

# Medición de energía en sistemas embebidos para el Internet de las Cosas

Alfonso Medina-Duran<sup>1</sup>, Mario Camarillo-Ramos<sup>2</sup>, Verónica Quintero-Rosas<sup>1</sup>,  
Arnoldo Díaz-Ramírez<sup>1</sup>, Francisco Ibañez-Salas<sup>1</sup>

Tecnológico Nacional de México/ Instituto Tecnológico de Mexicali, Mexicali B.C., México.

<sup>1</sup>Departamento de Sistemas y Computación, <sup>2</sup>Departamento de Eléctrica - Electrónica

## Resumen

Minimizar el consumo de energía en dispositivos con recursos limitados es necesario para el empleo del internet de las cosas, actualmente el alto costo de las plataformas de medición de energía es muy elevado, por lo que el uso de herramientas alternativas en la investigación es una realidad; pero estas son complejas, de baja precisión, rango de medición limitados, además estas herramientas existentes no son adecuadas para los nuevos sensores o dispositivos de tecnología para el 802.11.

El porcentaje de energía consumida por los dispositivos conectados aumenta debido al crecimiento significativo en el número de dispositivos del internet de las cosas.

Los fabricantes de microcontroladores proporcionan herramientas de simulación y medición para dispositivos donde el consumo de energía es optimizado tanto en software como hardware para obtener un valor de consumo menor. Si se desea realizar una comparativa del consumo de energía a través de diferentes fabricantes no es posible coincidir en una métrica, aunque el Embedded Microprocessor Benchmark Consortium (EEMBC) intenta hacerlo con su CoreMark pero deben tener ciertas características en los microcontroladores; una de ellas es tener un RTC interno y no todos los microcontroladores cuentan con este periférico. En este proyecto se desarrollara un sistema (hardware/software) automatizado de medición de la energía consumida por un dispositivo incrustado en Internet de las cosas.

## Abstract

Minimizing energy consumption in devices with limited resources is necessary for the use of the internet of things, currently the high cost of energy measurement platforms is very high, so the use of alternative tools in research is a reality ; but these are complex, low precision, limited measurement range, in addition these existing tools are not suitable for new sensors or technology devices for 802.11. The percentage of energy consumed by connected devices increases due to the significant growth in the number of internet devices of things. The internet of things devices work with batteries, so it is very important to profile and analyze the energy efficiency in these devices. The energy consumption increases as the activities carried out by these devices increase, such is the case of the Internet of things where it is not only necessary to perform the indicated task, the information must also be transferred wirelessly; the monitoring of vital signs in medical instruments where it is imperative to have reliable information and transfer it to possible actuators. These applications, among others, have turned energy consumption into a very important parameter for the design of these instruments. Microcontroller manufacturers provide simulation and measurement tools for devices where power consumption is optimized in both software and hardware to obtain a lower consumption value. If you want to make a comparison of energy consumption through different manufacturers it is not possible to agree on a metric, although the Embedded Microprocessor Benchmark Consortium (EEMBC) tries to do it with its CoreMark but they must have certain characteristics in the microcontrollers; One of them is to have an internal RTC and not all microcontrollers have this peripheral. This project will develop an automated system (hardware / software) for measuring the energy consumed by a device embedded in the Internet of things.

Palabras Clave: **Sistemas embebidos; Medición de energía; Internet de las cosas.**

Keywords: **Embedded systems; Energy measurement; Internet of things.**

## 1. INTRODUCCIÓN

Los microcontroladores y microprocesadores, sistemas inmersos o embebidos, se han convertido en un aspecto importante de la vida cotidiana. Se utilizan en la mayoría de las actividades que realizamos y probablemente ni idea tenemos que uno de estos realiza un trabajo para nosotros. En la medida que se demanda mayor actividad de ellos, también lo hacen ellos de las fuentes que proporcionan la energía para que operen. Estas actividades aumentan en complejidad y por ende incrementan los recursos internos necesarios para realizar dichas actividades. Estos recursos internos adicionales demandan energía que ahora debe considerarse para operar. Si es posible establecer cuánta energía demandan, será factible realizar adecuaciones para que consuman menores cantidades. Algunos fabricantes de microcontroladores se han dado a la tarea de evaluar sus productos y lo realizan con herramienta especializada, tal es el caso de [1] y [2]. El hardware y software utilizado es optimizado para aprovechar las características específicas del elemento en prueba. Es aquí donde la evaluación para el consumo de energía de los dispositivos se complica, ya que no es posible utilizar una herramienta para establecer el consumo de energía a través de las diferentes compañías.

Si es posible evaluar el consumo de energía independientemente del fabricante se tendrá una herramienta que permita determinar dónde se consume mayor energía en el código (incluyendo el hardware). En el presente se pretende generar una metodología que permita evaluar el consumo de energía independiente del compilador (parte de software) y del dispositivo (parte de hardware).

## 2. TRABAJO PREVIO

La energía en un dispositivo electrónico es indispensable para su correcta operación y funcionamiento. El consumo de energía es una preocupación creciente en la implementación de tareas en dispositivos electrónicos, ya sea para el denominado Internet de las Cosas con aplicaciones industriales (IIoT) [3] o para sistemas de monitoreo portátiles de variables médicas (ECG) [4]. Es de particular interés lo referente a sistemas médicos, principalmente los de monitoreo.

Los sistemas de monitoreo para signos vitales dependen tanto del hardware como software para operar en armonía. La necesidad de que estos sistemas consuman una menor cantidad de energía es de vital importancia. También lo es cumplir con las regulaciones de organismos oficiales para el diseño de dispositivos médicos. En la cuestión de hardware existen grandes avances para mejorar el consumo de energía utilizando técnicas de diseño e implementación. En cuestión de software no es tan claro ese rubro. El nuevo paradigma de la industria 4.0 para la comunicación a través del Internet de las cosas provee una oportunidad en la implementación de sistemas de monitoreo. Uno de ellos podría ser el médico.

En la actualidad existen sistemas de monitoreo de variables físicas médicas que ayudan al operador, sea paciente, técnico paramédico o médico, a tomar decisiones críticas en función de los parámetros mostrados. Dichos instrumentos pueden ser móviles o estacionarios. Tanto los móviles como estacionarios cuentan con baterías de respaldo para en caso de contingencia, continuar con su operación.

Aunque existen regulaciones y clasificaciones para instrumentos médicos [5] y software crítico [6] para ellos, no establecen un método para evaluar la capacidad de operación móvil o con baterías. En este departamento se observa un área de oportunidad ya que en el Estado de Baja California existen 26 empresas [5] dedicadas a la industria de manufactura o diseño de insumos o instrumentos médicos entre ellos los de monitoreo, sin contar los más de 2000 hospitales que se encuentran en el Estado [7]. La generación de investigación y tecnología para el bajo consumo de energía es de suma importancia en la creación de sistemas capaces de subsistir periodos prolongados con una fuente de energía finita. Cobra mayor importancia cuando se pretende aplicar ese conocimiento en instrumentos o equipos encargados de monitorear la salud de las personas. Además de las cuestiones de ahorro de energía, es importante cumplir con las especificaciones internacionales para estos instrumentos médicos. La manera de estructurar el software y generar la documentación apropiada sugiere una revisión en cómo el dispositivo interactúa con la metodología de programación implementada por el fabricante sin descuidar los nuevos estándares para software médico que permitirán reducir los problemas inherentes de programación.

