

# Aplicación de los métodos de Codiseño Hardware Software para la enseñanza y capacitación en el uso racional de la energía

Mazzeo, Hugo; Rapallini, José A.; Rodríguez, Omar E.; Zabaljauregui, Marcelo

*CODAPLI - Codiseño Hardware/Software para aplicaciones de tiempo real*  
*Universidad Tecnológica Nacional - Facultad Regional La Plata*  
[hmqvm@yahoo.com](mailto:hmqvm@yahoo.com), [josrap@gmail.com.ar](mailto:josrap@gmail.com.ar)

## RESUMEN

El objetivo de este artículo es presentar una metodología de trabajo utilizada en la cátedra Aplicaciones de Tiempo Real con alumnos de la carrera de Ingeniería en Sistemas de Información para despertar la conciencia en el uso racional de la energía. Para ello se exponen, en forma introductoria, las características y temática de la materia para luego mostrar algunos de los resultados que esta propuesta ha obtenido. El desarrollo de pequeños proyectos ha logrado incorporar la discusión de gasto energético y maximizar su aprovechamiento en el centro de cada clase, analizando soluciones y la manera de resolverlas prácticamente con las herramientas que los alumnos han aprendido a utilizar.

Además, hemos observado que el interés de los alumnos se mantiene luego de cursada la materia, lo que ha generado un incremento en la realización de prácticas supervisadas y proyectos de graduación en la temática. Esto nos alienta a avanzar y profundizar los temas dictados, acercando especialistas de otros departamentos y grupos de investigación de la Facultad a nuestros estudiantes.

Estos proyectos se enmarcan en un método didáctico unificado para modelar, generar código e implementar sistemas de tiempo real utilizando técnicas de codiseño hardware/software. Se pretende mostrar un método de trabajo orientado fundamentalmente a quienes poseen conocimientos importantes de programación, pero escasos o nulos de electrónica, por ser esta una carrera del área de la Informática. La existencia en el mercado de placas microcontroladas accesibles permite desarrollar sistemas embebidos sin necesidad de diseñar hardware costoso y complejo, generando soluciones simples y económicas para el ahorro de energía en distintos ámbitos.

**Palabras Clave:** Aplicaciones, Energía, Codiseño, Microcontroladores, Embebidos.

## 1. INTRODUCCIÓN

Las técnicas de codiseño hardware/software son motivo de estudio e investigación en el ámbito académico desde hace varios años, tratando de lograr metodologías para utilizar en el diseño de sistemas complejos mediante ambientes de trabajo flexibles que permitan cumplir fácilmente con las etapas que corresponden para la solución a estos problemas.

Para ello existen diferentes herramientas EDA (Electronic Design Automation) universitarias y comerciales y otras para la descripción de hardware y software, especificación y verificación de sistemas de tiempo real y sistemas embebidos [1].

La integración de estas herramientas genera la propuesta didáctica para los distintos proyectos a los que se apunta desde el plano docente, considerando entonces la especificación del sistema mediante una Red de Petri y su posterior representación en software mediante lenguajes de tiempo real como ADA [2]. Las posibilidades que ofrecen estos lenguajes para la confección de programas multitarea, manejo de tiempos y concurrencia, hacen que sean los más adecuados para la representación de sistemas de tiempo real. A partir de este punto, la etapa siguiente es la traslación del código obtenido a lenguaje VHDL para la implementación del sistema de control en un dispositivo lógico programable adecuado o con dispositivos embebidos basados en placas microcontroladas con entorno de desarrollo [3,4].

## 2. CASOS DE ESTUDIO

Para ilustrar el procedimiento se presentan tres casos testigo de proyectos realizados por alumnos del último año de la carrera de Ingeniería en Sistemas de Información, desarrollados como parte de la materia Aplicaciones de Tiempo Real o como prácticas supervisadas dirigidas por investigadores del Laboratorio CODAPLI (Codiseño Aplicado) [5], [6], desde donde se alienta a los alumnos de la carrera a iniciarse en tareas de investigación.

