

Control de consumo eléctrico residencial automatizado

Control of automated residential electrical consumption

Romario Pitti¹, Carlos Aguilar¹, Einar Pérez², Victoria Serrano^{3*}

¹ Licenciatura en Ingeniería Electrónica y Telecomunicaciones - Centro Regional de Chiriquí – Universidad Tecnológica de Panamá,

² Licenciatura en Ingeniería Eléctrica Electrónica – Centro Regional de Chiriquí – Universidad Tecnológica de Panamá

³ Facultad de Ingeniería Eléctrica - Centro Regional de Chiriquí - Universidad Tecnológica de Panamá

Resumen En este estudio se presenta el desarrollo de un sistema automatizado para disminuir el consumo de energía eléctrica residencial. Se utilizó la técnica Charlieplexing de una matriz para minimizar el uso de pines en el microcontrolador utilizado. Además, se aplicó un sistema de interrupción no convencional de TRIAC en sustitución al relay que es el tipo de componente comúnmente utilizado para controlar el flujo de corriente en tomacorrientes. Se logró demostrar que mediante este método es posible abaratar los costos en la implementación del circuito al igual que se puede llevar a cabo de manera confiable en cualquier tipo de residencia, sin importar la cantidad de tomacorrientes que esta posea en su instalación eléctrica.

Palabras claves Consumo eléctrico, domótica, control de flujo eléctrico, Charlieplexing.

Abstract This study presents an automated system to reduce the consumption of electricity at home. Charlieplexing technique of a matrix was implemented to minimize the number of pins used on the microcontroller. In addition, an unconventional interrupt system of TRIAC was applied to replace relays which are the commonly used component to control current flow in electrical outlets. It was demonstrated that this method can reduce costs in the circuit implementation. At the same time, it can be reliably implemented in any type of residence, regardless of the number of outlets it has in its electrical installation.

Keywords Energy consumption, home automation, energy flow control, Charlieplexing.

* Corresponding author: victoria.serrano@utp.ac.pa

1. Introducción

¿Podríamos ahorrar energía con el hecho de controlar los tomacorrientes? Conocemos que la energía es un recurso indispensable para nuestras vidas, en nosotros depende su debida utilización y conservación. En ocasiones nuestro consumo eléctrico, en los hogares resulta ser mayor, debido a un uso inadecuado; trayendo en consecuencia un incremento en nuestra facturación mensual.

Con el paso del tiempo, el desarrollo de la tecnología ha permitido mejorar la instalación y distribución de la energía; así como creación de métodos que aporten una disminución en el gasto [1]. Se debe tener claro que, para poder desarrollar un ahorro en el consumo, se debe conocer primeramente el gasto que realizamos [2]. Con la implementación de un sistema de control se logra una eficiencia energética automatizada encaminada al bienestar de las personas [3].

En la actualidad existen métodos para el control de energía en nuestros hogares, como el diseño de medidores prepagos [1]. También hay sistemas más modernos de control a aspectos específicos en los hogares como las bombillas [4]

o bien controles en equipos específicos para cuidar su resistencia [5].

El presente estudio propone un sistema de control de los tomacorrientes utilizando la técnica Charlieplexing [6]; que permite reducir la cantidad de pines de entrada y salida utilizados en una tarjeta Arduino [7]. Aunque la técnica Charlieplexing ha sido utilizada para controlar una matriz de diodos emisores de luz (LEDs), nunca se ha aplicado para el control de consumo eléctrico residencial. Adicional a la tarjeta Arduino se utilizó un TRIAC, que es un dispositivo que permite abrir y cerrar los circuitos de corriente para así tener control sobre los tomacorrientes [8].

Finalmente, con estos métodos presentados se busca que el usuario tenga la facilidad de apagar o encender, desde un punto en el hogar, los accesos de electricidad y se pueda controlar de manera general el flujo de corriente del hogar cuando no se esté presente con dispositivos de bajo costo.

Este trabajo se desarrolló de la siguiente manera:

- Los materiales que fueron empleados en la confección del proyecto se describen en la Sección 2.

- La Sección 3 detalla la descripción del método innovador que se utilizó para este proyecto.
- En la Sección 4 se muestra cómo se aplicó la técnica para el desarrollo del sistema.
- En la Sección 5 se da a conocer información referente al consumo fantasma y los datos recolectados para mostrar el grado de beneficio que puede crear el sistema.
- En la Sección 6 se detalla la explicación de la programación que se realizó con la tarjeta Arduino.
- Finalmente, la Sección 7 hace referencia a las conclusiones derivadas de este estudio.

