

Internet de los objetos empleando arduino para la gestión eléctrica domiciliaria

FECHA DE RECEPCIÓN: 9 de junio
FECHA DE APROBACIÓN: 12 de julio
Pp. 22-41

Adriana Marcela Vega E.*
Francisco Santamaría P.**
Edwin Rivas T.***

*Objects internet using "Arduino"
hardware for home electrical
management*

*Analyse des composants
internes des objets utilisant
de l'arduino pour la gestion
électrique des foyers*

*Internet dos objetos usando
Arduino para a gestão elétrica
domiciliar*



**Doctorado en Ingeniería (c),
Énfasis en Ciencias de la Información y el Conocimiento.
Universidad Distrital Francisco José de Caldas;
Maestría en Ingeniería Industrial énfasis en Organización, Sistemas de Información y Gestión empresarial.
Universidad Distrital Francisco José de Caldas;
Especialización en Planificación y Desarrollo Regional y Urbano
Escuela Superior de Administración Pública; Ingeniero Industrial de la Universidad América.*

***Doctor en Ingeniería; Magíster en Ingeniería Eléctrica; Ingeniero Electricista, Ingeniero Electricista
Profesor Asistente de la Facultad de Ingeniería de la Universidad Distrital Francisco José de Caldas*

****Doctorado Universidad Carlos III de Madrid en Ingeniería Eléctrica, Electrónica y Automática; Magister Universidad
del Valle en Sistemas de Generación de Energía Eléctrica;
Magister Universidad Carlos III de Madrid en Ingeniería Eléctrica, Electrónica y Automática;
Ingeniero eléctrico de la Universidad del Valle.*

RESUMEN

El presente artículo muestra la importancia de la gestión eléctrica domiciliar por medio del diseño, desarrollo e implementación de un prototipo de sistema domótico, que permite utilizar de manera remota una bombilla LED a través de las acciones de prender y apagar, según las necesidades del usuario, sin importar en donde se encuentre ubicado, es decir, que su localización geográfica no importa. Para su desarrollo se emplearon elementos de Hardware como el Arduino y de Software como los servicios en la nube de Windows Azure®; con esta aplicación, se pretende observar la interoperabilidad entre un dispositivo ubicado en el hogar y un servidor ubicado en el mundo virtual del Internet. Lo anterior permite hacer un acercamiento a las múltiples modalidades que se pueden desarrollar con el Internet de los objetos y que en un futuro próximo será empleando por las personas de manera permanente en sus residencias.

ABSTRACT

This article shows the relevance of home electrical management using design, development and implementation of a prototype of domotic system that allows us to use an LED bulb in a remote way to switch on and off according to the needs of the customer; i.e. no matter where the customer is located. For its development, hardware components were used such as "Arduino" and also software components such as the services of Windows Azure® cloud. With this application, the interoperability existing between this device located at home and a server located in the virtual world of Internet. Consequently, with all these practices, different modalities to be developed in objects internet are approached, and in a near future they will permanently be used by people in their homes.

RESUMÉ

Cet article montre l'importance de la gestion électrique des foyers par le biais de la conception, du développement et de la mise en place de prototypes de système de domotique permettant d'utiliser à distance une ampoule LED selon les besoins de l'utilisateur et cela où que celui-ci se trouve. Pour le développement du prototype nous avons utilisé des éléments tels que l'arduino et des logiciels comme le service cloud de Windows Azure®. Cette application est destinée à observer l'interopérabilité entre un périphérique situé dans une maison et un serveur virtuel se trouvant sur Internet. Le but de l'expérience étant de nous rapprocher des multiples modalités qu'il est possible de développer grâce aux réseaux internes des objets. Dans un avenir proche, la domotique sera utilisée en permanence dans les foyers.

RESUMO

Este artigo mostra a importância da gestão elétrica domiciliar através do desenho, desenvolvimento e implementação de um protótipo de sistema domótico que permite utilizar de maneira remota uma lâmpada de LED através das ações de ligar e desligar, segundo as necessidades do usuário onde estiver localizado, isto significa que sua localização geográfica não importa. Para seu desenvolvimento foram usados elementos de hardware como o Arduino e software como os serviços na nuvem de Windows Azure®, com esta aplicação pretende-se observar a interoperabilidade entre um dispositivo localizado no lar e um servidor localizado no mundo virtual da internet, isto permite abranger as múltiplas modalidades que podem ser desenvolvidas com a internet dos objetos e que num futuro próximo estarão sendo utilizadas pelas pessoas de maneira permanente em suas residências.



Palabras claves

Arduino
Domótica
Gestión
Internet de los objetos
Servidores



Key words

Arduin
Hardware
Domotic
Management
Objects internet and servers



Mots clefs

Arduino
Domotique
Gestion
Composants internes
Serveurs



Palavras-chave

Arduino
Domótica
Gestão
Internet dos objetos
Servidores

1. INTRODUCCIÓN

Este artículo tiene como objetivo diseñar, desarrollar e implementar un prototipo que permita gestionar una bombilla LED de manera remota para prenderla y apagarla empleando una placa o entorno denominado Arduino, el cual hace parte de un sistema de gestión de energía eléctrica domiciliaria que monitorea, controla y supervisa variables eléctricas, que se producen por el empleo de aparatos eléctricos en una casa.

El prototipo se basa en conceptos relacionados con el Internet de los objetos, ya que hay una capa de conectividad digital para unir la bombilla con el Internet y permitir su interacción por parte del usuario del sistema, es decir, una conexión del universo físico con el universo virtual. Para la configuración del *Software* se emplearon dos ambientes de desarrollo: el propio del Arduino y el *Software Visual Studio.Net*® utilizando los servicios *Web de Windows Azure*®, ubicados en la nube (universo virtual). Con relación a la parte física, aunque en el mercado existen muchas placas microcontroladoras con funcionalidades similares, se escogió el Arduino porque simplifica el proceso de trabajar con electrónica digital, bajos costos de adquisición, funciona con diferentes sistemas operativos (multiplataforma) y su *Software* es

de código abierto. Para poner en funcionamiento y ver la efectividad del prototipo, se llevó a cabo el siguiente esquema de trabajo:

- Diseño de la interfaz de potencia como prototipo a escala que permita gestionar un bombillo, la cual hace las veces de *Hardware* o componente físico.
- Desarrollo del programa o *Software* del Arduino, el cual sirvió para la gestión del bombillo.
- Configuración del servidor *Web Windows Azure*®, en donde se alojaron los componentes de *Software* que permitieron la gestión del bombillo de manera remota (componente virtual).
- Desarrollo del programa o *Software* con *Visual Studio.Net*® para la integración con el *Hardware* Arduino y la interfaz de potencia de la bombilla (Internet de los objetos).
- Realización de pruebas de validación y efectividad de manera remota del sistema prendiendo y apagando el bombillo por medio de la página *Web*, donde se constata el funcionamiento del Internet de los objetos.

