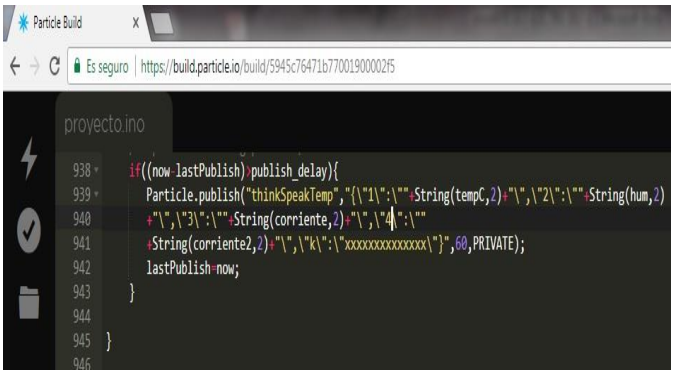

Fig. 11 Segmento de código 2.

Cabe recordar que los sensores envían los datos en milivolts y se tienen que hacer las conversiones necesarias para que se puedan interpretar los valores de forma coherente. Cada canal permite hasta 8 sensores de forma simultánea. Las lecturas que son gratuitas pueden variar de 15 a 60 segundos, sin embargo, si se desea que la lectura de variables que son leídas por el servicio web sea cada segundo existen planes en lo que hay que pagar cierta cantidad de dinero, además de que los canales donde se publica la información en internet pueden ser públicos o privados.

4. RESULTADOS EXPERIMENTALES

Una vez conectado e implementado el sistema los sensores de corriente no invasivos instalados en las conexiones de 2 equipos de aire acondicionado, uno de 12000 BTU y el otro de 18000 BTU se obtuvieron resultados que se pudieron visualizar vía ThingSpeak.

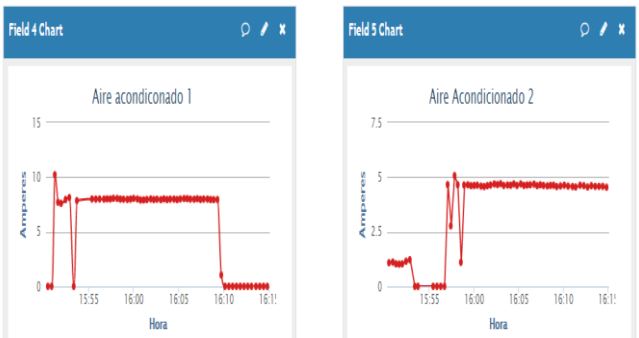

Fig. 12 Gráficas de consumo de energía en amperes de equipos de aire acondicionado.

La figura 12 nos muestra las lecturas de la intensidad de corriente nominal o amperaje de los aires acondicionados de prueba, en el cual se muestra un característico consumo un poco mayor en el arranque del aparato, situación que es normal en los aires acondicionados, las lecturas que obtuvieron del aire acondicionado de 18,000 BTU fueron aproximadas a los 8 amperes y las del aire de 12000 BTU fueron cercanas a 5 amperes, comparando estas medidas con las de un amperímetro de gancho, la variación fue de un 2%.

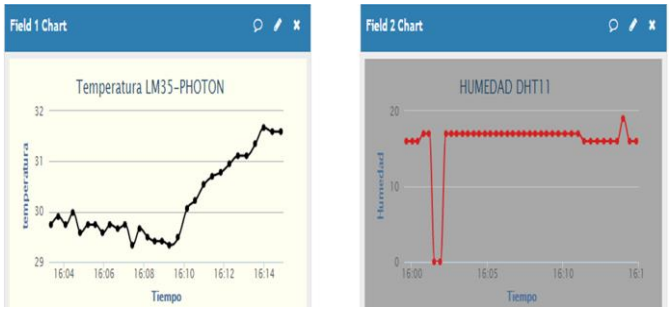

Fig. 13 Gráficas que muestran los datos de los sensores LM35 y DHT11.

En el caso del sensor de temperatura y el de humedad, cuyas gráficas se muestran en la figura 13, los resultados de los sensores variaron en el caso de sensor de temperatura 1% y el de humedad 3%, rangos aceptables dentro de las especificaciones del fabricante.

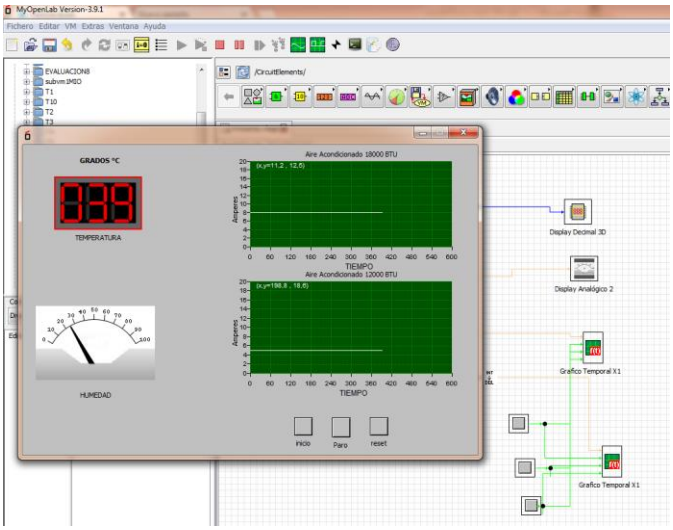

Fig. 14 Tablero creado con MyOpenLab recibiendo datos del Photon.