2. Materiales

Los materiales utilizados en el desarrollo de este estudio fueron los siguientes:

- Arduino Mega2560
- Optoacoplador
- TRIAC (Triodo de Corriente Alterna)
- Resistencias
- Capacitor
- Sensor de corriente no invasivo
- Tomacorrientes

3. Técnica Charlieplexing

Charlieplexing es una técnica que se propuso a principios de 1995 por Charlie Allen de Maxim Integrated Products, para controlar una pantalla multiplexada en la que los pines de entrada y salida (I/O), son escasos [6].

El método se enfoca en las capacidades de lógica tri-estado de microcontroladores con el objetivo de optimizar la eficiencia sobre multiplexación tradicional. A pesar de que es más útil en el uso de I/O, hay situaciones que hacen que sea más complejo para diseñar y hacer que sea imposible su implantación para pantallas grandes.

La técnica Charlieplexing fue utilizada para la automatización del consumo eléctrico ya que se busca reducir los costos del sistema al implementar una tarjeta más económica como la Arduino mini, que provee un número menor de pines de entrada y salida.

La técnica presenta ventajas en el sentido que se pueden controlar más tomacorrientes utilizando menos pines en la tarjeta Arduino. Para comprenderlo mejor, se hace referencia a la ecuación 1 en donde la cantidad de pines a utilizar está representada por la variable “x”, mientras que el resultado provee la cantidad de tomacorrientes que se puede controlar. Por ejemplo, teniendo tres pines como se muestra en la ecuación 2, indica que se pueden controlar seis tomacorrientes. En la tabla 1 se muestra la cantidad de tomacorrientes que se pueden controlar dependiendo de la

cantidad de pines de salida de la tarjeta Arduino hasta un máximo de siete pines.

$$N^{\circ} \text{ tomacorrientes} = x(x - 1) \quad (1)$$

Tabla 1. Cantidad de tomacorrientes que se puede usar por pines

N.º de Pines utilizados en la tarjeta Arduino	N.º Tomacorrientes que se pueden controlar
1	0
2	2
3	6
4	12
5	20
6	30
7	42

Adicionalmente, en la figura 1 se observa el número de tomacorrientes que pueden ser controlados con las tarjetas Arduino al utilizar distintos número de pines.

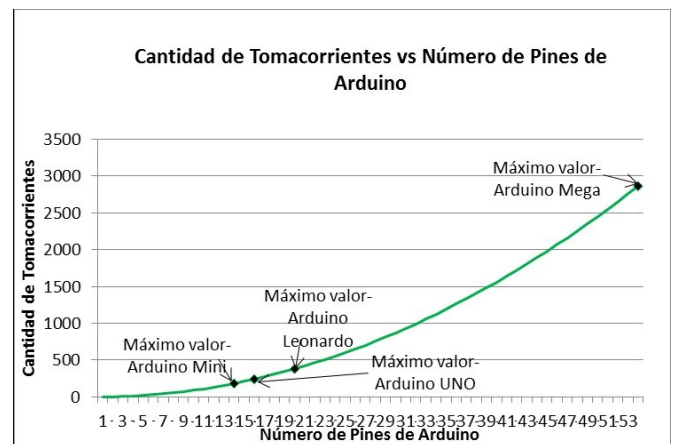


Figura 1. Relación que muestra la cantidad de tomacorrientes que pueden controlarse de acuerdo al número de pines de diferentes tarjetas Arduino aplicando la técnica Charlieplexing.

3.1 Beneficios de la Técnica Charlieplexing

A continuación, se presenta las ventajas que se obtienen al utilizar la técnica Charlieplexing. En la tabla 2 se muestra una comparación de los distintos microcontroladores que se pueden utilizar en el sistema. Por lo tanto, se puede obtener un ahorro ya que al implementar la técnica Charlieplexing se utilizan menos pines, lo que permite seleccionar un microcontrolador de menor tamaño y precio.

Tabla 2. Comparación de los diferentes tipos de tarjetas Arduino

Modelo	I/O digitales	Entradas analógicas	Salidas PWR	Memoria	Precio
Uno REV3	16	6	6	32kb	\$24.95
Leonardo	20	12	7	32kb	\$20.00
Mega REV3	54	16	14	256kb	\$45.95
MINI 05	14	6	8	32kb	\$15.40

Con los datos proporcionados en la figura 1 se observa el beneficio que produce la técnica Charlieplexing al reducir el número de pines utilizados en la tarjeta Arduino para controlar un número mayor de tomacorrientes. Pero, ¿qué pasaría si no se utiliza la técnica Charlieplexing? La figura 2 muestra una comparación del número de tarjetas Arduino que serían necesarias para manejar el máximo número de tomacorrientes para cada tarjeta mostrada en la figura 1. Para ello, sería necesario cuantificar el número de tomacorrientes e igualarlo al máximo número de pines de la tarjeta Arduino. De este modo, se obtienen que para controlar 182 tomacorrientes se necesitarían 13 Arduino Mini, para 380 tomacorrientes se necesitarían 15 Arduino Uno, para 240 tomacorrientes serían necesarios 380 Arduino Leonardo y para 2862 tomacorrientes se requieren 51 Arduino Mega.