2. CONCEPTUALIZACIÓN PARA DESARROLLAR EL SISTEMA

2.1 Domótica

El concepto domótica se refiere a la automatización y control (encendido / apagado, apertura / cierre y regulación), de aparatos y sistemas de instalaciones eléctricas y electrotécnicas (iluminación, climatización, persianas, puertas y ventanas motorizados, el riego, etc.), de forma centralizada y/o remota. Está determinado por una medición avanzada de la infraestructura de red eléctrica con un medidor inteligente, el cual cuenta con aparatos inteligentes interconectados (Snyder, Gunther, & Griffin, 2012). Hay una tendencia emergente que ayuda a los consumidores a reducir el consumo de energía de la casa mediante la supervisión y el control de los

electrodomésticos y la reprogramación de su tiempo de funcionamiento, de acuerdo con la demanda de energía y de suministro (De Silva, Morikama, & Petra, 2012). En domótica, los electrodomésticos pueden ser integrados a una red inalámbrica de control IEEE 802.15.4, Z-Wave o *Bluetooth*; bajo estas tecnologías, se utiliza la gestión global de la red de señalización puesto que es escalable, de amplia cobertura y robustez (Nowak, Schaefer, Brzozowski, Kraemer, & Kays, 2011).

Unido a la domótica hay tipos de hogares inteligentes. Se explica brevemente cuáles son ya que están asociados al Internet de los objetos y sin lugar a duda, con la gestión de elementos como el que se configura para este prototipo (Figura 1).

Figura 1. Hogares inteligentes para reducir consumo de electricidad.



Fuente. Harney, 2008.

Los sistemas de domótica actúan sobre los aparatos y sistemas eléctricos de la vivienda, teniendo en cuenta que cada uno tiene sus ventajas e inconvenientes, sin embargo, hay una gran oferta en el mercado y para cada situación hay uno o varios sistemas que se adaptan (Jarman, Hooton, Walker, & Wang, 2010). Las investigaciones están abordando el desarrollo a bajo costo y fácil despliegue de los sistemas de apoyo a la actividad de detección del contexto de aplicaciones en el hogar; por ejemplo, el uso de matrices de sensores de bajo costo como detectores de movimiento o interruptores de contacto (Berenguer, Giordani, Giraud, & Noury, 2008).

La arquitectura de los sistemas de domótica, hace referencia a la estructura de su red y la clasificación se realiza en donde reside la inteligencia del sistema domótico.

Las principales arquitecturas son:
arquitectura centralizada, arquitectura descentralizada, arquitectura distribuida y arquitectura híbrida o mixta (Moore, 2010).



2.2 Protocolos de domótica

Los protocolos de comunicación son los procedimientos utilizados por los sistemas de domótica para la comunicación entre todos los dispositivos, con la capacidad de controlar. Dichos protocolos pueden ser de tipo estándar abierto (uso libre para todos), estándar bajo licencia (abierto para todos bajo licencia), o propietario (uso exclusivo del fabricante o los fabricantes propietarios) (The Cambridge, 2012).

Existen diferentes protocolos en los cuatro procesos que se involucran en el sector energético, como son: generación, transmisión, distribución y el consumidor final. Se observa la relación de estos con las redes de comunicación y su interacción dentro del Modelo *Open System Interconnection* (OSI) de comunicaciones (Tolosa, 2014) (Figura 2).

2.3 Arduino

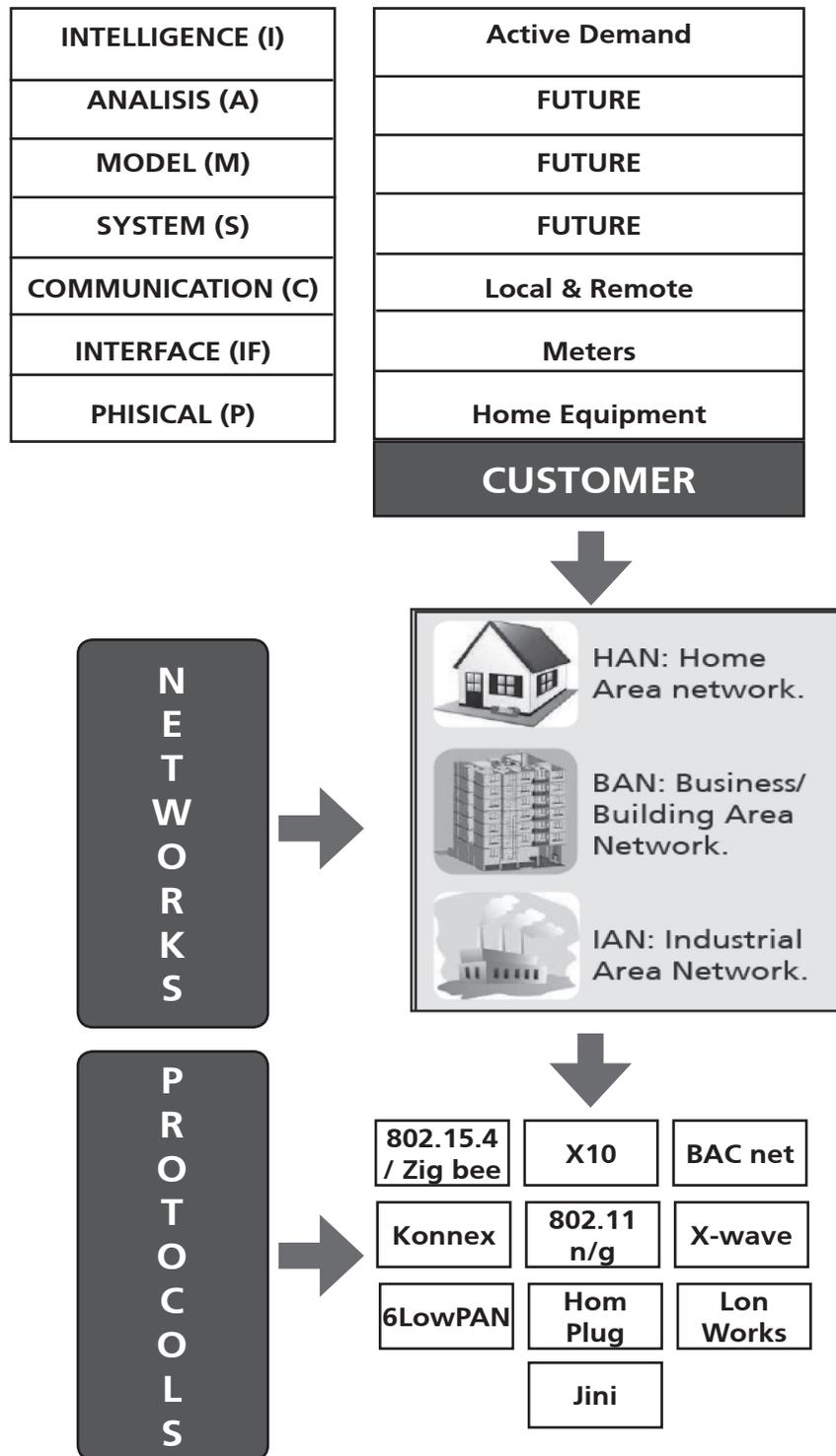
El Arduino, es un elemento de fácil conectividad a una red y adicionalmente permite implementar un servidor de protocolos de alto nivel, como el *Hypertext Transfer Protocol* (HTTP); tiene memoria, capacidad de procesamiento autónomo, compiladores de lenguajes de programación como C y puertos físicos para interconectar con dispositivos.