### 2.1. Control de un Panel Solar para el máximo aprovechamiento de la Radiación Luminosa incidente.

Se presenta el desarrollo e implementación de un dispositivo que permite realizar el seguimiento de la trayectoria solar, aumentando la radiación captada por el módulo solar mediante el movimiento continuo del panel. Tal movimiento busca mantener su área en la dirección de mayor radiación solar mejorando el rendimiento de una instalación fotovoltaica.

La incidencia de los rayos solares sobre el plano de los paneles fotovoltaicos difiere de la perpendicularidad a lo largo del día y por lo tanto se ve reducida su eficiencia. Es posible corregir esta situación utilizando un seguidor solar.

Este dispositivo sigue la trayectoria aparente del sol a través del cielo durante el día (Figura 1), exponiendo sus celdas fotovoltaicas a un incremento en la cantidad de radiación recibida y produciendo más electricidad [7,8]. Esto se debe a que las celdas fotovoltaicas funcionan mejor cuando están de cara al Sol [9,10].

El uso de seguidores solares ha reportado ganancias superiores al 20% en estudios de campo [11].

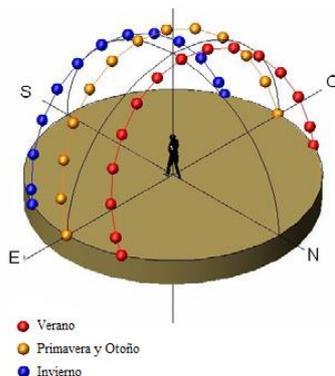


Figura 1. Variación en la trayectoria aparente del sol con las estaciones

Existen dos tipos principales de seguidores solares según el tipo de movimiento que presentan, los de un solo eje y los de dos ejes. En este caso se adoptó la segunda opción. Para lograrlo se requiere medir la radiación luminosa a través de cuatro sensores y controlar el movimiento del panel a través de 2 servomotores, uno para cada eje (Figura 2).

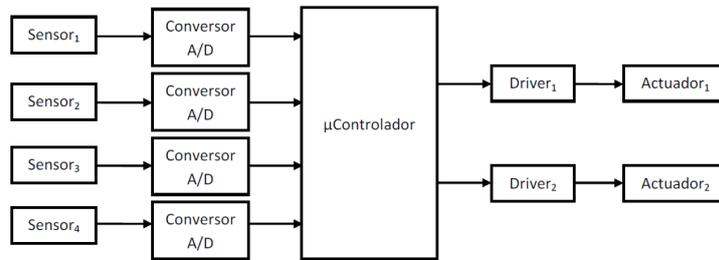


Figura 2. Diagrama en bloques del control del panel solar.

Se comenzó realizando la especificación del sistema mediante una Red de Petri y el diagrama funcional del sistema de control del panel solar, como puede verse en la Figura 3.