Y finalmente la figura 14 muestra los datos recibidos de los sensores vía el dispositivo Photon en forma gráfica y visualmente agradable en un tablero de control y visualización realizado con el software MyOpenLab, lo cual nos muestra la

interacción que se buscaba realizar entre el hardware y el software con el que se realizó el proyecto.
Debemos hacer énfasis en extremar precauciones cuando se trabaje con dispositivos o instalaciones que manejen corriente alterna como en el caso de nuestro proyecto, en el cual se tomaron las medidas necesarias.

5. CONCLUSIONES

Este trabajo nos permitió realizar la comunicación de un dispositivo enfocado al internet de las cosas, el dispositivo Photon, con una plataforma que tiene todos los atributos para ser utilizado en empresas de manera gratuita para crear sistemas mediante su simulación y modelado como lo es MyOpenLab. El costo de este sistema de monitoreo para este proyecto solo incluye lo erogado por el hardware utilizado ya que no hay que pagar licencias o rentas por el software a utilizar, MyOpenLab, ya que es un software de código abierto y eso es una gran ventaja contra el mismo sistema elaborado con LabVIEW. Este proyecto presenta una nueva alternativa a proyectos realizados con Arduino y LabVIEW. Sin embargo hasta el momento en los avances de este trabajo la conexión entre el hardware y software se limita a un puerto serial, que realmente es el éxito del proyecto ya que no se tiene conocimiento de que dicha conexión se haya realizado previamente. No obstante el sistema SCADA está funcionando obteniendo datos de los sensores que tiene conectados y éstos pueden ser visualizados desde cualquier parte del mundo, conociendo el canal adecuado. Como trabajo futuro se busca hacer el diseño formal de una placa de circuito impreso con al menos 6 conectores de 3.5 mm para recibir igual número de sensores de corriente no invasivos y cambiar los filtros de corriente alterna a continua por un amplificador operacional.

6. REFERENCIAS

[1] Secretaría de Energía, *Manual de Disposiciones Relativas al Suministro y Venta de Energía*. 2013, p. 14.
[2] J. Mix, "Inverter technology," *ASHRAE J.*, vol. 49, no. 10, p. 77, 2014.
[3] D. Charalampos, "Chapter 1 |The Internet of Things," in *Building Internet of Things with the Arduino*, 1st ed., athens: Createspace Independent Publishing Platform, 2012, p. 340.
[4] B. Ehsani, *Data Acquisition Using LabVIEW*, 1st ed. Packt Publishing, 2016.
[5] P. Aquilino Rodríguez, *Sistemas SCADA*, 3a. MARCOBO S.A., 2012.
[6] Y.-C. Wu, M.-J. Chen, B.-S. Chang, and M.-T. Tsai, "A low-cost web-based infrared remote control system for energy management of aggregated air conditioners," *Energy Build.*, vol. 72, pp. 24–30, 2014.
[7] Q. Liu, Y. Ma, M. Alhussein, Y. Zhang, and L. Peng, "Green data center with IoT sensing and cloud-assisted smart temperature control system," *Comput. Networks*, vol. 101, pp. 104–112, Jun. 2016.
[8] M. M. Rathore, A. Ahmad, A. Paul, and S. Rho, "Urban planning and building smart cities based on the Internet of Things using Big Data analytics," *Comput. Networks*, vol. 101, no. 2016, pp. 63–80, 2016.
[9] Sparkfun Electronics, "Non-Invasive Current Sensor - 30A - SEN-11005 - SparkFun Electronics." [Online]. Available: https://www.sparkfun.com/products/11005. [Accessed: 01-Jun-2017].
[10] L. ECHUN Electronic Co., "Datasheet ECS1030-L72," 2016.
[11] Texas Instruments Incorporated, "LM35 Precision Centigrade Temperature Sensors," 2016.
[12] Aosong Electronics, "Temperature and humidity module DHT11 Product Manual," 2016.
[13] "Inside and Outside Thermometer on AVR micro," 2012. [Online]. Available: http://www.radiolocman.com/shem/schematics.html?di=112850. [Accessed: 10-Jun-2017].
[14] "Particle Datasheets Documentation | Photon datasheet." [Online]. Available: https://docs.particle.io/datasheets/photon-datasheet/. [Accessed: 10-Jun-2017].
[15] "myopenlab." [Online]. Available: http://myopenlab.wixsite.com/myopenlab. [Accessed: 10-Jun-2017].
[16] J. A. Langbridge, *Arduino Sketches: Tools and Techniques for Programming Wizardry*, 1st ed. Indianapolis, Indiana: John Wiley & Sons, 2015.