La FDA (Food and Drug Administration de Estados Unidos) ha realizado recomendaciones para retirar del mercado bombas de infusión debido a los más de 56,000 reportes de incidentes desde 2005 a 2009 donde los principales son problemas de software [8]. También sugiere retirar defibriladores automáticos externos en función de los más de 45,000 reportes de fallas de estos dispositivos entre 2005 y 2012 donde los errores de software también se encuentran entre los de mayor incidencia [9]. Se han realizado investigaciones donde es de interés los problemas relacionados con software médico dando como resultado hasta un 98% de ellos prevenibles o detectables aplicando técnicas básicas de medición de errores [6]. Lo anterior despertó interés para crear un estándar específico para software médico, resultando en la IEC-62304:2006. Esta norma fue sugerida en 2006 [10] y aceptada en 2010. El Centro Nacional de Excelencia Tecnológica en Salud (CENETEC) establece en su documento para normas la IEC 62304 [10]. Dicha norma establece regulaciones en el desarrollo del software para dispositivos médicos. Estas medidas se acentúan con los problemas serios que pueden crear dispositivos médicos que no sean rigurosamente probados en software, tal es el caso de las muertes provocadas por el Therac-25 [11]. Mathworks ha presentado información respecto a este asunto, sugiriendo metodología para crear software y validarlo para equipo médico [12]. El estándar define a un dispositivo médico como cualquier instrumento, aparato, equipo, software, material u otro artículo, si se utiliza solo o en combinación con otros, incluyendo el software propuesto por el desarrollador para ser utilizado como diagnóstico y/o terapéutico. La inclusión de la palabra software es ahora de suma importancia, debido a que está condicionando una parte del proceso de desarrollo antes no contemplado [13]

Propone tres categorías de software médico basado en el posible daño que presenta al usuario o paciente:

Clase A.- Ningún daño o lesión a la salud es posible.

Clase B.- Daños o lesiones no catalogados como serios son posibles.

Clase C.- Muerte o lesiones catalogadas como serias son posibles.

El estándar define “SOUP” (Software de procedencia desconocida) a cualquier software que no tenga una documentación formal o que fue desarrollado por otros grupos y no cuente con evidencia del control utilizado para su desarrollo. Este software puede influir en la categoría para el grado médico, ya que no existe manera de analizar el posible daño o riesgo que presenta [5].

La adopción del estándar presentará ventajas a los usuarios, ya que tendrán la certeza de utilizar equipos médicos que cuenten con la seguridad necesaria no solamente en hardware, también en software. No solamente ayudará a los pacientes, también lo hará con el mejoramiento del proceso para el software de los desarrolladores [14].

El estándar sugiere prácticas apropiadas para la documentación del software, sin embargo, no presenta cómo impacta en el consumo de energía del dispositivo. Para ello, es necesario medir el consumo de energía una vez aplicadas las sugerencias del estándar.

### 3. DESARROLLO

Los desarrolladores de software actualmente están programando y codificando en la forma en que el hardware es utilizado en tiempo de ejecución, la tarea de habilitar la transparencia energética requerida puede ser vista como la transformación de código igual a energía requerida o consumida en hardware. Es decir un paradigma de programadores con conciencia energética. Para ello se deben proporcionar tanto el consumo de energía real como los límites (punto de referencia para posibles optimizaciones de energía), esto debe estar relacionado con el código, compilador y arquitectura de desarrollo. También es necesario tener en consideración el lenguaje de programación para permitir un desarrollo práctico y rentable en el lenguaje gramatical para tener la integración de código-energía. Por tanto se debe considerar casos de hilos, multi hilos y multi núcleos ya que en los últimos años se han desarrollado incrustaciones de múltiples núcleos por una alta demanda computacional lo cual debe ser considerado, ya que exploran códigos y procesamiento paralelo que ofrecen ahorro de energía potencialmente grandes cuando el número de núcleos aumenta y el voltaje y la frecuencia de los núcleos disminuyen.

La utilización de elementos inteligentes que controlen cuándo realizar tareas o no (microcontroladores principalmente) han permitido que dispositivos alimentados con fuentes de energía finita (baterías) extiendan su tiempo de vida. Es posible utilizar técnicas en estos dispositivos para el manejo de los diferentes módulos internos, así como los externos para hacer más eficiente el consumo de energía.

Así como existen métodos de monitoreo de consumo de energía de baterías para casa-habitación [1] que permiten el uso eficiente de la energía almacenada, es posible utilizar esas técnicas en microcontroladores. El microcontrolador puede gestionar el manejo de los módulos internos, así como los elementos externos para hacer más eficiente el consumo de energía. Por ejemplo en [2], un dispositivo que consume 5 mA al operar de manera normal, polarizada con una batería alcalina estándar de 2 A, podría estar en funcionamiento durante tres meses aproximadamente. El mismo dispositivo con técnicas apropiadas para la utilización de módulos de baja potencia, optimizando los recursos cuando está encendido y no, haciendo eficiente el algoritmo de procesamiento podría bajar el consumo de corriente a 500µA, extendiendo de esta manera la vida de la batería al doble, es decir, seis meses. Las técnicas a las que se hace referencia principalmente se dividen en dos: hardware y software [15]. Las relacionadas con hardware sugieren utilizar diferentes tecnologías para el semiconductor y las uniones entre los diversos dispositivos a implementar. Esta configuración no es modificable por el usuario final. Varios fabricantes de microcontroladores cuentan con esas características de ahorro de energía integradas [16] [17] (Energymicro, Renesas, Texas Instruments, Microchip, entre otros). La

optimización debido a la arquitectura del microcontrolador se logrará estructurando el conjunto de comandos del dispositivo para una ejecución armónica entre una instrucción y otra. Algunas de estas compañías proveen herramientas que calculan el consumo de energía de cada instrucción en un bloque de código. Con este resultado es posible estimar el perfil de energía [18]. Las categorías de manejo de la energía por medio del software son divididas en análisis de técnicas de manejo a nivel de código fuente, a nivel de algoritmo y arquitectura del software, y en esta categoría es donde el usuario final tiene mayor control del consumo de energía [15].

## Perfil de energía

La definición de bajo consumo de energía varía de una aplicación a otra. Para este experimento se considera el reducir al máximo la energía consumida por el microcontrolador. En algunos sistemas, existen fuentes de alimentación que no tendrán problema en suministrar la energía necesaria para la aplicación, sin embargo se puede optimizar el consumo para reducir costo o maximizar eficiencia. Cuando la fuente de energía es limitada (baterías) o el sistema energizado se desconecta y debe operar con la fuente secundaria (normalmente baterías), la operación del sistema dependerá de la capacidad de la fuente y en esta situación los sistemas requieren un análisis del perfil de energía. En los dispositivos electrónicos que utilizan la tecnología CMOS, como los microcontroladores y microprocesadores, la energía total que consume puede dividirse en dos rubros: la energía dinámica y estática. La energía dinámica es la que consume el dispositivo cuando está ejecutando alguna rutina del proceso. La energía estática es la que consume el dispositivo simplemente al estar en reposo y energizado.