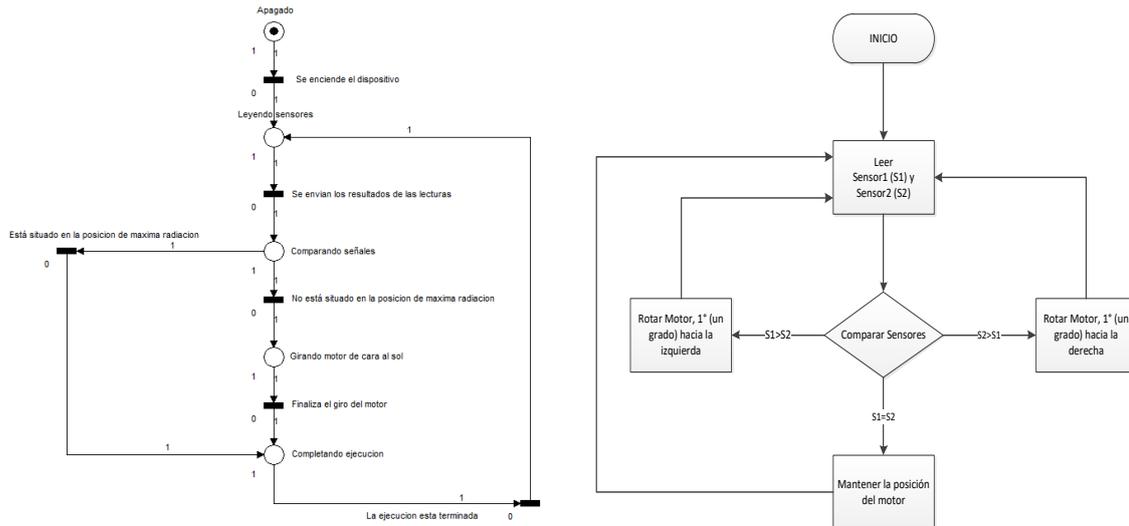


Figura 3. Diagramas de Petri y funcional del sistema de control.

La implementación del sistema de control del panel se hizo a partir de un prototipo a escala manejado por una placa Arduino Uno, la cual posee las entradas y salidas necesarias para realizar la lectura de los sensores LDR y el control de los servomotores.

Para la conversión de la señal se hace uso del módulo A/D de la placa Arduino. Cada sensor LDR disminuye su resistencia con el aumento de intensidad de luz incidente. Así, se puede hacer la comparación entre los dos sensores para saber de dónde proviene la mayor intensidad de luz.

Se definió un sentido de giro para el actuador (un servomotor Futaba S3003) cuando las señales de cada sensor sean diferentes, enviando una señal PWM al servomotor para indicarle en qué posición debe situarse y así alinearlos con la posición del Sol. En caso de ser iguales, el servomotor deberá permanecer inmóvil pero energizado para generar un torque suficiente para sostener el panel.

En la Figura 4 se muestran un diagrama parcial de conexión y una fotografía del prototipo desarrollado por los alumnos en la etapa de prueba con 2 sensores y un servomotor.

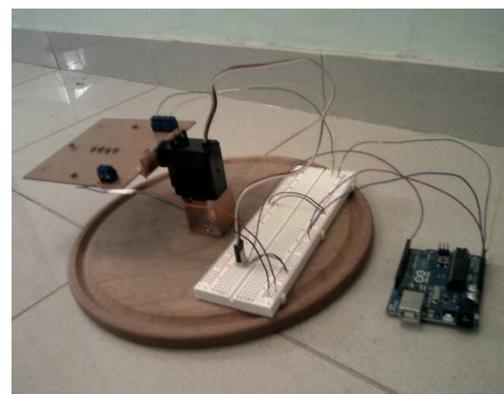
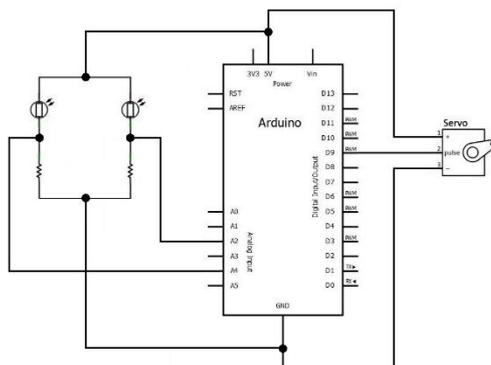


Figura 4. Diagrama de conexiones y prototipo del sistema de control del panel

## 2.2. Domótica e Internet de las Cosas

Como tema de una práctica supervisada se desarrolló una aplicación para el control de artefactos eléctricos en el hogar, no sólo para mejorar el confort y la seguridad sino también para racionalizar el consumo de energía.

La domótica controla y automatiza la gestión inteligente de la vivienda. Aporta confort, comunicación y seguridad, además de gestionar eficientemente el uso de la energía, favoreciendo el ahorro de agua, electricidad y combustibles. Mediante la incorporación de sistemas domóticos en el hogar, puede gestionarse inteligentemente la iluminación, climatización, agua caliente sanitaria, el riego, los electrodomésticos, etc., aprovechando mejor los recursos naturales y reduciendo la factura energética mientras gana en confort y seguridad.

Desde un punto de vista social, el ahorro y la eficiencia energética no sólo aseguran el abastecimiento energético y mejoran el medio ambiente, sino que también ayudan a incrementar la competitividad del sector industrial, beneficiando el aumento del Producto Bruto Interno del país.