El Arduino Mega 2560, es una placa electrónica basada en el ATmega2560 (hoja de datos). Tiene 54 pines digitales de entrada/salida, 16 entradas analógicas, cuatro UARTs (puertos seriales), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, una cabecera ICSP y un botón de reinicio. La Mega es compatible con la mayoría de los protectores diseñados para el Arduino Diecimila. (Arduino, 2013), (Tabla 1).

La Arduino *Ethernet Shield* está basada en el chip *Ethernet Wiznet W5100*, el cual se provee de una pila de red IP, soporta hasta cuatro conexiones de *Sockets* simultáneas, usa la librería Ethernet para escribir programas, dispone de conectores que permiten conectar a su vez otras placas sobre la placa Arduino y tiene un conector *Ethernet* estándar RJ45 (Schmidt, 2011).

Figura 2. Modelo OSI, unido a los protocolos de domótica.



Fuente. Tolosa, 2014.

Tabla 1. Resumen Arduino 2560

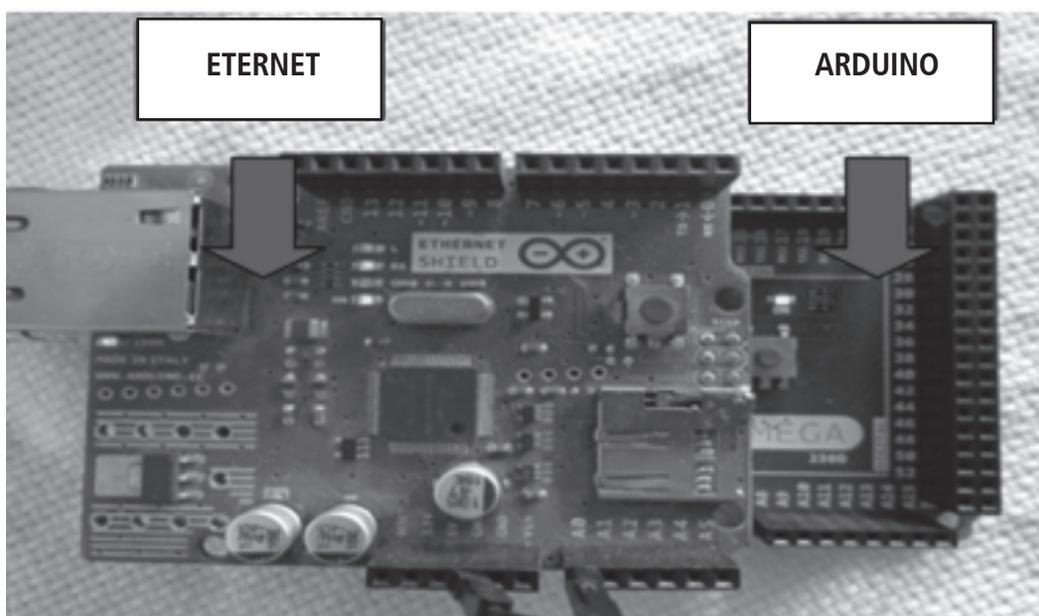
Microcontroller	ATMega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB which 8 KB used bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

La interfaz del Arduino y la tarjeta de conexión Ethernet Shield que se requieren para el desarrollo del prototipo de gestión domótica, se vinculan a través de los puertos de conexión o pines los cuales se ensamblan de manera precisa. Es así como se muestra la unión que se efectuó para el desarrollo del prototipo presentado en este artículo (Figura 3).

2.4 Window Azure®

Windows Azure® es una plataforma de nube abierta y flexible que permite compilar, implementar y administrar aplicaciones rápidamente en una red global de centros de datos administrados por Microsoft. Compila aplicaciones en cualquier lenguaje, herramienta o marco, e integra sus aplicaciones de nube pública con el entorno de tecnologías de la información existente (Windows Azure, 2014).

Figura 3. Conexión de la interfaz del Arduino con la placa *Ethernet Shield*.



La computación en nube se volvió una realidad. Las empresas pagan solo por lo que utilizan, como por ejemplo la capacidad de almacenamiento de sus datos, la electricidad u otros servicios públicos. Lo anterior reduce la inversión en tecnologías de información y el uso más eficiente de los centros de datos. Para beneficiarse de las ventajas de la computación en la nube, las organizaciones migran sus aplicaciones a esta y pagan por lo que emplean (Passos da Costa & Rosado, 2012).

2.5 Visual Studio. Net Express 2012

Se trata de un *Software* de desarrollo en Entorno de Desarrollo Integrado (IDE- por sus siglas en inglés), para sistemas operativos Windows; soporta lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y *Visual Basic*. Es proporcionado a la comunidad de desarrolladores de la programación *Web* y de aplicaciones, ofreciéndose a partir de la versión 2005 de *Microsoft Visual Studio* (Microsoft, 2014).

2.6 Internet de los objetos

El Internet de los objetos, se perfila como una de las principales tendencias que dan forma al desarrollo de las tecnologías de las comunicaciones. Es el paso del Internet utilizado para la interconexión de objetos físicos que se comunican entre sí, con los seres humanos, para ofrecer un determinado servicio. Se basa en tres pilares principales: ser identificables, para comunicarse y para interactuar (Miorandi, Sicari, De Pellegrini, & Chlamtac, 2012).

Esta tecnología de evolución permanente, permite instrumentar los objetos con que interactúan las personas, muchos dispositivos se conectan entre sí por medio de redes de comunicación, pequeños sensores efectúan diversas mediciones ganando capacidad de comunicación en tiempo real, lo cual hace que se disminuyan las barreras que separan el mundo real del virtual (Fundación de la Innovación - Bankinter, 2011).

2.7 Taxonomía de dispositivos inteligentes

En un futuro, todos los aparatos vinculados al sistema eléctrico deben ser inteligentes. Existen dos tipos de dispositivos especializados: los dispositivos eléctricos genéricos (actuadores y potenciadores de calidad de energía) y los dispositivos de medición; es decir, son de medición de puntos en el sistema y dispositivos de medición. Las variaciones entre estos dos, radica en que la medición de puntos suministra solo datos, mientras que los dispositivos de medición, controlan el sistema. Por otra parte, los dispositivos eléctricos generales se categorizan en no gestionables y gestionables; los primeros, participan en el consumo y la producción de energía, pero no son administrados, mientras que los dispositivos gestionables pueden ser activados o detenidos, es decir, controlados. Los anteriores dispositivos se dividen en gestionables activos y gestionables no activos, de acuerdo con el papel que desarrollan en el sistema ya sea consumiendo o produciendo energía (Moore, 2010).

