

# Técnicas y recursos electrónicos de bajo coste para la Formación en el Grado en Ingeniería en Tecnologías Industriales basadas en instrumentación micro-controlada con *Arduino* y simulaciones de circuitos electrónicos

Juan José González de la Rosa, José Carlos Palomares Salas, Agustín Agüera Pérez, Manuel Sanmartín de la Jara, José María Sierra Fernández, Álvaro Jiménez Montero, José Gabriel Ramiro Leo

Departamento de Ingeniería en Automática, Electrónica, Arquitectura y Redes de Computadores (IAEARC)), Grupo de Investigación en Instrumentación Computacional y Electrónica Industrial (ICEI-PAIDI-TIC-168). Escuela Politécnica Superior de Algeciras, Avda. Ramón Puyol S/N, E-11202-Algeciras. [juanjose.delarosa@uca.es](mailto:juanjose.delarosa@uca.es)

**RESUMEN:** El proyecto de innovación asociado a esta memoria describe tres técnicas de bajo coste, con sus correspondientes recursos, asociados a la formación en las Ingenierías en Tecnologías Industriales: los equipos micro-controlados, la simulación electrónica y la instrumentación virtual. Con respecto al primero, se han desarrollado un conjunto de prácticas de laboratorio y, como colofón, un coche-robot guiado mediante la luz controlado mediante una placa programable *Arduino*. Las aplicaciones de simulación electrónica abarcan varios circuitos electrónicos con los que el alumno puede interaccionar durante su formación, y han sido desarrolladas en la hoja de cálculo, fácilmente exportable a software libre, sin necesidad de simulador electrónico específico. Finalmente, se han desarrollado paneles de instrumentación electrónica virtual, a través de los cuales, el alumno puede analizar señales reales y analizar los fundamentos de los instrumentos electrónicos de medida. Dichas técnicas están siendo ya implantadas en la enseñanza de los nuevos grados desde el curso 2011-2012, y pretenden dar continuidad a la labor innovadora que se realiza en nuestro departamento y en el grupo de investigación PIDI-TIC-168 desde su fundación en 1999.

**PALABRAS CLAVE:** innovación, mejora docente, docencia, instrumentación electrónica, circuitos electrónicos, simulación, micro-controlador, *Arduino*, software libre, bajo coste.

## INTRODUCCIÓN

La eficaz introducción de las nuevas tecnologías en las enseñanzas del grado en Ingeniería en Tecnologías Industriales está jugando un papel crucial en el desarrollo de métodos de formación basados en tecnologías bajo coste, que tienen un valor añadido doble, ya que en periodos de crisis económica mantienen el nivel de la docencia e, incluso en determinados casos, son capaces de elevarlo.

Durante el presente proyecto de innovación se han desarrollado prácticas de laboratorio y un mini-proyecto de sistema micro-controlado, simulaciones de circuitos electrónicos en hojas de cálculo e instrumentos electrónicos de medida virtuales, comenzando ya a emplearlas durante el curso 2011-12, en asignaturas del Grado en Ingeniería en Tecnologías Industriales, la I.T.I. en Electrónica Industrial y el Segundo ciclo de Ingeniería Industrial en la E.P.S. de Algeciras.

Durante la exposición del sistema micro-controlado, hemos pensado conveniente incluir ciertos aspectos técnicos mezclados con explicaciones didácticas y un vídeo (YouTube), con el fin de que el siguiente apartado sea válido para especialistas y también no especialistas en este campo, que puedan apreciar tanto el contenido cualitativo como el cuantitativo del presente proyecto de innovación.

## SISTEMA MICROCONTROLADO

### Introducción

*Arduino* es una plataforma de desarrollo de computación física (*physical computing*) de código abierto, basada en una

placa micro-controlada y un entorno de desarrollo para crear software (programas) para la placa. Hoy en día la placa programable *Arduino* está adquiriendo cada vez más importancia en el mundo de la Electrónica para todo tipo de aplicaciones debido a su flexibilidad y bajo coste.

Entre otras posibilidades, *Arduino* permite crear objetos interactivos, leyendo datos provenientes de una gran variedad de interruptores y sensores y controlar multitud de tipos de luces, motores y otros actuadores físicos. Los proyectos con *Arduino* pueden ser autónomos o/y comunicarse con un programa (software) que se ejecute en otro ordenador (1)-(5).