La Energía Dinámica incluye la pérdida de energía cuando los circuitos CMOS conmutan y las corrientes que polarizan los circuitos analógicos (osciladores, ADC, DAC, entre otros). Dicha Energía Dinámica se calcula para cada uno de los circuitos de la siguiente manera como se muestra en la ecuación 1:

$$E_{\text{Dinámica}} = V^2 \cdot f \cdot C \quad \text{Ecuación (1)}$$

donde  $V$  es el voltaje de operación (VDD),  $f$  es la frecuencia de conmutación y  $C$  es la capacitancia de carga de cada uno de los elementos CMOS [21] [22]. La frecuencia de conmutación es particular para cada diseño del fabricante, por lo que no se tiene control sobre este parámetro. La capacitancia de carga  $C$  es complicada medirla y por ende controlarla. El único parámetro accesible para el usuario es el voltaje de polarización VDD. El reducir el voltaje de polarización es una forma de disminuir la energía consumida por el microcontrolador. Otras técnicas específicas para el hardware se sugieren para las terminales no utilizadas del microcontrolador. Dichas terminales pueden ser configuradas como entradas o salidas. Si optan por entradas, deben estar conectadas con resistencias de jalón hacia VDD o VSS. Si la alternativa es configurarlas como salidas es posible dejarlas “flotando” pero deben generar un estado lógico para que el microcontrolador procure siempre tener un valor de voltaje estable en dicha terminal y no active o desactive la región lineal del circuito CMOS interno [19] [20] [23]. Por último se cuenta con las configuraciones de bajo consumo de energía específicos de los fabricantes. Aunque estas opciones pueden activarse en la etapa de pre procesamiento y/o en la ejecución del algoritmo y podrían considerarse como parte del software, se consideran parte del hardware debido a que son específicamente diseñadas en los microcontroladores para esas funciones [18] [19] [20] [23]. Dichas configuraciones permiten la desactivación o activación de sistemas internos y podrían optimizar el consumo dependiendo la aplicación.

Tiwari [24] presenta en sus investigaciones el impacto del código generado en relación a la energía consumida por el sistema. Experimentó con diferentes instrucciones y técnicas de programación para determinar cuánta energía consumía el programa. Obtuvo resultados favorables al reducir el consumo de energía en un 40% y plantear nuevas técnicas de compilación [25].

Además del algoritmo, varios estudios han demostrado que la manera de programar impacta directamente en el consumo de energía del dispositivo, otros estudios detectaron que el nivel de optimización utilizado para generar el código máquina tiene repercusiones en el uso eficiente de la energía [26] [27] [28].

Los niveles de optimización del compilador pueden reducir de manera importante el tamaño de código en lenguaje de alto nivel a transformar en código máquina y por ende, implementarlo en dispositivos de menores recursos. Sin embargo, este resultado produce código diseñado para una ejecución lo más rápida posible pero aumentando el consumo de energía [29]. El no utilizar optimizaciones para la generación de código máquina en los diferentes programas de compilación podría ser una manera para no incrementar el consumo de energía, pero el resultado puede ser hasta un 250% más instrucciones [29]. Se considera que esto no necesariamente garantiza un consumo de energía menor, por lo que se proponen herramientas de medición del consumo independiente de compiladores y estimaciones de fabricantes.

En la medida que la frecuencia aumenta, también lo hace el consumo de energía, pero contrario a lo que se podría pensar si se utiliza una frecuencia baja, esta provoca un consumo mayor de energía en algunos casos [15] [28]. El tiempo que se invierte en algoritmos complejos sería mayor, por lo que la energía acumulada tendería a elevarse. La tendencia es utilizar algoritmos eficientes durante periodos cortos al máximo de la capacidad que proporciona el dispositivo y regresar al estado de mínimo consumo de energía [15] [28].

Investigaciones han demostrado que además de las optimizaciones a la compilación de software, se debe cuidar el número de variables, la extensión de las mismas y las ocasiones en las que se accede a ellas. Además, la estructura del dispositivo y el acceso a la memoria aportan significativamente al consumo de energía [25] [26] [27] [28].

## Experimentación

Para obtener la energía, es necesario medir la potencia que consume el dispositivo bajo prueba durante el periodo de interés. Al obtener dicha información, es necesario integrarla. En la mayoría de las ocasiones para obtener la energía (figura 1), se mide la corriente promedio del dispositivo en conjunto con el voltaje para obtener la potencia y de esta manera tener acceso a la energía promedio (integrando la potencia).

En la suma de productos del tiempo en el que se encuentra ejecutando alguna tarea específica y en modo de bajo consumo [23], la energía consumida por el dispositivo es la integral de la potencia instantánea en un periodo de la señal. Si la corriente se mide de manera instantánea y el voltaje también (figura 3), la potencia será instantánea y la energía también. En este trabajo se realiza una medición de la corriente instantánea, por lo que la energía será también instantánea.