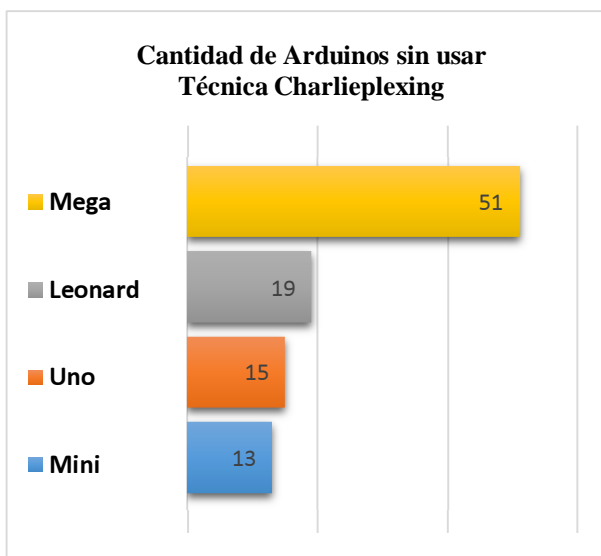


Figura 2. Cantidad de Tarjetas Arduino necesarias para controlar el máximo número de tomacorrientes, sin utilizar la técnica Charlieplexing.

Se puede observar que cada tarjeta cuenta con una disposición de pines para controlar los tomacorrientes. Al incrementarse la cantidad de los tomacorrientes a controlar, se incrementa asimismo la cantidad de tarjetas al no aplicar la técnica Charlieplexing. Por lo tanto, la técnica Charlieplexing

nos permite maximizar el número de conexiones a controlar con menos pines disponibles.

Esto indiscutiblemente crea un ahorro muy significativo si se busca controlar un gran número de tomacorrientes. Se ve como ejemplo la tarjeta Arduino Mega, que una sola de ella utilizando la técnica Charlieplexing equivaldría a 51 tarjetas Arduino Mega sin aplicar la técnica.

Esto, además, repercute en un ahorro económico al implementar el control de consumo eléctrico con la técnica Charlieplexing. La figura 3 muestra una comparación de lo que costaría implementar un sistema para el máximo número de pines de cada tarjeta Arduino con y sin la técnica Charlieplexing.

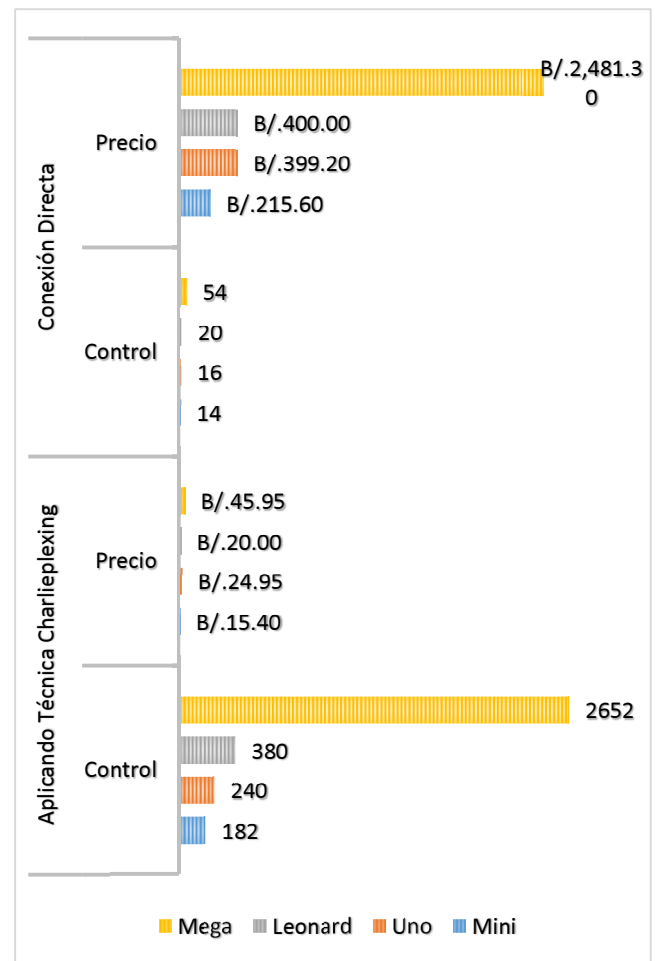


Figura 3. Rentabilidad económica del uso de la técnica Charlieplexing.