En nuestra aplicación utilizaremos el modelo más básico (*Arduino UNO*) que, entre otras muchas características, posee 14 pines que pueden usarse como entradas o salidas digitales, 6 entradas analógicas y 6 salidas analógicas para modulación PWM. Destacamos que *Arduino* es capaz de suministrar toda la potencia necesaria para controlar los elementos que tenemos conectados, aunque más adelante explicaremos brevemente el uso de un sistema de potencia exclusivo para los motores.

### Elementos del equipo micro-controlado

Se describen cada uno de los elementos que constituyen la aplicación de control del robot. Dos motores eléctricos de 3 V D.C. y 1,6 W, con reductora. Se muestra en la Fig. 1.



Figura. 1. Motor eléctrico que se va a controlar.

Estos dos motores se encargan de proporcionar la potencia necesaria a las ruedas del coche-robot. Demandan una potencia que *Arduino* no puede suministrar, por lo que se ha de conectar entre ambos una etapa de potencia constituida por un Controlador dual de puente completo (L298N) y dos puentes de diodos, que ha sido diseñada adicionalmente en nuestro laboratorio. Este chip permite la alimentación de los motores desde una fuente externa a *Arduino*, siguiendo las órdenes emitidas por él. De esta forma se pueden controlar gran variedad de motores de corriente continua, permitiendo este chip hasta 200 W por motor, a 48 V. Los puentes de diodos cumplen la función de diodos de libre circulación, permitiendo la devolución de energía de los motores a la fuente en la situación de frenada. La etapa de potencia queda como indica la Fig. 2.

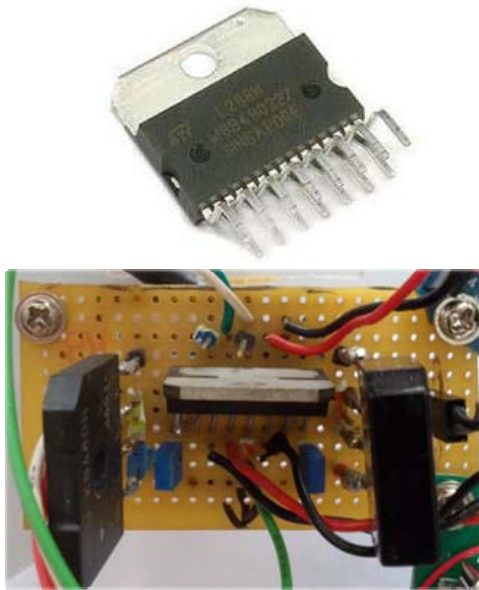


Figura. 2. Etapa de potencia.

El juego de fotodiodos, mostrado en la Fig. 3, constituyen los "ojos" del coche. Son los encargados de captar la luz.

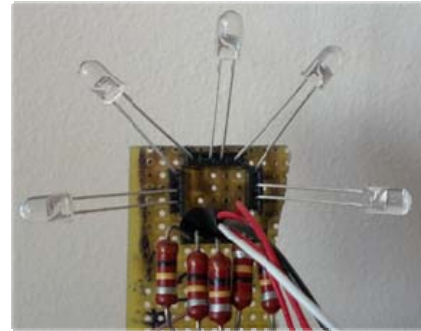


Figura. 3. Fotodiodos en una placa adicional construida en nuestro laboratorio. Si la luz incide en la dirección de uno de ellos, el robot se dirigirá a la fuente de luz.

Los receptores se conectan a las entradas analógicas de la placa. De esta manera se comprueba cuál da el valor máximo, y ese será el que más luz recibe. *Arduino* será el encargado de dar la instrucción correspondiente a los motores para que se desplacen hacia donde ha captado esa luz. Para comunicar información se instala una pantalla LCD. Todo lo anteriormente indicado se conecta a la placa y se coloca en un chasis de metacrilato, como se muestra en la Fig. 4.