La domótica es una tecnología que aún tiene mucho por crecer. Existe una barrera a superar para que cada vez más gente la adopte en sus hogares: la necesidad de que un técnico especializado instale el servicio y deba configurarlo completamente. Esto genera desconfianza y un gasto extra que la sociedad no siempre está dispuesta a aceptar.

Fue así como se planteó desarrollar una solución para evitar esta situación, facilitando al usuario la posibilidad de visualizar, modificar y agregar dispositivos al sistema, además de hacerlo flexible y escalable sin dificultades. Para ello se formularon los siguientes objetivos:

- Brindar "dispositivos inteligentes" conectados a Internet que manejen los distintos sensores y artefactos electrodomésticos: detectores de temperatura, movimiento, sonido, luces, calefactores, aires acondicionado, alarmas, etc.
- Brindar una Aplicación Web y Mobile para poder conocer el estado de los dispositivos y modificarlos de forma remota.
- Guardar un histórico de los cambios de estado de los dispositivos.
- Brindar un sistema de usuarios para asegurar la privacidad.

En este caso la metodología de codiseño llevó a un particionamiento hardware/software conformado por un protocolo para la comunicación entre sensores y una placa de hardware abierto para la implementación de estos sensores.

El control del sistema se implementó a través de la comunicación de los dispositivos con un servidor en la nube mediante uno de los protocolos más utilizados para el IoT (**Internet of Things**), el MQTT, basado en el modelo Publicación/Suscripción sobre TCP/IP (también se lo puede correr sobre UDP, pero hay que realizar un mayor control sobre el estado de los paquetes) [12].

El protocolo MQTT (**Message Queue Telemetry Transport**) es utilizado para la comunicación machine-to-machine (M2M). Está orientado a la comunicación de sensores, debido a que consume muy poco ancho de banda y puede ser utilizado en la mayoría de los dispositivos empotrados con pocos recursos (CPU, RAM, ...). Su arquitectura sigue una topología de estrella, con un nodo central que hace de servidor o "broker" con una capacidad de hasta 10000 clientes. El broker es el encargado de gestionar la red y de transmitir los mensajes para mantener activo el canal. Los clientes mandan periódicamente un paquete y esperan la respuesta del broker. La comunicación puede ser cifrada entre otras muchas opciones.

El servidor, llamado Broker, va a escuchar los distintos Topics (canales), reenviando los mensajes que los publicadores envíen a todos los que estén suscritos al canal facilitando la comunicación de dos vías (Figura 5).

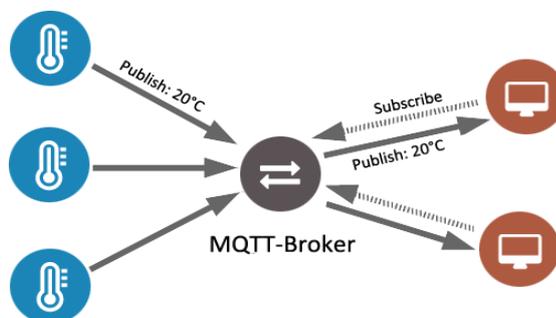


Figura 5. Modelo Publicación/Suscripción de MQTT

El modelo es escalable de manera horizontal instalando varios Brokers que se comuniquen entre ellos, permitiendo crecer de forma sencilla.

La principal ventaja que ofrece este protocolo es que cada cliente abre una conexión TCP contra el Broker, evitando la necesidad de establecer conexiones cada vez que se quiera enviar un paquete. De esta manera se puede atravesar el router doméstico, pudiendo recibir información y evitando la necesidad de que el dispositivo deba recurrir a la técnica de "Polling", consultando constantemente al servidor sobre si tiene que cambiar o no su estado (gastando así muchos recursos y ancho de banda).

Se decidió trabajar con una arquitectura orientada a microservicios porque brinda una robustez, tolerancia a fallos y escalabilidad horizontal superior a un sistema monolítico.