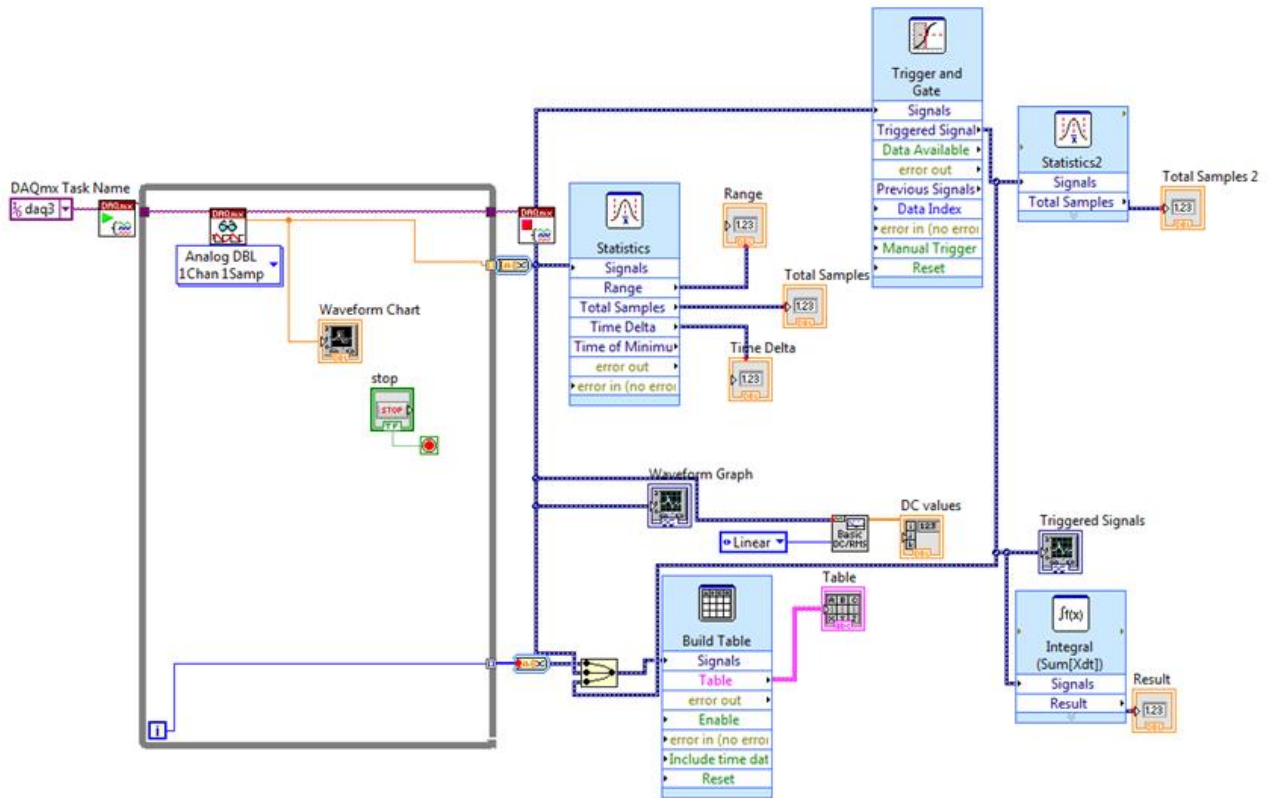


Figura 1: Sistema de adquisición de datos, etapa de pre-procesamiento, detección de periodos y obtención de la energía.

Al ser instantánea la medición permite generar un perfil de energía en armonía con los ciclos de reloj de ejecución del microcontrolador, proporcionando información sobre cuáles son las áreas de oportunidad para una mejor estructuración en software para el consumo de energía, figura 2,3.

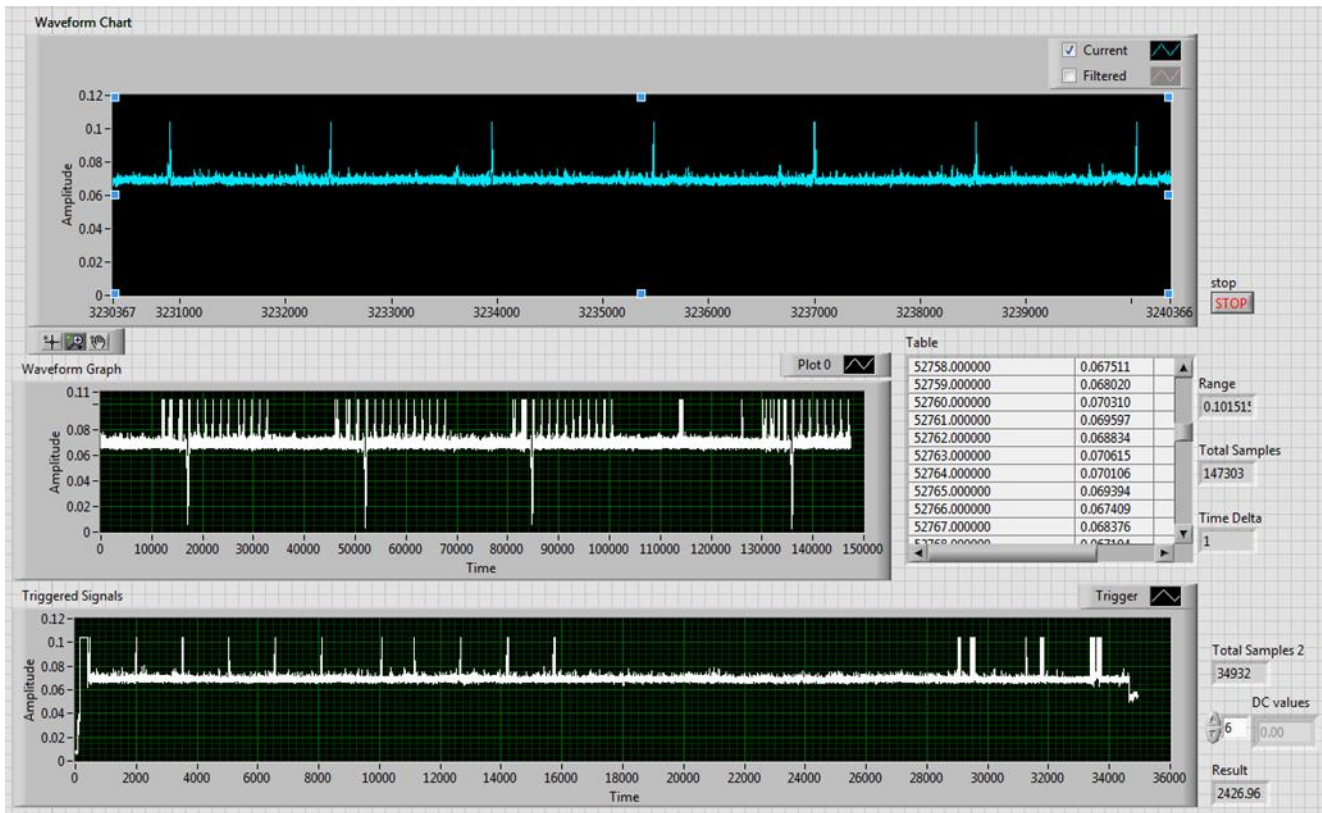
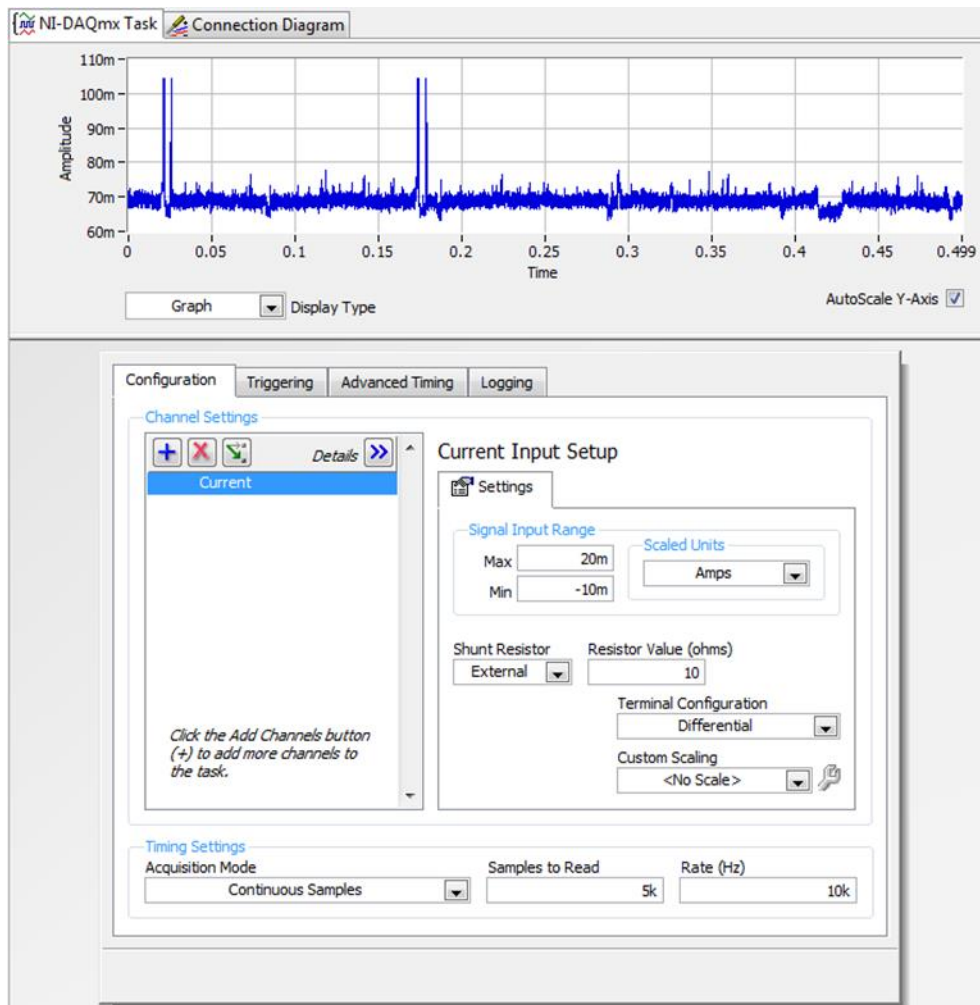