3. DESARROLLO

El prototipo que se presenta en este artículo, es un componente que hace parte de un sistema de gestión para redes eléctricas domiciliarias, el cual involucra aspectos relacionados con el monitoreo, control y supervisión de variables que se producen con la manipulación de dispositivos eléctricos en el hogar por parte de los usuarios.

El diseño, desarrollo e implementación de los componentes físicos y el *Software* que hacen parte del prototipo de gestión domiciliaria aquí planteado, a partir de la utilización de Arduino de manera remota, manejan la siguiente estructura para su funcionamiento y visualización.

3.1 Configuración física. Elementos que se requieren para el desarrollo del prototipo

Para la gestión de una bombilla de manera remota empleando el Arduino, se requiere tener los elementos de *Hardware* y de *Software* que se presentan a continuación (Tabla 2).

Tabla 2. Elementos empleados para gestionar una bombilla con Arduino

Hardware	Software
Arduino Mega	Sistema de desarrollo para el Arduino.
Tarjeta de conexión a redes denominada Arduino <i>Ethernet Shield</i> .	<i>Software</i> de Instalación de herramientas para Windows Azure®.
Interfaz de potencia para manejar corriente.	VisualStudio.NET 2012 <i>Net Express</i> .
Dispositivo a controlar (bombillo con conexión).	Navegador
Enrutador	
Computador	

3.2 Ajuste de puertos por medio de cables de conexión para su acceso remoto

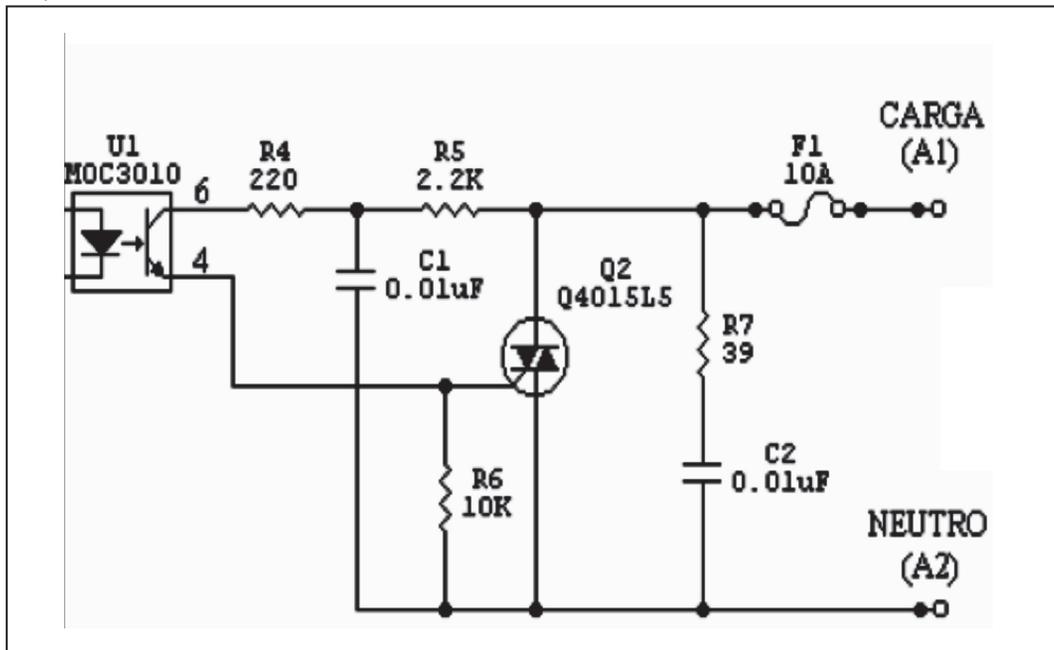
Se hace necesario asignar los números de pines o puertos de conexión entre la placa *Ethernet Shield*, el Arduino y el elemento físico que se va a gestionar, el cual en este caso, es la bombilla y quedan de la siguiente manera: al Pin número 2 se le asigna la salida y se le establece el cable amarillo. El Pin GND (*Ground*) o puesta a tierra, se le establece el cable negro.

3.3 Diseño y desarrollo de la interfaz de potencia para manejar corriente eléctrica

Para el diseño del prototipo de gestión se empleó un diagrama esquemático de conexiones, que es una representación con símbolos de un circuito electrónico. Se observa (Figura 4), el diagrama para realizar la interfaz que servirá de puente entre el Arduino y la bombilla que se va a gestionar de manera remota. Este diagrama permite manejar corriente alterna para generar señales de control de bajo nivel.

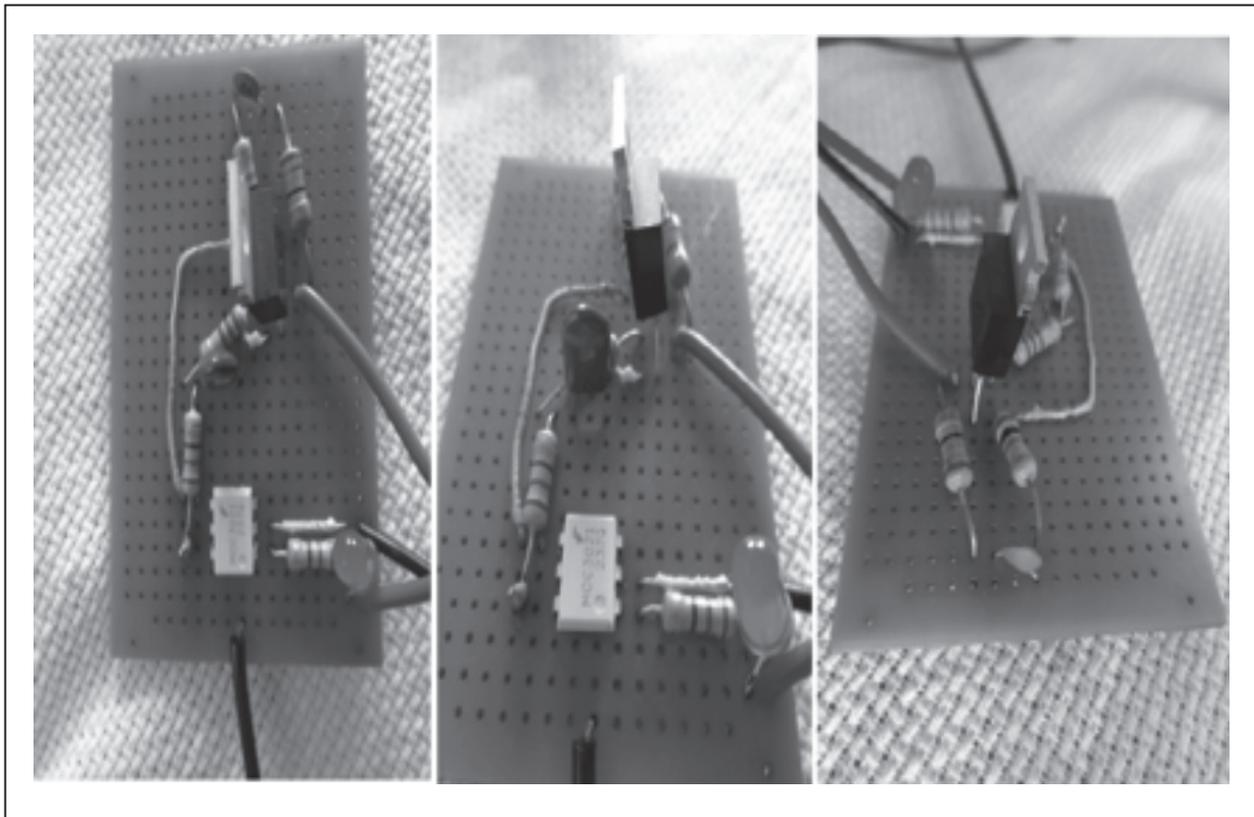
Basados en el diagrama esquemático presentado anteriormente, se procedió a efectuar el prototipo del modelo físico de la interfaz de potencia (Figura 5), empleando los siguientes elementos: R4 - 220 Ω , R6-10 k Ω , R7 - 39 Ω , C1, C2 - 0.01 μ F, U1, optoacoplador - MOC3010, F1, fusible 10 A. De manera simultánea, se le adicionó un bombillo *Light-Emitting Diode* (LED), para que sirviera de validación adicional en el momento de realizar las pruebas de funcionamiento. Se detalló también el prototipo físico o interfaz de potencia debidamente ensamblada con los elementos mencionados anteriormente, desde tres ángulos diferentes.