Consta principalmente de 2 secciones, ambas dispuestas en el servidor en la nube (Figura 6):

- **Servidor web**

Los usuarios se comunican e interactúan con él pudiendo ver y modificar el estado de sus dispositivos en tiempo real.

Está implementado en PHP sobre el framework Laravel usando una persistencia con PostgreSQL a través del ORM Eloquent.

Contiene un sistema de usuarios con contraseñas encriptadas y un panel de control para agregar, editar y remover dispositivos dinámicamente de forma tal que el cliente final pueda comprar un nuevo dispositivo inteligente y registrarlo sin intervención de un administrador. Permite además ver un histórico de los estados por los que pasó con sus respectivos horarios.

- **Traductor MQTT-HTTP (Parser)**

Comunica los dispositivos con el servidor web y con un Broker en la nube, pasando de un protocolo a otro sin mantener estados, lo que permite una escalabilidad directa.

Su responsabilidad es conectarse al Servidor MQTT-Broker, suscribirse a todos los tópicos existentes y reenviar todos los mensajes transmitidos por los dispositivos al servidor web a través de HTTP.

El servicio se desarrolló en NodeJs para mantener conexiones asincrónicas abiertas, evitando quedar bloqueado y ocupar muchos recursos del servidor.

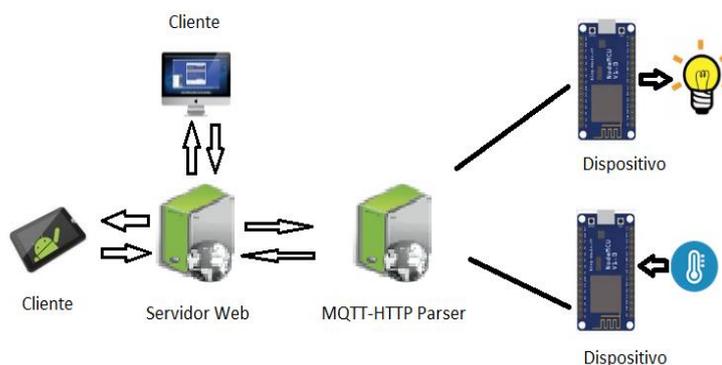


Figura 6. Diagrama del sistema completo

### Dispositivos inteligentes

Se utilizó la placa de hardware abierto ESP8226 orientada al IoT (Figura 7). Se la eligió ya que cuenta con varias ventajas por sobre otros competidores como Raspberry PI o Arduino [13]:

- Es sencilla de programar, pudiéndose utilizar los lenguajes JavaScript, microPython o el nativo Processing de Arduino.
- Cuenta con una gran comunidad.
- Viene con un Shield integrado de Wifi.
- Es económica y de muy bajo consumo eléctrico.
- Tiene un tamaño muy pequeño, pudiéndose integrar en cualquier espacio.

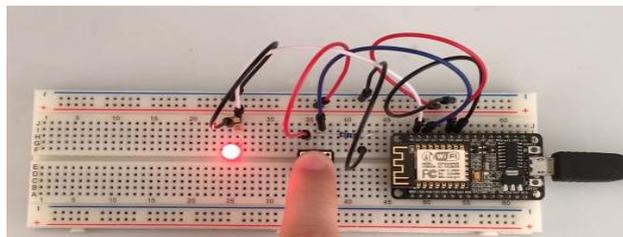


Figura 7. Placa ESP8226 en etapa de prueba

### 2.3. Sistema para Adquisición de Datos de Consumo Energético

El objetivo consistió en desarrollar un prototipo que permita monitorear el consumo energético de cualquier equipo/máquina que se conecte a un sensor de corriente, para luego implementar un componente que contribuya a un sistema computacional destinado a la gestión energética en PyMES.

El equipo/máquina también puede ser asimilado a la carga presentada por todo el consumo de una vivienda o edificio, en cuyo caso el sistema permitiría medir y controlar el gasto de energía eléctrica en tiempo real.

Para aplicar la metodología de codiseño se partió de un diagrama software/hardware como se muestra en la Figura 8.