Figura 2: Perfil de energía en experimento con comportamiento independiente del transitorio.





**Figura 3:** Configuración de la adquisición de corriente en el experimento.

La bobina de Rogowski es utilizada para medir la corriente alterna que consume un dispositivo bajo prueba. Es una construcción de una bobina en forma circular que tiene sus extremos conectados a un integrador. Tiene la ventaja de no ser invasiva al lazo de corriente que debe establecerse con el dispositivo bajo prueba (como se mostrará con las otras técnicas). La medición de corriente y posible implementación de medición de energía se ven limitadas al ser solamente de corriente alterna y procurar corrientes significativamente mayores en comparación con los dispositivos de bajo consumo de energía (un orden de magnitud mayor) [29]. El espejo de corriente es una técnica que permite igualar la corriente que consume una rama en un circuito con otra. Una de las ventajas de esta técnica es la impedancia de salida elevada que permite mantener la corriente independientemente de la carga [30]. Ejemplos de aplicación de espejo de corriente se observan en [31] [32] [33] y [34]. Utilizan un espejo de corriente Wilson completo [30] para duplicar la corriente del dispositivo bajo prueba a medir. El agregar cuatro transistores para realizar el espejo completo de Wilson complica la medición al realizar un instrumento de mayor complejidad. No solo eso, la medición de corriente utilizando shunt depende de la precisión de la carga utilizada para medir. Otra técnica de medición de energía implica medir la carga almacenada en un capacitor y por ende, la corriente. Cuando un dispositivo opera en modo de bajo consumo de energía (microcontrolador), se comporta como una fuente constante de corriente. Esta técnica emplea la ecuación 2 de corriente en un capacitor para medir la descarga del capacitor [20] utilizando un voltaje

definido durante un periodo establecido. Esto permite realizar una conversión para la corriente consumida por el microcontrolador.

$$I=C dv/dt \quad \text{Ecuación (2)}$$

Donde  $I$  es la corriente que consume el dispositivo en amperes,  $V$  es el voltaje generado en volts, tiempo  $t$  en segundos y  $C$  es la carga almacenada en el capacitor en Faradios. Esta técnica también es utilizada en [32] y [34], donde ahora se procura medir el tiempo que tarda la descarga para calcular directamente la carga del capacitor. Al utilizar espejos de corriente, la variable de interés ahora es la carga.

Esta técnica es muy precisa, ya que permite definir la carga que está utilizando el microcontrolador. Sin embargo, es necesario garantizar una correcta medición de la corriente, el voltaje y el tiempo, además de tener un capacitor con una corriente de fuga pequeña. Es posible también medir la corriente utilizando el fenómeno conocido como efecto Hall. Dicho comportamiento se presenta cuando a través de un conductor se presenta una corriente, esta genera un campo magnético perpendicular al sentido de la misma y produce una caída de tensión proporcional en magnitud y sentido a la corriente [35]. Esta técnica de medición puede ser utilizada para medir corrientes pequeñas como se sugiere en [36] donde es posible medir microamperes, sin embargo, el bajo consumo de energía se encuentra un orden de magnitud menor (en ocasiones dos) que eso. Existen fabricantes de dispositivos que integran esta tecnología como Allegro [39] y Melexis [38], aunque para la medición de energía en cuestión, no son adecuados debido a su pobre resolución (medición en décimas de Ampere).

La técnica más utilizada por ser la más directa y sencilla es la conocida como Shunt. Consta de colocar una resistencia en el paso de la corriente a medir y realizar una medición de voltaje en ella. Es básicamente un convertidor de voltaje a corriente. Esta medición es proporcional respecto a la ecuación de corriente de Ohm, como se muestra en la ecuación 3:

$$I=V/R \quad \text{Ecuación (3)}$$

Donde  $I$  es la corriente en Amperes presente en la resistencia  $R$  en Ohms debido a la diferencia de potencial  $V$  en Volts. Cabe señalar que debe medirse voltaje diferencial y dicho voltaje puede ser medido por amplificadores operacionales.

Las principales ventajas en utilizar esta técnica es su bajo costo (al utilizar una resistencia), alta precisión y la posibilidad de medir corrientes pequeñas (micro, nano e inclusive pico Amperes). Las principales desventajas de esta técnica es la de agregar resistencia al lazo de polarización, debido a que es una medición directa y la disipación de energía debido a la potencia generada en la resistencia con la que probablemente se modifique el comportamiento del dispositivo si la caída de tensión es significativa en el shunt [39]. Aunque la medición de energía a través de resistencias es directa, existen dos técnicas a utilizar con shunt, la medición de lado alto y la medición de lado bajo.

La medición de lado alto para un shunt obedece a la característica de la resistencia de estar entre la fuente de alimentación y el dispositivo de medición. Una de las ventajas de esta medición es que el dispositivo bajo prueba está conectado directamente al punto en común (tierra) sin embargo, al realizar una medición donde directamente se mide la polarización del dispositivo, es necesario que el dispositivo utilizado para medir dicho voltaje tenga la capacidad de medir los probables valores elevados de polarización. Una muestra de esta implementación se observa en [40] donde se realizan modelados para la medición del consumo de energía por

instrucciones. La medición de lado bajo sugiere una resistencia entre el sistema bajo prueba y tierra o punto en común. Una de las ventajas de esta medición es que la referencia tanto del instrumento de medición como del shunt están conectados directamente al punto en común (tierra) sin embargo, el voltaje diferencial medido en el dispositivo bajo prueba se encuentra referenciado respecto al punto más positivo del shunt, y no al punto común (está flotado).