Figura 4. Diagrama esquemático.



Fuente. Foros de electrónica, 2014.

Figura 5. Interfaz de potencia para gestionar el bombillo



3.4 Diseño y desarrollo del elemento a gestionar

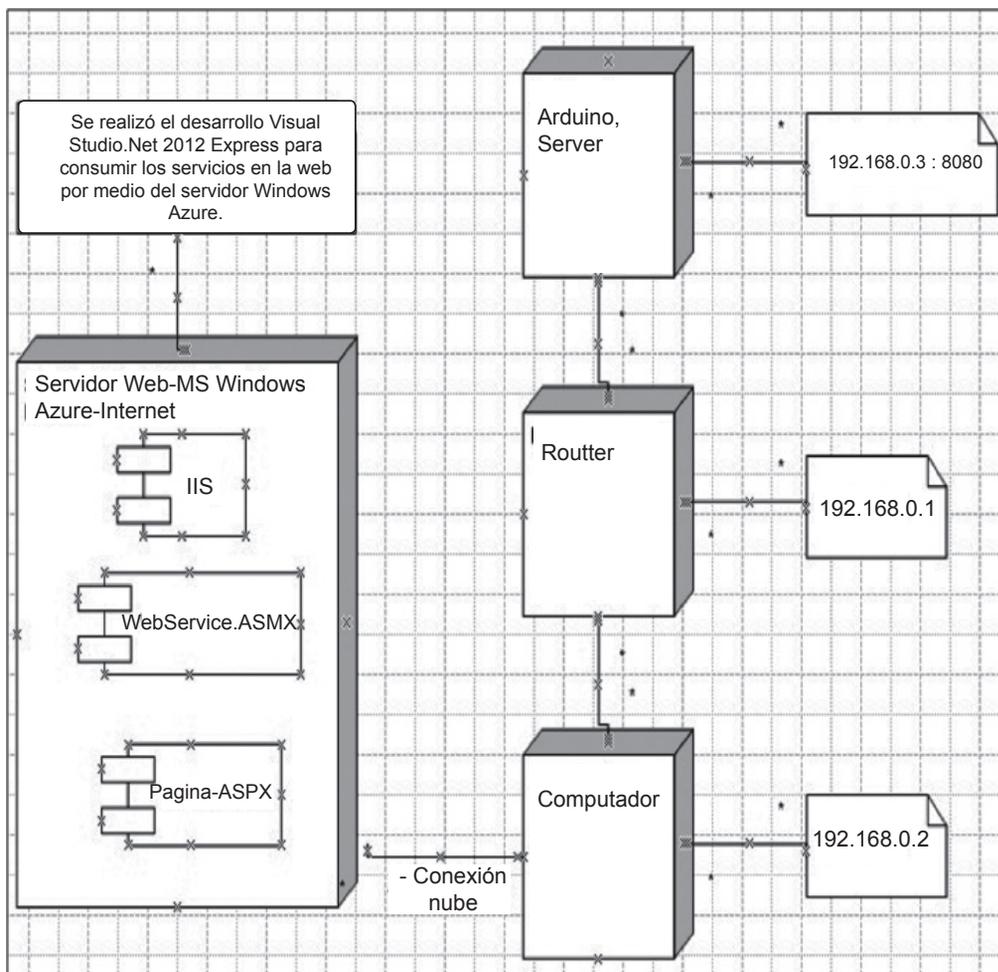
El prototipo que se diseñó soporta hasta 10 A, equivalentes a 1 kW. Para hacer la gestión de manera remota por medio del Arduino, se seleccionó una bombilla de 5W, la cual está unida a la placa Arduino y a la interfaz de potencia que actúa de dos formas: prendida o apagada. Esta es la capa física que interactuará con la capa virtual, de manera que la unión de las dos permita visualizar la tecnología del Internet de los objetos.

3.5 Diseño de diagrama de la red

A continuación, se muestra el diagrama de la red que se adapta al prototipo de gestión (Figura 6). Se emplea un enrutador (*Router*), el cual es un mecanismo que sirve para la interconexión de una red que opera en la capa tres (nivel de red) del modelo OSI y permite que todos los elementos se conecten entre ellos (González, 2014). Las direcciones de los componentes de *Hardware* que hacen parte del prototipo son:

- Dirección del Router: 192.168.0.1.
- Dirección del computador: 192.168.0.2.
- Dirección del Arduino: 192.168.0.3.
- Puerto: 8080.

Figura 6. Diagrama de la red.



3.6 Configuración del Software para trabajar con Windows Azure®

Para gestionar el bombillo de manera remota, es necesario desarrollar componentes de *Software* para que actúen unidos con el prototipo físico que se realizó. En este caso, se eligió utilizar el servidor *Windows Azure®*, ya que es una plataforma amigable y multipropósito. En su página de inicio se observan los diferentes entornos en los que trabaja el servidor *Azure*; de acuerdo con las necesidades del usuario, se procede a activar las requeridas para realizar el desarrollo del prototipo, que en esta oportunidad activó los servicios en la nube creando el respectivo servicio *Web* que permite efectuar la gestión remota de la bombilla.