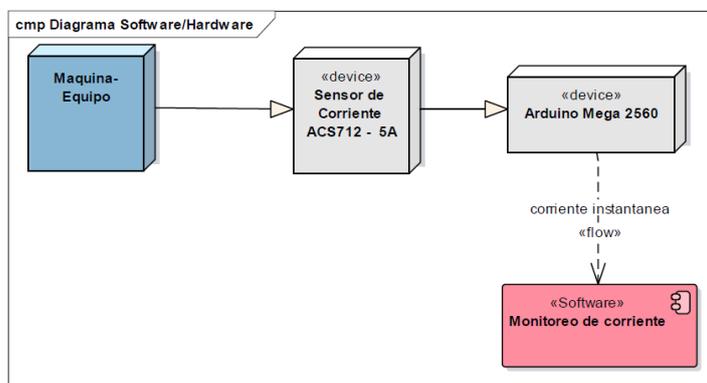


Figura 8. Diagrama utilizado para el particionamiento software/hardware

Los datos de corriente instantánea que devuelve el sensor son usados para hacer cálculos del consumo de energía del equipo usado en la prueba. El consumo y la corriente instantánea se muestran a través de una interfaz software hecha con Matlab.

Para realizar la especificación del sistema se utilizó el diagrama estado/transición (Figura 9) lo que permite visualizar en un sólo esquema todas sus actividades y la vinculación entre ellas.

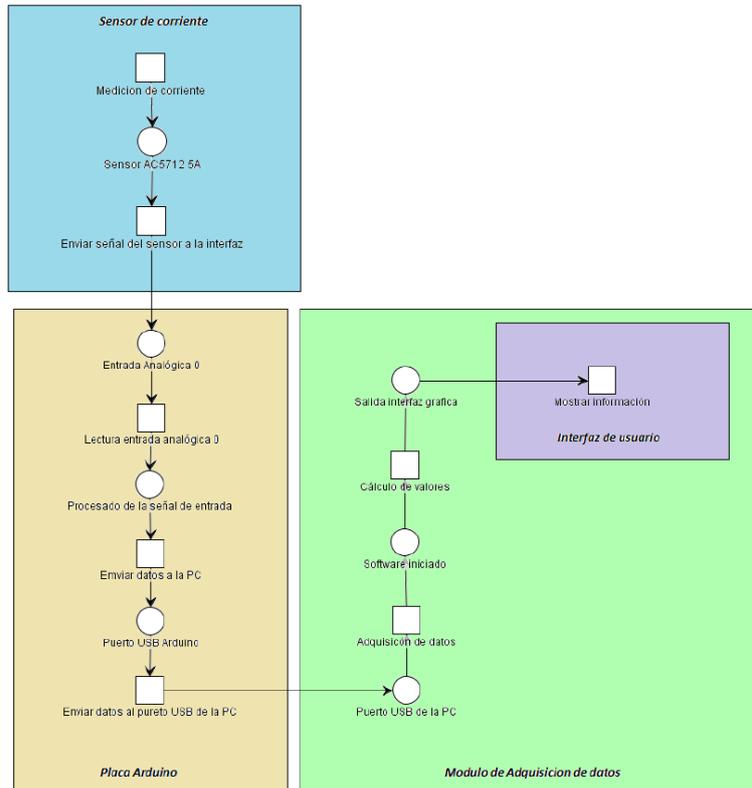


Figura 9. Especificación del sistema mediante el diagrama estado/transición

## Particionamiento Hardware/Software

### Componentes Hardware

Se utilizó el sensor ACS712 para medir la corriente cuya salida se conecta a una de las entradas analógicas de una placa Arduino ATmega 2560 (Figura 10).

El ACS712 es un sensor de corriente con tecnología de efecto Hall que provee un voltaje de salida proporcional a la corriente que fluye en el circuito. Tiene una capacidad de 5A tanto en corriente alterna como en corriente directa [14].

La placa Arduino Mega 2560 está basada en el microcontrolador ATmega2560. Tiene 54 pines de entradas/salidas digitales (14 de las cuales pueden ser utilizadas como salidas PWM), 16 entradas analógicas, 4 UARTs (puertos serial por hardware), cristal oscilador de 16 Mhz, conexión USB, jack de alimentación, conector ICSP y botón de reset. Puede trabajar simplemente conectándola a una PC por medio de un cable USB o con una fuente de alimentación externa y su selección es automática [15].