3.2 Optoacoplador

Solo es posible lograr la técnica Charlieplexing gracias a la forma en la que están compuestos los diodos. En los pines digitales de Arduino, solo se puede tener dos estados: alto (HIGH) o bajo (LOW). La técnica Charlieplexing consiste en utilizar un tercer estado que permita seleccionar un LED específico. Esto se puede hacer gracias a la naturaleza de los Leds.

LED, es el acrónimo de Light-Emitting Diode (diodo emisor de luz) y por lo tanto se comporta como lo que es, un diodo.

De esta manera, se requería un dispositivo que fuese un diodo para poder aplicar la técnica al control del flujo de corriente en los tomacorrientes; por lo tanto, se utilizó un optoacoplador, como se muestra en la figura 4 en donde se puede apreciar que el optoacoplador está compuesto por diodo.

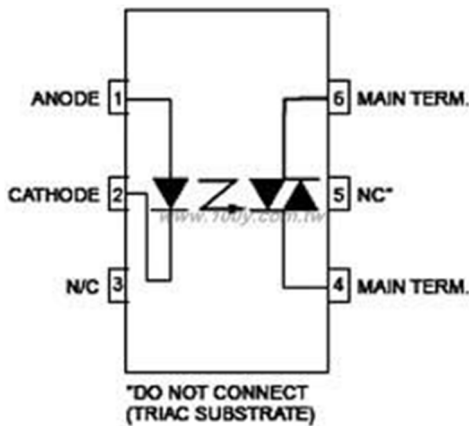


Figura 4. Esquema electrónico.

El optoacoplador es un componente que transmite y recibe corriente, para funcionar como un interruptor por medio de la luz emitida por un diodo LED que satura un componente optoelectrónico, comúnmente en forma de fototransistor o fototriac [9].

3.3 TRIAC (Triodo de Corriente Alterna)

Debido a que para controlar el flujo de corriente se necesita un tipo de interruptor se pensó utilizar un relevador, pero debido a sus desventajas como: demora en el tiempo de conmutación y por su desgaste mecánico no se considera conveniente utilizarlo en este sistema. Por eso, en su lugar, se tomó la decisión de utilizar un Triac.

Un TRIAC es un interruptor capaz de controlar la corriente alterna. Además, es un componente semiconductor. Posee tres electrodos y el flujo de energía se activa aplicando una descarga eléctrica al tercer electrodo [10].

4. Sistema Físico

Como ya antes se ha explicado se utiliza la técnica Charlieplexing como método innovador en el control de consumo eléctrico, junto a un Triac como dispositivo interruptor. De esta forma se puede interrumpir o no el flujo de corriente y conseguir un control en el consumo eléctrico. En la figura 5 se puede observar el esquema básico del sistema, donde se utilizaron focos para explicar el funcionamiento de la matriz.

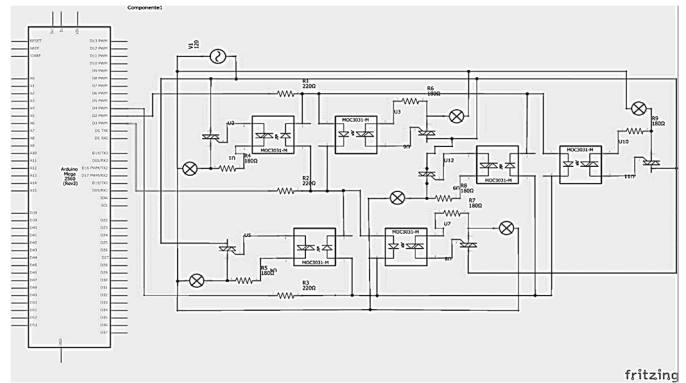


Figura 5. Esquema del circuito.

El circuito utiliza optoacopladores MOC3022 y TRIAC BT136 que es el indicado para un nivel de voltaje de 120VAC que circulan normalmente en una instalación eléctrica residencial. Las conexiones están hechas de modo tal que se puedan controlar cada uno de forma independiente como se muestra en la figura 6.