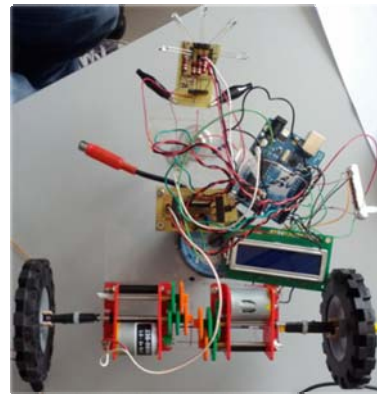


Figura. 4. Coche-robot completo:  
[http://www.youtube.com/watch?v=4cedJHsK\\_Hc](http://www.youtube.com/watch?v=4cedJHsK_Hc)

Este coche robot, además de poner el colofón a las prácticas de laboratorio desarrolladas, ha servido para hacer una demostración a los alumnos de segundo de los Grados en Ingeniería en Tecnologías Industriales, con el fin de facilitarles la orientación, en el sentido de la elección de la especialidad y de sus salidas profesionales.

### Programación del robot basado en *Arduino*

Incluimos en este apartado conceptos más técnicos pero que en realidad son muy comunes en el campo de las TIC, y que están relacionados con la programación en lenguaje C. Esto es imprescindible para comprender el desarrollo del proyecto.

Con el fin de simplificar la programación y reducir el tamaño del programa principal, *Arduino* permite crear librerías para en la aplicación desarrollada. Por una parte para crear la librería necesitamos dos archivos: un archivo de cabecera (w / con extensión. H) y el código fuente (w / extensión. cpp). El archivo de cabecera contiene las definiciones de las funciones de la librería, mientras que el archivo del código fuente contiene la declaración de cada una de las funciones que utilizaremos en

la aplicación. En la Fig.5 se muestra un diagrama de bloques que muestra cómo interactúan ambos archivos.

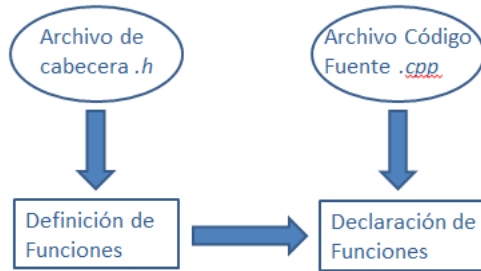


Figura. 5. Diagrama de bloques de los componentes de la librería.

El archivo de cabecera adopta el siguiente aspecto, con cada una de las definiciones de las funciones utilizadas en el código fuente.

```

/*
CocheDigital.h
Libreria para el control del Coche desarrollado por el
PAIDI-TIC_168 con control digital de los motores.
Abril 2012.
*/

#ifndef CocheDigital_h
#define CocheDigital_h

#include "Arduino.h"

class CocheDigital {
public:
    CocheDigital(int izq_en,int izq_avan,int izq_retro,int
der_en,int der_avan,int der_retro);
    void avanza();
    void retrocede();
    void avanza(int tiempo);
    void retrocede(int tiempo);
    void para();
    void frena();
    void derecha();
    void derecha(int tiempo);
    void izquierda();
    void izquierda(int tiempo);
    void derecha_avanza();
    void derecha_avanza(int tiempo);
    void izquierda_avanza();
    void izquierda_avanza(int tiempo);
    void derecha_retrocede();
    void derecha_retrocede(int tiempo);
    void izquierda_retrocede();
    void izquierda_retrocede(int tiempo);
private:
    int _izq_avan;
    int _izq_retro;
    int _der_avan;
    int _der_retro;
  
```

```

int _izq_en;
int _der_en;
};
#endif
  
```

Mientras que para el segundo archivo, llamado código fuente, vamos a exponer un ejemplo que ilustra el desarrollo de una de las funciones anteriormente declaradas en la cabecera. Concretamente, se realiza una breve descripción de la función *avanza()* que, como su nombre indica, es la encargada de que el coche avance en dirección recta, por lo cual le asignamos un valor alto a los pines de salidas digitales (*\_izq\_avan* y *\_der\_avan*), que son los que están conectados a ambos motores que moverán las ruedas del coche. El código de la función es:

```

void CocheDigital::avanza()
{
    digitalWrite(_izq_avan,HIGH);
    digitalWrite(_der_avan,HIGH);
    digitalWrite(_izq_retro,LOW);
    digitalWrite(_der_retro,LOW);
    digitalWrite(_izq_en,HIGH);
    digitalWrite(_der_en,HIGH);
}
  
```

En esta función observamos dos pines habilitados de ENABLE la función del *enable* se ve recogida claramente en la siguiente Tabla 1.