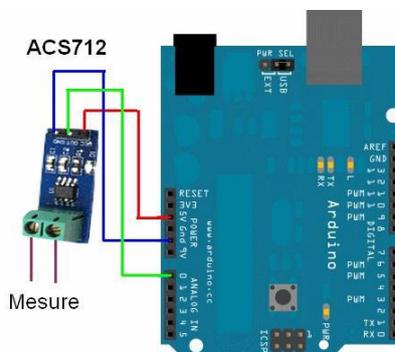


Figura 10. Esquema de conexiones del sensor a la placa Arduino

### Componentes software

El software del prototipo está dividido en dos partes: por un lado, el sketch de Arduino, el cual se encarga de obtener los datos a través del muestreo de la señal del sensor. Se hacen varias lecturas y se envía su sumatoria hacia el siguiente componente software. La otra parte está constituida por un software con una interfaz gráfica hecha en Matlab, que recibe los datos vía interfaz serial, y

calcula el valor eficaz de la corriente y el consumo de energía en tiempo real. Ambos valores son mostrados por pantalla.

Para el cálculo del consumo energético se debe emplear el valor medio cuadrático de la corriente (valor eficaz) mediante la ecuación:

$$I_{rms} = \sqrt{\frac{1}{n} \sum_{i=1}^n X_i^2}$$

donde los valores de  $X_i$  son las muestras de la magnitud medida. El valor de la muestra "n" variará por varios factores, como por ejemplo la frecuencia de muestreo o la velocidad que necesitemos en nuestro software. Empíricamente se llegó a la conclusión de que se obtienen buenos resultados con un  $n=100$ . Este valor permite además obtener unas 100 muestras de  $I_{rms}$  por segundo (teniendo en cuenta una frecuencia de muestreo ideal de 10000 muestras por segundo).

Los valores obtenidos con son enviados a través de la interfaz serial a un programa hecho en Matlab.

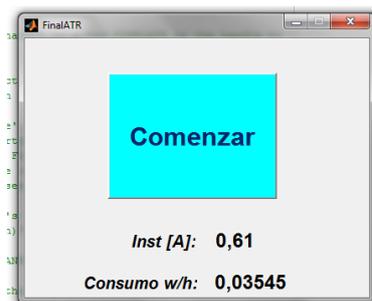


Figura 11. Interfaz de usuario

Este programa hará los cálculos necesarios para obtener el mencionado valor medio cuadrático de la corriente, con el cual, sabiendo el voltaje del circuito y a través de la medición del tiempo entre lecturas de sucesivos valores de corriente efectiva, se logra calcular el consumo de energía (Figura 11).

### 3. CONCLUSIONES

A través de estos trabajos, realizados y resueltos satisfactoriamente, los alumnos han podido tener una experiencia concreta sobre las características energéticas como centro de tarea, utilizando las herramientas estudiadas para resolver cada problemática.

Desde el punto de vista didáctico, esta forma de trabajo permite asociar herramientas gráficas y de software para la obtención de un diseño en hardware sin necesidad de profundos conocimientos de electricidad o electrónica. El hecho de obtener la implementación completa del modelo propuesto estimula el aprendizaje de todos estos temas de una forma más relacional y coordinada. La idea de partir de un modelo en papel hasta obtener un prototipo funcionando, crea un mayor grado de entusiasmo que el hecho de estudiar los temas en forma aislada y sin un objetivo claro, estimulando el desarrollo de proyectos sobre temas prácticos que de otro modo nunca llegan a implementarse.

Los resultados obtenidos a través de varios años de experiencia continua en la cátedra demuestran que genera en el alumnado un entusiasmo particular, que corresponde a la concreción del proceso completo: Especificación – Simulación – Programación – Implementación.