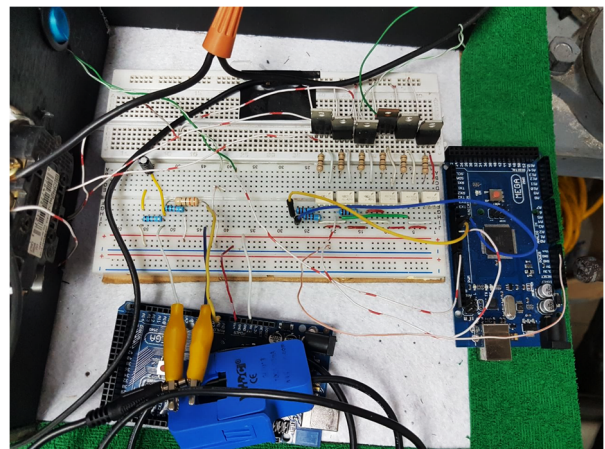


Figura 6. Circuito del sistema.

Todo se desarrolla empleando un modelo a escala de un hogar, utilizando tomacorrientes de uso común y un bombillo de 18W que representa la nevera de la vivienda como se observa en la figura 7.



Figura 7. Maqueta del hogar.

5. Consumo fantasma

El consumo fantasma se produce al tener los dispositivos eléctricos conectados al suministro de la red eléctrica sin utilizarlos o se encuentran en modo “Stand by”. Según estudios en España proporcionados por el Instituto para la Diversificación y Ahorro de la Energía (IDAE), se señala que anualmente esto refleja de un 7% al 11% de la facturación [11].

Para poder obtener la información de datos del consumo fantasma que fluctúa en hogares a nuestro entorno, se utilizó un sensor de corriente modelo SCT-013 para así medir la corriente alterna.

5.1 Calibración del sensor para uso con la tarjeta Arduino

El sensor mide una señal alterna, esto quiere decir que el rango puede tomar valores negativos o positivos. Sin embargo, la tarjeta Arduino solo puede leer voltajes alternos positivos. Por lo tanto, se realizó una configuración electrónica que pudiese transformar corriente alterna a voltajes positivos en un rango de (0 a +5V), que recibe la entrada analógica del Arduino [12].

Para realizar esta calibración es necesario conocer primeramente el pico máximo de corriente que el sensor puede manejar. Se sabe que el sensor SCT-013 está diseñado para medir rangos hasta 100A AC.

La corriente alterna posee una onda de forma senoidal, tomando el valor de 100A como valor eficaz, se calcula la corriente media mediante la ecuación (3).

$$\begin{aligned} \text{corriente media} &= \sqrt{2} * \text{valor eficaz} & (3) \\ &= \sqrt{2} * 100A \\ &= 141.4 A \end{aligned}$$

Para conocer la salida de corriente del sensor, se necesita conocer el número de vueltas del embobinado del sensor. Así pues, el sensor de corriente cuenta con 2000 vueltas. Por lo tanto, la corriente del sensor se determina por la ecuación (4).

$$\begin{aligned} \text{corriente del sensor} &= \frac{\text{corriente media}}{n^{\circ} \text{ vueltas}} & (4) \\ &= \frac{141.4}{2000} \\ &= 0.0707 A \end{aligned}$$

Ahora bien, teniendo estos datos se puede calcular la resistencia más adecuada que podría tener el circuito por medio de la ecuación (5).

$$\begin{aligned} \text{resistencia} &= \frac{\text{voltaje del sensor}}{\text{corriente del sensor}} & (5) \\ &= \frac{2.5 V}{0.0707 A} \\ &= 35.4 \Omega \end{aligned}$$

La resistencia más adecuada debe ser de 35 Ω, pero al no existir una resistencia de este valor, se utilizó una de 33 Ω.

Además, se añaden 2 resistencias de 470 k Ω, para que el nivel de voltaje se encuentre dentro del rango de voltaje medible por el Arduino (0-5V) [13]. La figura 8 muestra el circuito implementado para utilizar el sensor de corriente con la tarjeta Arduino.

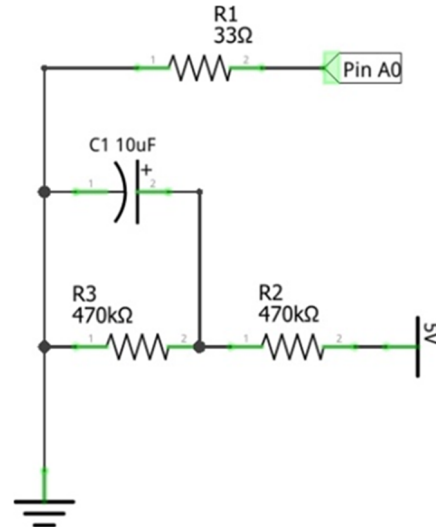


Figura 8. Diseño del circuito para el sensor SCT-013.