3.7 Desarrollo de solución *Web* bajo *Visual Studio.NET 2012 Express®*

Para emplear los servicios que ofrece *Windows Azure®* desde la nube, se instala el *Software Visual Studio.Net 2012 Express®* para la *Web*. Por medio de esta plataforma, se crea una solución *Web* denominada por defecto "WindowsAzure1"; dentro de esta, se estableció un proyecto denominado "WebRole1" en el cual se crearon las siguientes páginas: la primera, es la que aloja la página *Active Server Pages* (ASP), que consiste en páginas de ambiente de aplicación abierta y combinadas con los ficheros de *Hyper Text Markup Language* (HTML). Para este caso, la página se denominó "ApagarPrenderSWCloud.aspx". A continuación se muestra el código que se desarrolló.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
```

```
namespace WebRole1
{
public partial class PrendeApagaWSCloud : System.
Web.UI.Page
{tigreWSCloud.WebServiceControlArduinoCloud
tigre;
```

```
protected void Button1_Click(object sender,
EventArgs e)
{
tigre = new tigreWSCloud.
WebServiceControlArduinoCloud();
string dir = tigre.Prende();
Response.Redirect(dir);
Button2.Enabled = true;
Button1.Enabled = false;
}
```

```
protected void Button2_Click(object sender,
EventArgs e)
{
tigre = new tigreWSCloud.
WebServiceControlArduinoCloud();
string dir = tigre.Apaga();
Response.Redirect(dir);
Button1.Enabled = true;
Button2.Enabled = false;
}
}
```

Por otra parte, se creó la página para establecer el servicio *Web*: *WebServiceControlArduinoCloud.aspx*, la extensión *.asmx* es la empleada para ejecutar los servicios *Web*. El código que se desarrolló fue el siguiente:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Services;
namespace WebRole1
{
/// Descripción breve de
WebServiceControlArduinoCloud
public String Prende()
{
```



p.205

```
return "http://192.168.0.3:8080/?dig2=1";  
// esta es la direccion del arduino donde esta  
la pagina que enciende y apaga  
}  
public String Apaga()  
{  
return "http://192.168.0.3:8080/?dig2=0";  
// esta es la direccion del arduino donde esta  
la pagina que enciende y apaga  
}  
}  
}
```

3.8 Desarrollo del *Software* en el sistema del Arduino

El *Software* de desarrollo con el Arduino se realiza configurándolo desde la página de Internet. En esta interfaz, se procede a desarrollar las líneas de código que permiten prender y apagar el bombillo de manera remota. Para este prototipo de gestión la página se denomina "FinalFinal".

Cuando se emplea la placa Arduino en el desarrollo del código de implementación, se debe tener en cuenta que se lea la información remitida por la página, adecuando el mensaje entregado en un arreglo tipo *char*, es decir, una matriz de caracteres; luego, mediante una sentencia *if*, entendida como la comprobación de si una condición es verdadera o falsa, anidada en una estructura *Loop* la cual permite realizar bucles que se

repite mientras una condición dada sea cierta, esto es cuando la condición no se cumpla, se revisa el mensaje emitido desde la página *Web*. Si la posición 2 es leída, puede ocurrir uno de dos estados:

- **Primer estado:** el caracter R, es activado. Se envía el estado actual de los dispositivos conectados en el Arduino.
- **Segundo estado:** el caracter L, es activado. Se verifica el valor de la posición 0 del arreglo para prender o apagar el pin de acuerdo con el valor que se leyó. Si se solicita el estado de los dispositivos conectados, envía un mensaje haciendo uso de la función "*client.write*", la cual es leída por el *Socket* mediante la función "*socket_read*". Según el valor enviado, se actualiza el estado del bombillo y la página *Web*. Si no puede establecer conexión con el Arduino, se despliega un mensaje de error.

Una vez se termina de desarrollar el código, se procede a su compilación; si en este proceso no se presenta ningún tipo de error, se efectúa la ejecución del programa en el dispositivo y se realiza cualquiera de las dos opciones: prender o apagar.

Para que el Arduino sea accedido desde Internet, es necesario modificar la tabla *Network Address Translation* (NAT) del enrutador, concibiendo un redireccionamiento de puertos. De esta forma, las peticiones que lleguen al *Internet Protocol*, es decir, la dirección IP pública del Arduino, son redireccionadas a la IP privada del Arduino *Ethernet Shield* y trabajan de forma coordinada.

4. RESULTADOS DEL DESARROLLO

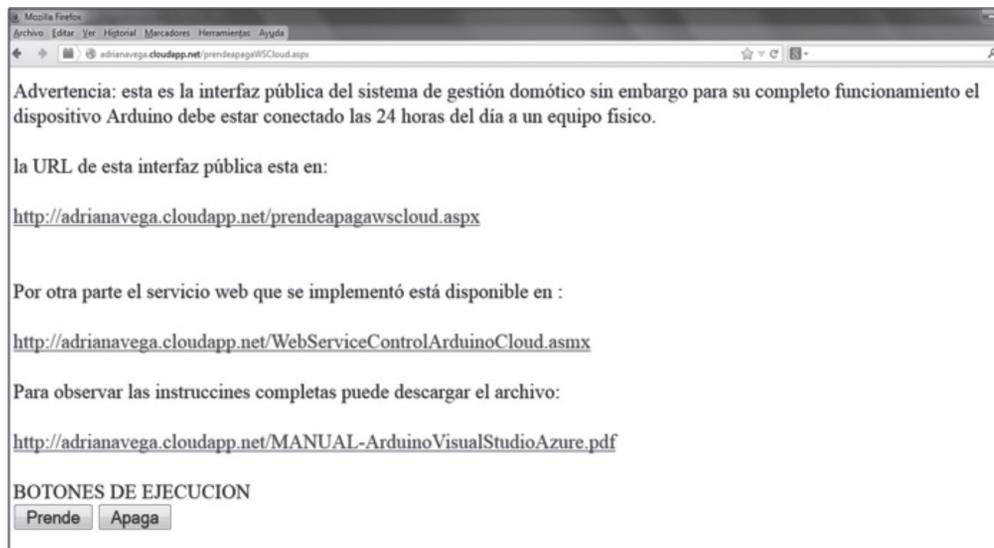
4.1 Ejecución del programa desde el servidor Azure

La verificación del comportamiento del sistema de gestión remota de una bombilla empleando Arduino, se realiza digitando la dirección del servicio *Web* que se creó en el navegador, la cual para este caso es:

“<http://adrianaavega.cloudapp.net/prendeapagaWSCloud.aspx>”; allí, se observa la página de inicio donde están los mensajes (Figura 7).