Las tareas se realizan durante todo el año lectivo y dependiendo del problema a resolver, el curso se divide en comisiones de 2, 3 ó 4 alumnos, siendo cada uno de ellos responsable de una tarea en particular, pero con una presentación completa a través de un coloquio, donde cada alumno es interrogado con preguntas del proceso completo.

Como indicador adicional, que consideramos permite consolidar la metodología, en el último año se incorporaron varios alumnos al Laboratorio de Investigación CODAPLI del Departamento de Sistemas de Información de la UTN Facultad Regional La Plata, para continuar y ampliar sus conocimientos sobre sus trabajos, con el objetivo de intentar llevarlos al plano comercial como emprendimientos propios.

Una tarea prevista consiste en generar "grupos de competencia" sobre un mismo trabajo para lograr mayor integración y espíritu crítico en el curso.

### 4. REFERENCIAS

- [1] E. Villar y A. López, “Especificación de sistemas embebidos en Sistemas digitales: elementos para un diseño a alto nivel”, A.García, Comp. Bogotá: Programa de Ciencia y Tecnología para el Desarrollo – Ediciones Uniandes, 1999, pp. 25 – 96.
- [2] Michael A. Smith, “Object-oriented Software in ADA 95” Second Edition. School of Computing University of Brighton.
- [3] H. Mazzeo, J. Rapallini, A. Quijano, “Utilización de ADA y VHDL para el Codiseño Hardware Software de Sistemas de Tiempo Real, especificados mediante Redes de Petri”. I Congreso de Microelectrónica Aplicada. UnLAM, Buenos Aires, julio 2010. ISBN: 978-987-1635-14-6.
- [4] H. Mazzeo y J. Rapallini, “Método didáctico para la enseñanza del Codiseño Hardware Software en Sistemas de Tiempo Real”. VII Jornadas Argentinas de Robótica, UNICEN, Olavarría, noviembre 2012. ISBN: 978-950-658-316-3.
- [5] H. Mazzeo, J. Rapallini, M. Zabaljauregui, O. Roríguez, “Laboratorio remoto con placa didáctica para desarrollo de prácticas con circuitos digitales”. VII Congreso de Microelectrónica Aplicada, Universidad Nacional de San Luis, octubre 2016. ISBN: 978-987-733-068-7.
- [6] H. Mazzeo, J. Rapallini, M. Zabaljauregui, O. Roríguez, “Diseño de plataforma remota para prácticas de laboratorio”. II Jornada Argentina de Tecnología, Innovación y Creatividad. Universidad CAECE, Mar del Plata, noviembre 2016. ISBN; 978-987-46267-0-7.
- [7] “Los Paneles Solares Fotovoltaicos”, [http://www.sitiosolar.com/paneles\\_fotovoltaicas.htm](http://www.sitiosolar.com/paneles_fotovoltaicas.htm), 2009.
- [8] D. G. Bragoni, J. Baron, M. Cortellezzi, “La situación Energética Local: Balance y perspectiva de una problemática compleja”, Universidad Nacional de Cuyo, Abril de 2004.
- [9] J. F Curtin, J. Magolis, “El Uso Eficiente de la Energía”, Departamento de Estado de Estados Unidos, Oficina de Programa de Información Internacional, vol. 14, numero 4, Abril 2009.
- [10] J. R. Cedeño, “La Situación Energética Mundial y sus Implicaciones para Puerto Rico”, CIAP, Instituto de Ingenieros Electricistas, marzo de 2009.
- [11] Nota Conceptual ONUDI, “Hacia una Agenda Energética Integrada mas allá de 2020”, Conferencia Internacional de Energía, Viena (Hofburg), Austria, Junio de 2009.
- [12] MQTT Official website <http://www.mqtt.org>,
- [13] ESP8226, Espressif started guide <https://www.espressif.com/en/support/explore/get-started/esp8266/getting-started-guide>
- [14] «Especificacion - Sensor ACS712 - 5 Amps,» [En línea]. Available: <https://www.sparkfun.com/datasheets/BreakoutBoards/0712.pdf>
- [15] «Especificacion - Arduino Mega 2560» [En línea]. Available: <https://www.arduino.cc/en/Main/ArduinoBoardMega2560>