5.2 Recolección de datos de equipos eléctricos

Después de tener el sensor debidamente calibrado, se procedieron a tomar las siguientes medidas mostradas en la tabla 3. Estas medidas se estiman en un hogar común. Los datos pueden variar en distintos hogares debido a la cantidad de equipos que se tengan y los tipos de marcas. El consumo fantasma se puede observar como un valor pequeño a comparación a un consumo regular, pero este termina representando un porcentaje similar en todos los hogares.

Tabla 3. Equipos en Stand by

Equipo Eléctrico	Consumo Fantasma (KWh)
Impresora	0.00100
Aire Acondicionado	0.00100
Televisor	0.00306
computadora de mesa	0.00284
Caja de cable	0.00510
Microondas	0.00308
Laptop	0.00030
Equipo de sonido	0.00060
Cargador Laptop	0.00075
Cafetera	0.00080
Cargador Celular	0.00026
Abanico	0.00040
Consola	0.00060
Total	0.01979

Los equipos electrónicos mencionados en la tabla 3 son los que habitualmente se encuentran en una casa. Considerando una unidad de cada equipo se determinó que el consumo fantasma reflejaba un 0.01979 Kilo-Watt por hora (KWh). Ahora bien, si se asume que estos equipos pasan toda la noche conectados sin ser utilizados así como parte del día, y que en general dentro de un período de 24 horas ellos permanecen 9 horas en promedio sin ser usados. Esto equivaldría a un total 3285 horas al año sin ser utilizados y generan un consumo fantasma como se muestra en la ecuación (6).

$$\text{Consumo Fantasma Anual} = (0.01979 \text{ kw})(3285 \text{ h}) \quad (6)$$

$$= 65.01 \text{ kwh}$$

Donde 65.01 kwh sería la cantidad de energía perdida al tener los equipos conectados, pero sin estar siendo utilizados. Conociendo el consumo de los equipos en modo Stand-by, ahora se presenta una Tabla 4 que registra el consumo general de una casa por todo un año en el área de distribución de Chiriquí.

Tabla 4. Consumo anual de un hogar en el área de distribución de EDECHI.

Consumo	
	kwh
Agosto	144
Septiembre	113
Octubre	118
Noviembre	102
Diciembre	118
Enero	142
Febrero	133
Marzo	159
Abril	152
Mayo	137
Junio	140
Julio	135
Total	1593

Conociendo ya finalmente el consumo fantasma y el consumo general del hogar se puede determinar el porcentaje que refleja el consumo fantasma como lo muestra la ecuación (7).

$$\% \text{ Consumo Fantasma} = \frac{\text{consumo fantasma}}{\text{consumo general}} \quad (7)$$

$$= \frac{65.01 \text{ kwh}}{1593 \text{ kwh}} * 100\%$$

$$= 0.04081 * 100\%$$

$$= 4.081\%$$

Es decir, 4.081% es el porcentaje que equivaldría al consumo fantasma anual. Dicho de otra manera, es lo que se paga sin ser utilizado. Sin embargo, es importante recordar que este porcentaje puede variar dependiendo del número de equipos que se tengan conectados en casa.

5.3 Costo de la energía en Panamá

Los datos proporcionados por Gas Natural Fenosa, muestran el cambio sobre el valor del kwh mensual desde el 2015 al presente año. Estos datos se presentan cada 6 meses por medio de un pliego tarifario [14].

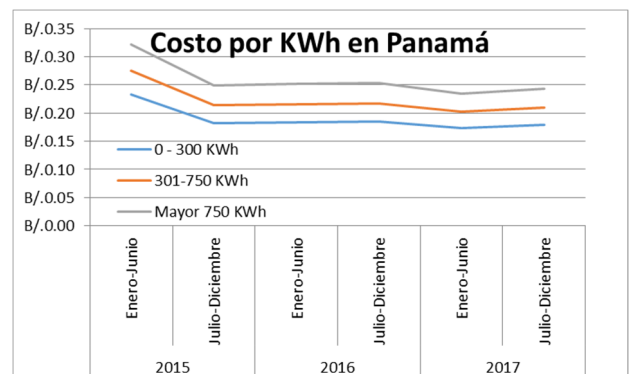


Figura 9. Costo de la energía eléctrica en Panamá. Estos valores se clasifican en tres tipos dependiendo del consumo que tenga el consumidor.

Por lo tanto, se puede observar el costo de la energía eléctrica y la necesidad de buscar medidas de ahorro. Se destaca también que estos precios son dirigidos al área de cobertura en la provincia de Chiriquí.