Una representación de esta medición se observa en [41], donde se mide la energía consumida por aplicaciones biomédicas. Tektronix cuenta con una punta de prueba (TCP312) para osciloscopio capaz de realizar mediciones del orden de miliamperes [42]. Utiliza el principio de la bobina de Rogowski y efecto Hall. Su elevado costo y el solo poder acceder a miliamperes lo hace impráctico para la medición de bajo consumo de energía (uno o dos órdenes de magnitud menor son necesarios). EEMBC cuenta con un sistema de inyección de carga para el monitoreo del consumo de energía denominado EnergyBench [43]. Plantean una modulación de energía que se entrega a un sistema bajo prueba.

Existen requerimientos para su utilización, por ejemplo, debe tener un módulo RTC (un reloj y calendario con su propio oscilador) el dispositivo bajo prueba; además solo puede aplicarse a dispositivos de 3.3 volts.

#### 4. RESULTADOS

La utilización de sistemas e instrumentos de adquisición de datos para analizar y tener el control de escenarios eléctricos y del tiempo (periodos de pre-procesamiento en tiempo real) fue clave para el análisis de esta investigación (figura 4).

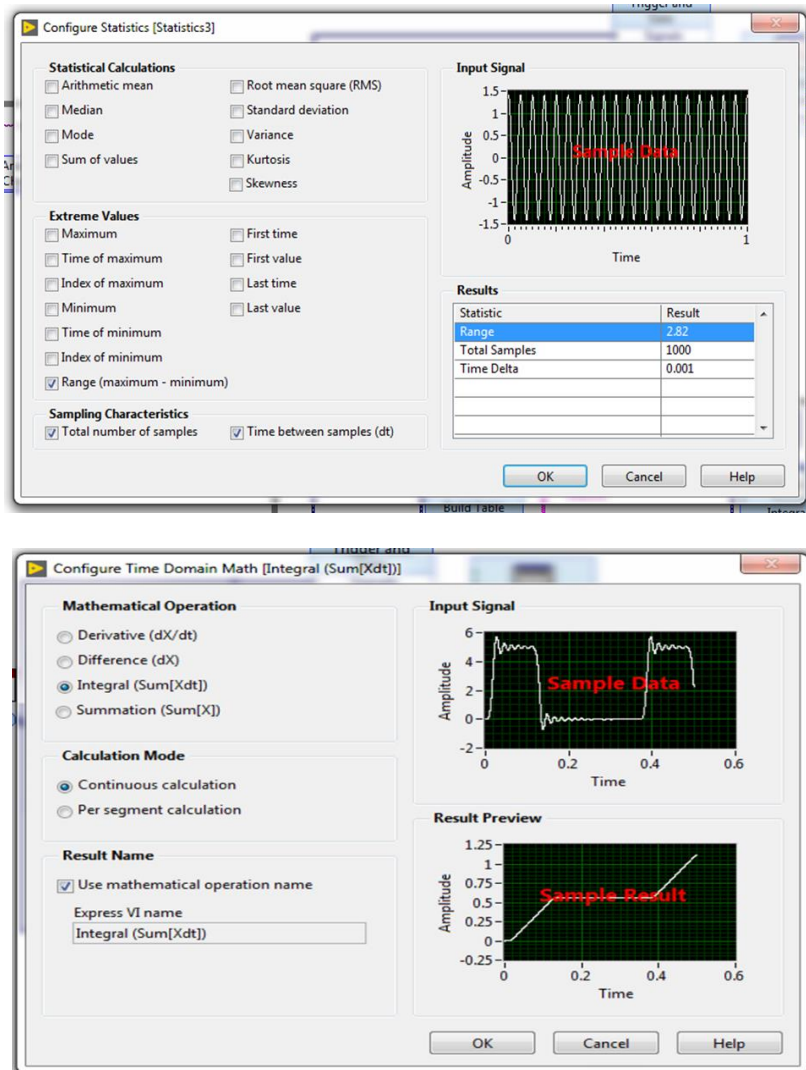
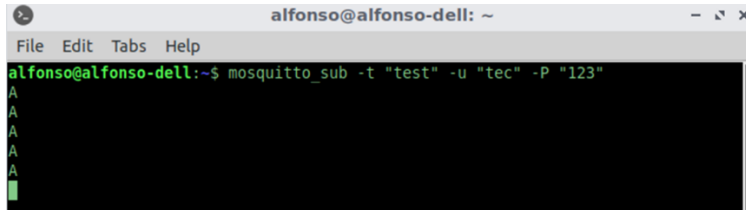


Figura 4: Configuración de energía consumida



```

//esp wifi
#include <ESP8266WiFi.h>

//libreria MQTT
#include <PubSubClient.h>

//libreria HTTP
#include <ESP8266HTTPClient.h>

//configurar conexion
const char* ssid = "CAII";
const char* password = "posgradol6IIM";
//configuracion de host
const char* mqttServer = "192.168.1.16";
const char* httpServer = "http://192.168.1.16/test.html";

//usuario y password MQTT
const int mqttPort = 1883;
const char* mqttUser = "tec";
const char* mqttPassword = "123";

// inicializacion de cliente MQTT
WiFiClient espClient;
PubSubClient client(espClient);

//inicializacion de cliente HTTP
HTTPClient http;

//variable para conteo de iteraciones
int pin_reset=16;

void setup() {
    Serial.begin(115200);// monitoreo

    //iniciar conexion wifi
    WiFi.begin(ssid, password);
    while(WiFi.status() != WL_CONNECTED) ;

    Serial.begin(115200);// monitoreo
    //Serial.println("Conexion exitosa");// monitoreo

    //configuracion de protocolos
    config_mqtt();
    //config_http();

    //envio mediante protocolo////////////////////////////////////

    envio_mqtt();
    //envio_http();

    //*****wdt*****
    //wdt_enable(WDTO_8S);
    delay(1);
    ESP.restart();
}

void config_mqtt(){
    client.setServer(mqttServer, mqttPort);
    client.connect("ESP8266Client", mqttUser, mqttPassword );
}

void config_http(){
    http.begin(httpServer);
}

void envio_mqtt(){
    client.publish("test", "A");
}

void envio_http(){
    http.POST("A");
    String payload = http.getString();// monitoreo
    Serial.println(payload);// monitoreo
}

```

Figura 5: Algoritmo analizado en este proyecto

El programa de prueba comunica con un servidor el cual envía un carácter (a) y presenta la información una vez que el sistema recibe la información, este envío y recepción es la medición de energía que se analizó.