- **Mensaje de advertencia:** mensaje que le muestra al usuario del sistema, que el dispositivo Arduino debe estar acoplado para su correcto funcionamiento en el computador. El mensaje dice: “esta es la interfaz pública del sistema de gestión domótico, sin embargo, para su completo funcionamiento el dispositivo Arduino debe estar conectado las 24 horas del día a un equipo físico”.
- **Mensaje de ubicación URL:** mensaje en donde se le indica la ubicación de la interfaz pública a los usuarios, URL: <http://adrianaavega.cloudapp.net/prendeapagawscloud.aspx>
- **Mensaje de ubicación del servicio Web:** mensaje que indica la dirección en donde se implementó el servicio *Web*, la cual está disponible en: <http://adrianaavega.cloudapp.net/WebServiceControlArduinoCloud.asmx>
- **Mensaje de instrucciones:** mensaje para observar el manual de cómo se desarrolló el proyecto de gestión domótica para prender y apagar una bombilla, el cual se descarga en el vínculo: <http://adrianaavega.cloudapp.net/MANUAL-ArduinoVisualStudioAzure.pdf>
- **Botones:** al final de la página, se observan los dos botones para prender y apagar la bombilla de manera remota.

Figura 7. Página de inicio



Al oprimir el botón “Prender” ubicado en la página principal, se invocan las transacciones del proyecto WebRole1 en donde están las respectivas direcciones

que se emplean. En este caso la del Arduino, invocando el método que enciende la bombilla; desde este dispositivo, se muestra el mensaje en la página de Internet (Figura 8).

Figura 8. Mensaje del Arduino donde muestra que el bombillo esta prendido.

Se trabaja para este ejercicio del Doctorado con el Bit Digital
Nº. 2 Prueba encendido y apagado LED y bombillo.

El LED y el bombillo si esta conectado a una toma deben estar
ENCENDIDOS.
Favor verifique la operación **ENCENDIDO.**

**NOTA: ESTA PÁGINA SE ENVIA DESDE EL SERVIDOR HTTP
IMPLEMENTADO EN EL ARDUINO.**
Observar la dirección IP: 192.168.0.3:8080.
Que es establecido para el ARDUINO.
Esta página es direccionada a través del **SERVICIO WEB.**

**Para APAGAR EL LED Y EL BOMBILLO, pulse la opción (<..)
Volver atrás en el navegador.**

Una vez el mensaje es mostrado en la página de Internet, la bombilla se prende por el dispositivo Arduino de manera remota efectuando así la gestión domótica (Figura 9). De otro lado, si el usuario desea apagar la

bombilla, en la página de inicio se oprime el botón de "Apagar" y se despliega la siguiente página indicando que el dispositivo efectuó la función de apagar; adicional al mensaje, el bombillo es apagado (Figura 10).

Figura 9. Bombillo prendido consumiendo los servicios de Windows Azure®.

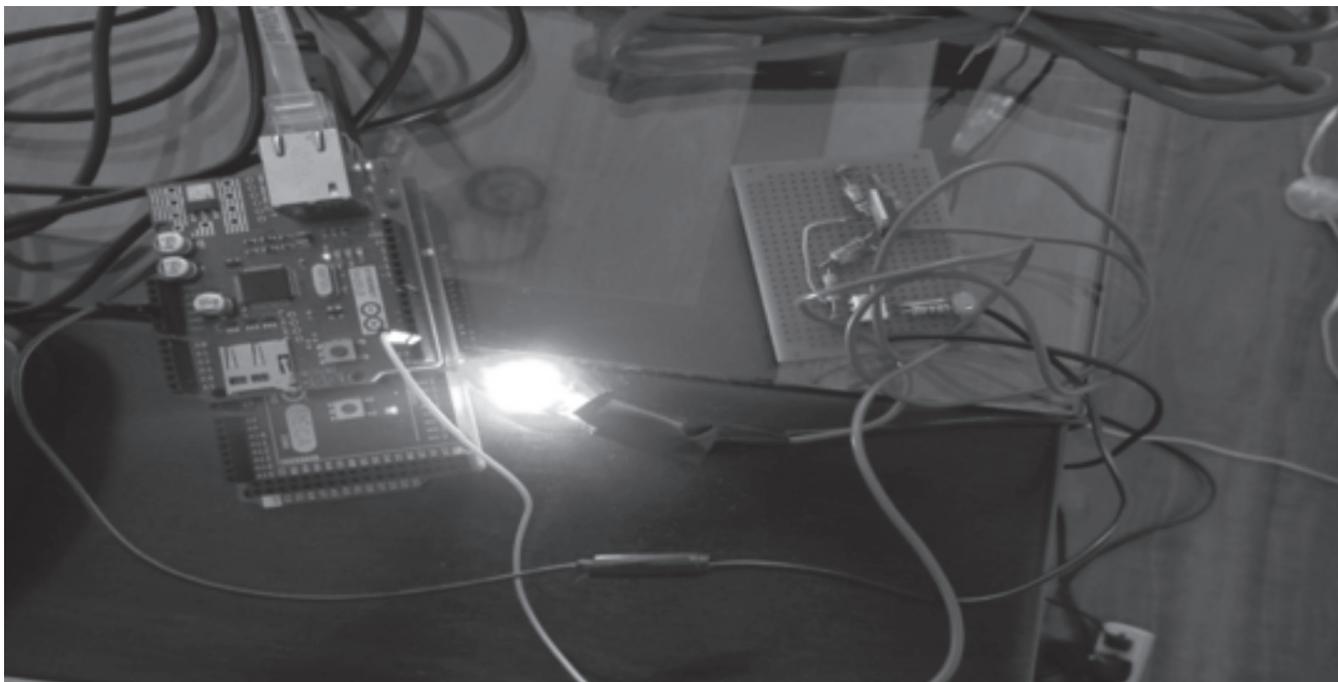


Figura 10. Mensaje del Arduino donde muestra que el bombillo está apagado.

**Se trabaja para este ejercicio del Doctorado con el Bit Digital
Nº. 2 Prueba encendido y apagado LED y bombillo.**

**El LED y el bombillo si esta conectado a una toma deben estar
APAGADOS.
Favor Verifique la operación APAGADO.**

**NOTA: ESTA PÁGINA SE ENVIA DESDE EL SERVIDOR HTTP
IMPLEMENTADO EN EL ARDUINO.
Observar la dirección IP: 192.168.0.3:8080.
Que es establecido para el ARDUINO.
Esta página es direccionada a través del SERVICIO WEB.**

**Para ENCENDER EL LED Y EL BOMBILLO, pulse la opción (<..)
Volver atrás en el navegador.**

Con la puesta en funcionamiento de este prototipo, se constata la importancia que tiene el Internet de los objetos a través de la manipulación de dispositivos ubicados en diferentes lugares, por medio de las Tecnologías de la Información y las Comunicación, lo cual permite a los usuarios una mejora en el confort de su

vida cotidiana teniendo el control permanente de los elementos que hacen parte habitual de su rol, es decir, el prender y apagar a distancia una bombilla no solo permite su manejo, sino también un uso racional de energía empleándolo cuando realmente se requiere.