	Avanza	Retrocede	Operación
E=1	1	0	Avanza
	0	1	Retrocede
	1	1	Bloqueo
E=0	X	x	Libre

Tabla 1. Funciones de la variable ENABLE.

### Conclusiones relativas a Arduino

La placa programable *Arduino* puede constituir el cerebro de una gran variedad de aplicaciones. Se han desarrollado en el presente proyecto una batería de prácticas, cuyo colofón es el robot guiado por luz. Dependiendo del potencial de la aplicación utilizaremos un modelo u otro de *Arduino*, en nuestro caso hemos cubierto todos los pines del modelo *Arduino UNO*.

El robot guiado por luz ha sido la experiencia demostración final de las prácticas de laboratorio. Su funcionamiento se observa en el vídeo que hemos desarrollado: [http://www.youtube.com/watch?v=4cedJHsk\\_Hc](http://www.youtube.com/watch?v=4cedJHsk_Hc)

Como mejora, es necesario indicar que la colaboración del alumnado es esencial para el desarrollo de las prácticas, con el fin de optimizar el tiempo. En este sentido, es necesario poder disponer de más tiempo y de conectar el contenido de las prácticas con el resto de la asignatura donde estén incluidas, con el fin de optimizar el rendimiento de los estudiantes. Se trata en definitiva de fomentar el contacto con el profesor en el laboratorio previo trabajo autónomo del estudiante.

## INSTRUMENTACIÓN ELECTRÓNICA VIRTUAL

El empleo de la instrumentación electrónica virtual como técnica docente se ha extendido totalmente debido a su alta flexibilidad y bajo coste. En el presente proyecto hemos desarrollado con herramientas propias (software de entorno de programación gráfica LabVIEW™ e instrumentación hardware de laboratorio docente e investigadora). La Fig. 6 muestra una aplicación que adquiere la señal medida por un osciloscopio digital.

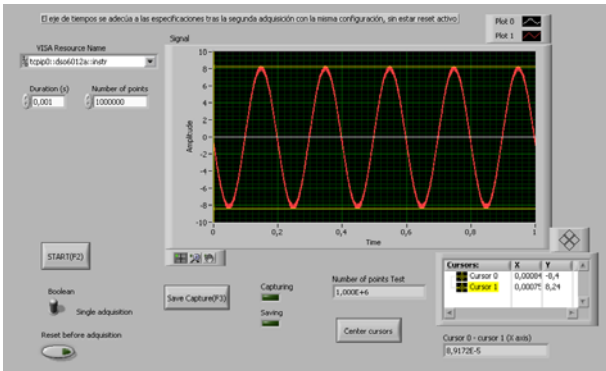


Figura 6. Panel principal de un instrumento electrónico virtual que controla un osciloscopio digital.

La siguiente Fig. 7 muestra la disposición física de los instrumentos en el laboratorio.



Figura 7. Disposición en el laboratorio del osciloscopio que está conectado a internet. Este osciloscopio posee una dirección IP fija, que fue suministrada por nuestro CITI.

La siguiente Fig. 8 muestra la pantalla del osciloscopio, en la que aparece la señal que ha sido capturada por el programa de control.

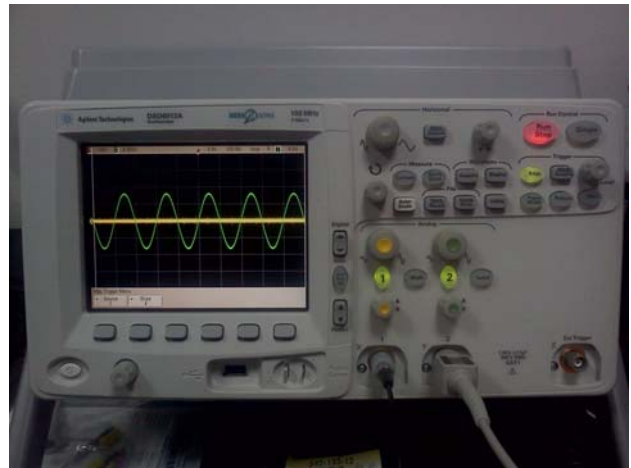


Figura 8. Disposición en el laboratorio del osciloscopio que está conectado a internet. Este osciloscopio posee una dirección IP fija, que fue suministrada por nuestro CITI.