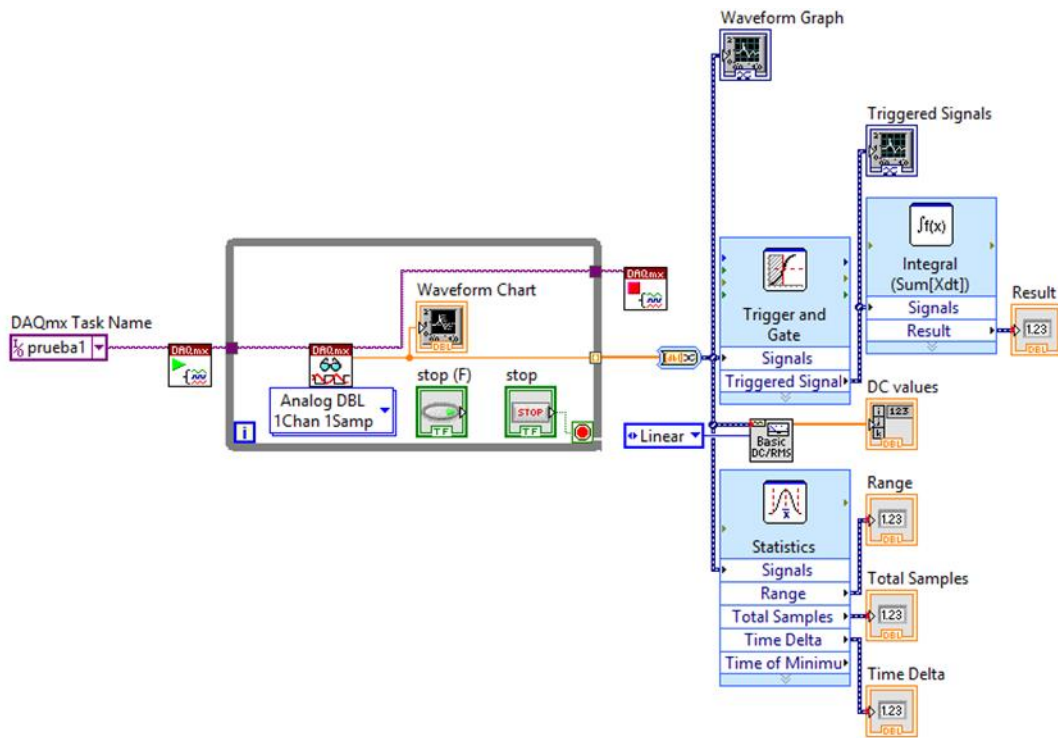


Figura 6: Esquema general del experimento de medición de energía. Perfil final de energía.

## 5. CONCLUSIONES

Mediante el uso de herramientas como Labview es posible desarrollar software que programe y codifique en tiempo real, tiempos de ejecución de la energía; la tarea de habilitar la transparencia energética requerida puede ser vista como la transformación de código igual a energía requerida o consumida en hardware.

Es muy importante proporcionar tanto el consumo de energía real como los límites, es decir el punto de referencia para posibles optimizaciones de energía, relacionado el código, compilador y arquitectura de desarrollo. También es importante el lenguaje de programación para permitir evaluar el consumo de energía y determinar dónde se consume mayor energía en el código (incluyendo el hardware). La metodología que se utilizó en esta investigación permite evaluar el consumo de energía independiente del compilador y del dispositivo (hardware). Minimizar el consumo de energía en dispositivos con recursos limitados es necesario para el empleo del internet de las cosas, sin embargo hace falta hacer más experimentación en esta investigación para determinar un perfil de energía que sea capaz de definir en qué fase se consume más energía al enviar o recibir información, tanto en hardware como software.

El porcentaje de energía consumida por los dispositivos conectados aumenta debido al crecimiento significativo en el número de dispositivos del internet de las cosas, es por ello que se necesita hacer más investigación.

6. - REFERENCIAS BIBLIOGRÁFICAS

- [1] B. Belvedere, M. Bianchi, A. Borghetti, C. Nucci, M. Paolone and A. Peretto, "A Microcontroller-Based Power Management System for Standalone Microgrids With Hybrid Power Supply," *Sustainable Energy*, IEEE Transactions on, vol. 3, no. 3, pp. 422-431, 2012.
- [2] Digi-Key, "http://www.designnews.com," 14 Enero 2013. [Online]. Available: [http://www.designnews.com/lecture-calendar.asp?p\\_ed=CEC\\_Semester\\_Three\\_2013#lecture\\_track\\_cgid\\_168](http://www.designnews.com/lecture-calendar.asp?p_ed=CEC_Semester_Three_2013#lecture_track_cgid_168). [Accessed 31 Diciembre 2018].
- [3] K. Dhondge, Shorey and R. Tew, "HOLA: Heuristic and opportunistic link selection algorithm for energy efficiency in Industrial Internet of Things (IIoT) systems," 2016 8th International Conference on Communication Systems and Networks (COMSNETS), pp. 1-6, 2016.
- [4] S. Sridhara, "Ultra-low power microcontrollers for portable, wearable, and implantable medical electronics," 16th Asia and South Pacific Design Automation Conference, pp. 556-560, 2011.
- [5] Buntz, Brian, "http://www.emdt.co.uk," 1 Junio 2010. [Online]. Available: <http://www.emdt.co.uk/article/developing-medical-device-software-iso-62304>. [Accessed 1 Enero 2019].
- [6] K. Sandler, L. Ohrstrom, L. Moy and R. McVay, "Software Freedom Law Center," 21 Julio 2010. [Online]. Available: <http://www.softwarefreedom.org/resources/2010/transparent-medical-devices.html>. [Accessed 1 Enero 2019].
- [7] "http://www.industriamedica.org," [Online]. Available: <http://www.industriamedica.org/directorio.php>. [Accessed 2 Enero 2019].
- [8] "http://www.seccionamarilla.com.mx," [Online]. Available: <http://www.seccionamarilla.com.mx/resultados/hospitales/baja-california/1>. [Accessed 2 Enero 2019].
- [9] FDA, "Food and Drug Administration," 23 Abril 2010. [Online]. Available: <http://www.fda.gov/medicaldevices/productsandmedicalprocedures/generalhospitaldevicesandsupplies/infusionpumps/ucm206000.htm>. [Accessed 2 Enero 2019].
- [10] E. Jefferson, "http://www.fda.gov," Food and Drug Administration, 22 03 2013. [Online]. Available: <http://www.fda.gov/NewsEvents/Newsroom/PressAnnouncements/ucm345062.htm>. [Accessed 2 Enero 2019].
- [11] P. Jordan, "Standard IEC 62304 - Medical Device Software - Software Lifecycle Processes," *Software for Medical Devices*, The Institution of Engineering and Technology Seminar on, pp. 41-47, 2006.
- [12] J. Hoyo, "http://www.cenetec.salud.gob.mx," 28 Junio 2007. [Online]. Available: [http://www.cenetec.salud.gob.mx/descargas/normas/Normas\\_ISO.pdf](http://www.cenetec.salud.gob.mx/descargas/normas/Normas_ISO.pdf). [Accessed 2 Enero 2019].
- [13] N. Leveson and C. Turner, "An investigation of the Therac-25 accidents," *Computer*, vol. 26, no. 7, pp. 18-41, 1993.
- [14] Mathworks, "Mathworks," 22 Octubre 2013. [Online]. Available: [http://www.mathworks.com/videos/developing-iec-62304-compliant-medical-device-software-using-model-based-design-86469.html?form\\_seq=conf1386&confirmation\\_page&wfsid=5070835](http://www.mathworks.com/videos/developing-iec-62304-compliant-medical-device-software-using-model-based-design-86469.html?form_seq=conf1386&confirmation_page&wfsid=5070835). [Accessed 3 Enero 2019].
- [15] M. McHugh, F. McCaffery and V. Casey, "Software process improvement to assist medical device software development organisations to comply with the amendments to the medical device directive," *Software*, IET, vol. 6, no. 5, pp. 431-437, 2012.
- [16] P. Jordan, "Medical device software standards," *Medical Device Standards & Regulations*, IEE Seminar on , pp. 6/1 - 6/6, 2005.
- [17] G. Luo, B. Guo, Y. Shen, H. Liao and L. Ren, "Analysis and Optimization of Embedded Software Energy Consumption on the Source Code and Algorithm Level," *Embedded and Multimedia Computing*, pp. 1-5, 2009.
- [18] Microchip, "MPLAB REAL ICE Power Monitor," Microchip, 4 12 2013. [Online]. Available: <http://www.microchip.com/Developmenttools/ProductDetails.aspx?PartNO=AC244008>. [Accessed 5 Enero 2019].
- [19] Texas, "MSP EnergyTrace Technology," TI, 2014. [Online]. Available: <http://www.ti.com/tool/energytrace>. [Accessed 5 Enero 2019].
- [20] Renesas Electronics, "http://www.techonline.com," 18 Septiembre 2013. [Online]. Available: <http://www.techonline.com/electrical-engineers/education-training/webinars/4420344/Implementing-Ultra-low-Power-Design-Techniques-for-High-performance-32-bit-MCU-based-Applications>. [Accessed 1 Enero 2019].
- [21] A. Holberg, "ATMEL," 02 2006. [Online]. Available: [www.atmel.com](http://www.atmel.com). [Accessed 1 Enero 2019].
- [22] B. Ivey, "Microchip," 2011. [Online]. Available: [www.microchip.com](http://www.microchip.com). [Accessed 1 Enero 2019].
- [23] Tsekoura, Rebel, Glösekötter and Berekovic, "An evaluation of energy efficient microcontrollers," *Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC)*, pp. 1-5, 2014.
- [24] R. Richey, *Low Power Design Using PICmicro Microcontrollers AN606*, 1997.
- [25] "ST," Mayo 2013. [Online]. Available: [www.st.com](http://www.st.com). [Accessed 1 Enero 2019].
- [26] V. Tiwari, S. Malik and A. Wolfe, "Power Analysis Of Embedded Software: A First Step Towards Software Power Minimization," *Computer-Aided Design*, IEEE/ACM International Conference on, pp. 384-390, 1994.
- [27] V. Tiwari, S. Malik and A. Wolfe, "Compilation Techniques for Low Energy: An Overview," *Low Power Electronics, Digest of Technical Papers., IEEE Symposium*, pp. 38-39, 1994.
- [28] C. Robertson and C. Martinez, "Analyzing the software aspect of an embedded system's power consumption," *Electrical and Computer Engineering (CCECE)*, 24th Canadian Conference on, pp. 853-856, 2011.
- [29] D. Ortiz and N. Santiago, "Impact of Source Code Optimizations on Power Consumption of Embedded Systems," *Circuits and Systems and TAISA Conference*, pp. 133-136, 2008.