5. CONCLUSIONES

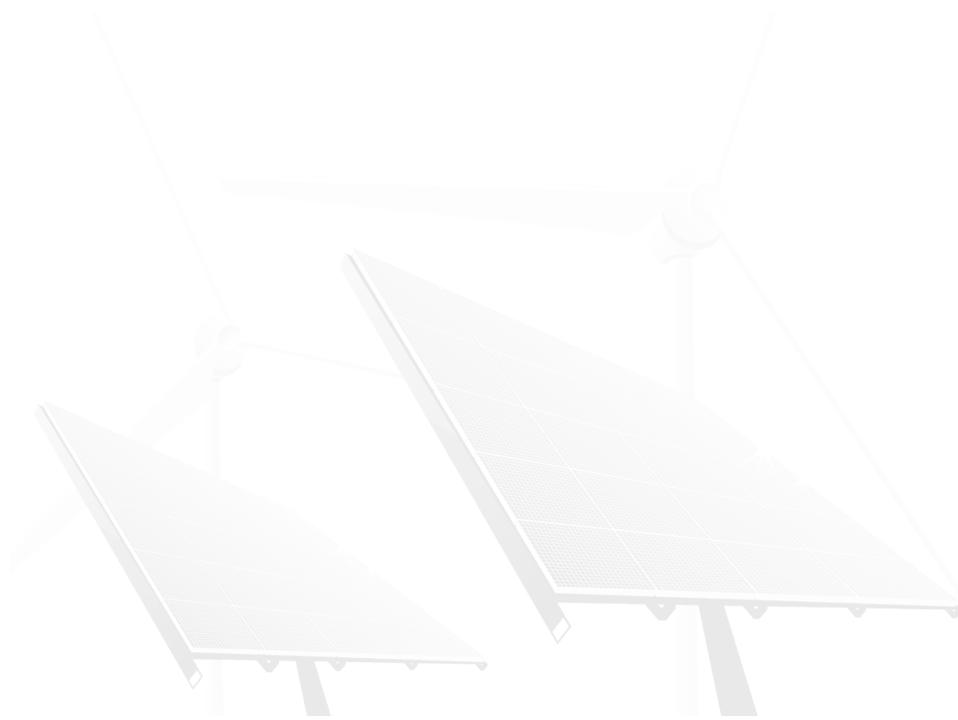
La gestión de la energía de los dispositivos eléctricos ubicados en las casas, se ejecuta con facilidad haciendo uso de interfaces de programación como el Arduino, el cual permite la conexión entre cualquier electrodoméstico que se quiera controlar remotamente por medio de comunicación vía Internet y haciendo uso de sus puertos de conexión.

Los usuarios de energía residenciales, podrán aportar a la sostenibilidad de este recurso interviniendo directamente desde cualquier lugar y en todo momento sobre el control de sus electrodomésticos.

Los sistemas domóticos son flexibles y se implementan con el Arduino en varios frentes, como son los servicios de iluminación, aires acondicionados, electrodomésticos, alarmas, cámaras, etc. Las unidades son programadas y reprogramadas para su optimización en cualquier momento y con facilidad, como se mostró en este caso práctico y de acuerdo a las necesidades de los usuarios sin tener ningún conocimiento técnico para operarlo.

La plataforma de Windows Azure® permite entre sus diversos escenarios, introducir servicios en la nube con diferentes lenguajes de programación. Se evidenció que la publicación en ese servidor, se desarrolló de manera rápida y sencilla, asegurando que su ejecución se realizara conforme a las necesidades de los usuarios; además, los costos de uso fueron bajos teniendo en cuenta su potencial de uso.

Con este prototipo de sistema de gestión de energía, se observó que el Internet de los objetos es algo que se está involucrando permanentemente con las personas las cuales no pueden estar descontextualizadas. Así, las residencias estarán inmersas en todas estas tecnologías unidas a las comunicaciones, que deberán estar soportadas y respaldadas por servicios ofrecidos en la nube para una mayor comodidad y gestión desde cualquier lugar, es decir, que la conexión del universo físico con el universo virtual se hará cada día mas corta.



6. REFERENCIAS BIBLIOGRÁFICAS

- Arduino. (2013). Arduino. Recuperado de <http://arduino.cc/es/Main/ArduinoEthernetShield>.
- Berenguer, M., Giordani, M., Giraud, F., & Noury, N. (2008). *Automatic detection of activities of daily living from detecting and classifying electrical events on the residential power line*. E-health Networking, Applications and Services, 10th International Conference., 29 - 32.
- De Silva, L., Morikama, C., & Petra, I. (2012). *State of art of smart homes*. *Engineering Applications of Artificial Intelligence*. Elsevier, 25, 1313 - 1321.
- Foros de electrónica. (2014). *Relé de estado sólido*. Recuperado de <http://www.forosdeelectronica.com/proyectos/rele-estado-solido.htm>.
- Fundación de la Innovación - Bankinter. (2011). *El internet de las cosas - En un mundo conectado de objetos inteligentes* (s.i), (s.e).
- González, R. (2014). *Conceptos básicos de enrutamiento*. Recuperado de: <http://www.ditae.uat.edu.mx/ver2013/xtras/ConceptosBasicosRoutingRicardo.pdf>
- Harney, A. (2008). *Smart Metering Technology Promotes Energy Efficiency For A Greenerworld*. *Analog Devices*.(s.i) (s.e).
- Jarman, P., Hooton, R., Walker, L., & Wang, Z. (2010). *Transformer Life Prediction Using Data From Units Removed From Service and Thermal Modelling*. (s.i), (s.e).
- Microsoft. (2014). Microsoft. Recuperado de <http://www.microsoft.com/visualstudio/esn/visual-studio-2013>
- Miorandi, D., Sicari, S., De Pellegrini, F., & Chlamtac, I. (2012). *Internet of things: Vision, applications and research challenges*. *Ad Hoc Networks*, 10, 1497-1516.
- Moore, R. (2010). Ethernet For IEC61850. *Pac World*, 13.
- Nowak, S., Schaefer, F., Brzozowski, M., Kraemer, R., & Kays, R. (2011). *Towards a convergent digital home network infrastructure*. 57, 1695 - 1703.
- Passos da Costa, P., & Rosado, A. (2012). *Migration to Windows Azure – Analysis and Comparison*. *Conference on Health and Social Care Information Systems and Technologies*. Elsevier Procedia Technology.
- Schmidt, M. (2011). *Arduino: A Quick Start Guide*. Elsevier Pragmatic Bookshelf, 201.
- Snyder, A., Gunther, E., & Griffin, S. (2012). *The smart grid homeowner: An IT guru?* Future of Instrumentation International Workshop (FIW), 1 - 4.
- The Cambridge. (2012). *The Cambridge Dictionary of Statistics*. (s.i) (s.e.).
- Tolosa, G. (2014). *Protocolos y Modelo OSI*. Recuperado de <http://www.tyr.unlu.edu.ar/TYR-publica/02-Protocolos-y-OSI.pdf>.
- Windows Azure® . (2014). *Windows Azure®* . Recuperado de <http://www.windowsazure.com/es-es/>