## Conclusiones del uso de los paneles de instrumentación virtual

Se ha podido comprobar la buena respuesta del alumnado al empleo de la instrumentación interactiva y la gran realización personal de poder controlar hardware con programas de propósito específico y que están muy orientados a las salidas profesionales.

Por otra parte, sería necesario poder disponer de una versión de estudiante de ese programa para que ellos diseñaran sus aplicaciones durante las horas de estudio.

## HOJAS DE CÁLCULO DE SIMULACIÓN DE CIRCUITOS ELECTRÓNICOS

Las hojas de cálculo desarrolladas permiten al estudiante simular sus circuitos electrónicos. Se han desarrollado también en Open-Office, con lo cual el coste del material está optimizado. La Fig. 9 muestra un ejemplo de hoja de cálculo. El alumno introduce los parámetros del circuito en una zona de celdas dedicadas a tal fin. Automáticamente, y como resultado de ello, se produce el análisis y representación gráfica de las curvas y gráficas involucradas en el circuito.

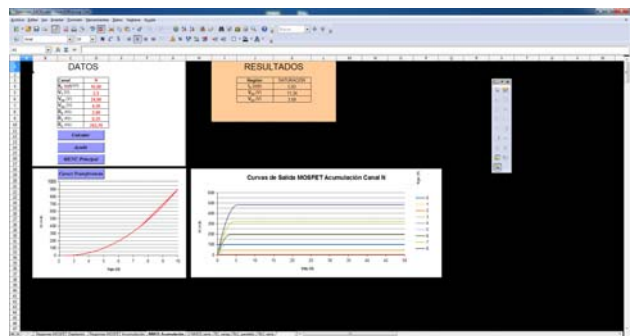


Figura 9. Ejemplo de análisis de un circuito parametrizado según una hoja de cálculo en OpenOffice.

## Conclusiones en relación al uso de las hojas de cálculo como simulador electrónico

Aunque existen en el mercado numerosos programas de simulación electrónica, las aplicaciones desarrolladas en el presente proyecto de innovación han sido desarrolladas ex profeso. Además, permiten al alumno comprobar su aprendizaje sin necesidad de construir el circuito. Tampoco es necesaria la instalación de ningún programa de Electrónica o Electricidad ni tener licencia de Microsoft, ya que está hecho en una aplicación de software libre.

## REFERENCIAS

1. Timmis, H. *Practical Arduino Engineering*, 2011, Ed. Apress.
2. Evans, B.W. *Arduino Programming Notebook*, lulu.com 2011.
3. Banz, M., *Getting Started with Arduino*, 2<sup>nd</sup> Edition”, O’Reilly Media, 2011.
4. Arduino, The Documentary: <http://arduinohedocumentary.org/>. Último acceso el 9 de julio de 2012.
5. McRoberts, M.R., *A Complete Beginners Guide to the Arduino Earthshine Electronics*, 2009.

## ANEXOS

Lo componen las prácticas de laboratorio de *Arduino* (Anexos 1-6) y una ampliación de la redacción del grado de consecución de los objetivos del proyecto (Anexo 7).

*PI1\_12\_001\_Anexo 1.pdf*

*PI1\_12\_001\_Anexo 2.pdf*

*PI1\_12\_001\_Anexo 3.pdf*

*PI1\_12\_001\_Anexo 4.pdf*

*PI1\_12\_001\_Anexo 5.pdf*

*PI1\_12\_001\_Anexo 6.pdf*

*PI1\_12\_001\_Anexo 7.pdf*

## AGRADECIMIENTOS

Queremos agradecer a la Universidad de Cádiz la consideración y e indispensable dotación el proyecto de Innovación Docente PI1\_12\_001 titulado “Instrumentos electrónicos de medida micro-controlados, virtuales y distribuidos y circuitos electrónicos. Aplicaciones en la formación en tecnologías industriales”. Sin olvidar el agradecimiento a la Junta de Andalucía por apoyar al Grupo de Investigación PAIDI-TIC-168 en Instrumentación Computacional y Electrónica Industrial, en el seno del cual se ha desarrollado este trabajo.