- [30] M. Ibrahim, M. Rupp and S. E. -D. Habib, "Compiler-Based Optimizations Impact on Embedded Software Power Consumption," Circuits and Systems and TAISA Conference, pp. 1-4, 2009.
- [31] J. Liao, X. Guo, C. Luo, M. Zhu and Z. Yang, "Studies of Rogowski coil current transducer for low amplitude current (100A) measurement," Electrical and Computer Engineering, 2003. IEEE CCECE 2003, vol. I, pp. 463-466, 2003.
- [32] dmercer, "Analog Devices University," Analog Devices, 09 Enero 2014. [Online]. Available: <https://wiki.analog.com/university/courses/electronics/text/chapter-11>. [Accessed 1 Enero 2019].
- [33] Borovyi, Kochan, Sachenko, Konstantakos and Yaskilka, "Analysis of Circuits for Measurement of Energy of Processing Units," Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, pp. 42-46, 2007.
- [34] Konstantakos, Chatzigeorgiou, Nikolaidis and Laopoulos, "Energy Consumption Estimation in Embedded Systems," IEEE Transactions on Instrumentation and Measurement, vol. 57, no. 4, pp. 797-804, 2008.
- [35] Konstantakos and Laopoulos, "A Power Measuring Technique for Built-in Test Purposes," IEEE Instrumentation and Measurement Technology Conference Proceedings, pp. 90-95, 2006.
- [36] Konstantakos, Kosmatopoulos, Nikolaidis and Laopoulos, "Measurement of Power Consumption in Digital Systems," IEEE Transactions on Instrumentation and Measurement, vol. 55, no. 5, pp. 1662-1670, 2006.
- [37] E. Ramsden, Hall-Effect Sensors Theory and Applications, Newnes, 2006.
- [38] Cirstea, Cernaianu and Gontean, "An inductive system for measuring microampere currents," Design and Technology in Electronic Packaging (SIITME), pp. 197-200, 2012.
- [39] Allegromicro, "Allegro MicroSystems," 2004. [Online]. Available: <http://www.allegromicro.com/en/Products/Current-Sensor-ICs.aspx>. [Accessed 5 Enero 2019].
- [40] Melexis, "Melexis," 2012. [Online]. Available: <https://www.melexis.com/en/products/sense/current-sensors>. [Accessed 5 Enero 2019].
- [41] Zhen, "Current Sensing Circuit Concepts and Fundamentals AN1332," 2011. [Online]. Available: [ww1.microchip.com/downloads/en/AppNotes/01332B.pdf](http://ww1.microchip.com/downloads/en/AppNotes/01332B.pdf). [Accessed 5 Enero 2019].
- [42] M. Bazzaz, S. M and A. Ejlali, "An Accurate Instruction-Level Energy Estimation Model and Tool for Embedded Systems," IEEE Transactions on Instrumentation and Measurement, vol. 62, no. 7, pp. 1927-1934, 2013.
- [43] Kim, "Performance comparison of low current measurement systems for biomedical applications," Proceedings of 2010 IEEE International Symposium on Circuits and Systems, pp. 3469-3472, 2010.
- [44] Tektronix, "TEquipment," Tektronix, 2009. [Online]. Available: [http://www.tequipment.net/TektronixTCP312.html?gclid=Cj0KEQjwztG8BRCJgseTvZLctr8BEiQAA\\_kBD7Zf1VqE-Ms0oFcwfs5eEoKzdRe\\_F\\_BZVaW0T8agDbwaAiKM8P8HAQ#tab-reviews](http://www.tequipment.net/TektronixTCP312.html?gclid=Cj0KEQjwztG8BRCJgseTvZLctr8BEiQAA_kBD7Zf1VqE-Ms0oFcwfs5eEoKzdRe_F_BZVaW0T8agDbwaAiKM8P8HAQ#tab-reviews). [Accessed 5 Enero 2019].
- [45] EEMBC, "EEMBC," EEMBC, 13 Abril 2014. [Online]. Available: [http://www.eembc.org/benchmark/power\\_sl.php](http://www.eembc.org/benchmark/power_sl.php). [Accessed 5 Enero 2019].
- [46] CMicrotek, "CMicrotek," CMicrotek, 2010. [Online]. Available: <http://www.cmicrotek.com/uCP.htm>. [Accessed 5 Enero 2019].