Al realizar el análisis de datos para el consumo eléctrico residencial detallado anteriormente, se puede determinar en dólares la cantidad de dinero que se pierde al tener conectados los dispositivos. Siendo B/. 0.17993 el precio actual a pagar por kwh actualmente, el consumo anual está dado por la ecuación (8).

$$\text{Costo anual} = 1593 \text{ kwh} * 0.17993 \quad (8)$$

$$= \text{B}/.286.63$$

La cantidad de B/.286.63 es el costo al año en electricidad y dentro de ese gasto el 4.081% representa el consumo fantasma, que puede calcularse en la ecuación (9).

$$\text{Consumo fantasma} = 286.63 * 4.081\% \quad (9)$$

$$= \text{B}/.11.70$$

Anualmente se está perdiendo un total de B/.11.70 por solamente dejar los dispositivos conectados. En la figura 10 se muestra el costo del consumo fantasma de 4.081% estimado para un consumo mensual de la casa en estudio, así como para 500KWh y 1000 KWh. Se puede observar que el mismo corresponde a un total de B/11.70, B/51.39 y

B/119.25, respectivamente. Este sería el dinero aproximado perdido por año para dichos consumos energéticos.

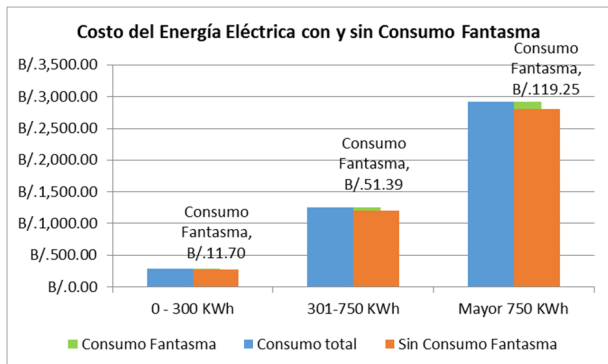


Figura 10. Costo de la energía eléctrica con y sin consumo fantasma.

6. Código

El código consta de una función básica de *switch* para el control de la matriz, como se muestra a continuación. Con el uso de este código se pudo ejecutar de manera sencilla el flujo de energía de los tomacorrientes.

```
// En esta parte se definirán los estados de los pines para
// encender de manera sencilla los diferentes interruptores
// Función que pondrá en los estados correctos para
// encender un LED (HIGH, LOW e INPUT)
void ponerEstados(int pinHigh, int pinLow, int pinInput)
{
  pinMode(pinHigh,OUTPUT);
  digitalWrite(pinHigh,HIGH);
  pinMode(pinLow,OUTPUT);
  digitalWrite(pinLow,LOW);
  pinMode(pinInput,INPUT);
}
// Esta función se va a encargar de aplicar la lógica
// dependiendo del LED que queramos encender
void encenderLed(int led_num)
{
  switch(led_num)
  {
  case 1:
    ponerEstados(PIN_B,PIN_A,PIN_C);
    break;
  case 2:
    ponerEstados(PIN_A,PIN_B,PIN_C);
    break;
  case 3:
    ponerEstados(PIN_C,PIN_B,PIN_A);
    break;
  case 4:
```

```
    ponerEstados(PIN_B,PIN_C,PIN_A);
    break;
  case 5:
    ponerEstados(PIN_C,PIN_A,PIN_B);
    break;
  case 6:
    ponerEstados(PIN_A,PIN_C,PIN_B);
    break;
  }
}
```

También se utilizó la función de decisión *if-else* como se muestra a continuación controlar el encendido y apagado del sistema con el interruptor. De esta manera se puede modificar en un futuro de manera sencilla el programa.

```
void loop()
{
  digitalWrite(13,HIGH); // El led integrado confirmara el
  // inicio del programa
  if ( digitalRead (pulsador) == LOW) // Si el pulsador1
  // está presionado
  {
    encenderLed (1);
  }
  if (digitalRead (pulsador) ==HIGH)
  {
    encenderLed (1);
    encenderLed (2);
    encenderLed (3);
    encenderLed (4);
    encenderLed (5);
    encenderLed (6);
  }
}
```

Adicionalmente, se utilizó aparte una programación para el uso del sensor de corriente como se muestra a continuación:

```
#include "EmonLib.h" // Include Emon Library
EnergyMonitor emon1; // Create an instance
float watt;
float amp;
float it;
void setup()
{
  Serial.begin(9600);
```

```

emon1.current(0,60);    // Current: input pin,
calibration.
//calibration is explained bellow
}

void loop()
{
  double Irms = emon1.calcIrms(1220); // Calculate Irms
  only
  it= (Irms)-(0.20);
  watt=(Irms)*(110.0);
  amp=it;

  if(watt<0 || amp<0){
    watt=0;
    amp=0;
    Serial.print(watt);    // Apparent power
    Serial.print(" ");
    Serial.println(amp,4);    // Irms
    delay(1000);
  }
  else
    Serial.print(watt);    // Apparent power
    Serial.print(" ");
    Serial.println(amp,4);    // Irms
    delay(1000);
  }
}

```

7. Conclusiones

En este trabajo se logró aplicar el método de Charlieplexing para el control del consumo eléctrico residencial. Este es un método originalmente utilizado para controlar múltiples LED'S, con una menor cantidad de pines de un controlador. La técnica Charlieplexing nos facilita el control del flujo de corriente en un circuito utilizando la programación Arduino. Este método fue utilizado en conjunto con optoacopladores y componentes TRIACS que trabajan de la misma forma que un interruptor. El diseño realizado logra específicamente que un usuario tenga el control del flujo de corriente en el hogar. Además, al minimizar el número de pines utilizados en el microcontrolador, se logra un ahorro económico en la implementación del sistema.

Como trabajo futuro se puede investigar cómo lograr controlar el sistema mientras se encuentra en el hogar de una manera más amigable al usuario, que no requiera que la persona aplique un nivel de programación compleja.

REFERENCIAS

- [1] A. Sánchez y C. Luengas, «Diseño y Construcción de un Prototipo para el Control de Consumo de Energía Eléctrica Residencial,» *Scientia Et Technica*, vol. 16, n° 44, 2010.
- [2] F. J. C. C. y J. G. Carrión, «SUPERVISIÓN ENERGÉTICA,» *3C Tecnología*, vol. 4, n° 1, 2015.
- [3] A. Jurenoks y D. Jokic, «Sensor Network Information Flow Control Method with Static,» *EL SEVIER*, 2016.
- [4] A. M. Vega, F. Santamaría y E. Rivas, «Empleando Arduino para la Gestión Eléctrica Domiciliaria,» *Revista EAN*, n° 77, pp. 22-41, 2014.
- [5] International Journal of Advanced Computer Science and Applications, «Reducing the Electrical Consumption in the Humidity Control Process for Electric Cells using an Intelligent Fuzzy Logic Controller,» *IJACSA*, vol. 7, n° 12, 2016.
- [6] A. N. 1880, «Maxim integrated,» 10 February 2003. [En línea]. Available: <https://www.maximintegrated.com/en/app-notes/index.mvp/id/1880>. [Último acceso: 10 September 2017].
- [7] L. d. V. Hernández, «Programarfacil.com,» [En línea]. Available: <https://programarfacil.com/blog/controlar-matriz-de-leds-con-arduino/>.
- [8] A. G. González, «Panama Hitek,» 20 Junio 2013. [En línea]. Available: <http://panamahitek.com/herramienta-de-control-para-arduino-el-triac-conmutador-para-corriente-alterna/>. [Último acceso: Mayo 2017].
- [9] Wikipedia, «Optoacoplador,» 21 Febrero 2017. [En línea]. Available: <https://es.wikipedia.org/wiki/Optoacoplador>.
- [10] Wikipedia, «Triac,» 21 Abril 2017. [En línea]. Available: <https://es.wikipedia.org/wiki/Triac>.
- [11] Xataka, «Consumo fantasma. Mitos, verdades y soluciones,» 29 Enero 2014. [En línea]. Available: <https://www.xataka.com/energia/consumo-fantasma-mitos-verdades-y-soluciones>. [Último acceso: 1 Septiembre 2017].
- [12] N. Mechatronics, «Tutorial sensor de corriente AC no invasivo SCT-013,» [En línea]. Available: http://www.naylampmechatronics.com/blog/51_tutorial-sensor-de-corriente-ac-no-invasivo-s.html. [Último acceso: 31 Agosto 2017].
- [13] V. Demay, «Homautomation,» 17 Sentiembre 2013. [En línea]. Available: <http://www.homautomation.org/2013/09/17/current-monitoring-with-non-invasive-sensor-and-arduino/>. [Último acceso: 12 Agosto 2017].
- [14] gasNatural Fenosa, «gasNatural Fenosa,» Gas Natural Fenosa, [En línea]. Available: <http://www.gasnaturalfenosa.com.pa/pa/grandes+clientes/distribucion+de+electricidad/1297100756887/tarifas.html>. [Último acceso: 8 Septiembre 2017].