

MEMORIA FINAL

Actuaciones Avaladas para la Mejora Docente, Formación del Profesorado y Difusión de Resultados Modalidad C

Identificación de la actuación	
Código:	AAC_12_025
Título:	XX Congreso Universitario de Innovación Educativa en las Enseñanzas Técnicas (XX CUIEET)

Responsable	
Apellidos y nombre:	Morgado Estévez, Arturo
Correo electrónico:	arturo.morgado@uca.es
Departamento:	Ingeniería en Automática, Electrónica, Arquitectura y Redes de Computadores.

1. Describa la contribución a la actuación de cada uno de los participantes. Copie y pegue las líneas que necesite para contemplarlos a todos y disponga del espacio que necesite.

Apellidos y nombre:	Arturo Morgado-Estévez
Coautor de la comunicación	

Apellidos y nombre:	Fernando Pérez-Peña
Coautor de la comunicación	

Apellidos y nombre:	Miguel Ángel López-Gordo
Coautor de la comunicación	

Apellidos y nombre:	Carlos Rodríguez-Cordón
Coautor de la comunicación	

Apellidos y nombre:	Manuel Matias-Casado
Coautor de la comunicación	

Apellidos y nombre:	Rafael Jesús Montero-González
Coautor de la comunicación	

Apellidos y nombre:	Jose María Rodríguez-Corral
Coautor de la comunicación	

Apellidos y nombre:	Carlos Rioja-del-Rio
Coautor de la comunicación	

2. Aporte el producto final generado para la difusión.

Trabajo: Aplicación de la plataforma Arduino como introducción hardware-software en asignaturas tecnológicas.

El trabajo fue aceptado con fecha 3 de julio de 2012 en el congreso **XX Congreso Universitario de Innovación Educativa en las Enseñanzas Técnicas (XX CUIEET)** que se celebró en Las Palmas de Gran Canaria del 18 al 20 de julio de 2012. Su página web es <http://www.eiic.ulpgc.es/xxcuiet/>.

Aplicación de la plataforma Arduino como introducción hardware-software en asignaturas tecnológicas

Arturo Morgado-Estévez, Fernando Pérez-Peña, Miguel Ángel López-Gordo, Carlos Rodríguez-Cordón, Manuel Matias-Casado, Rafael Jesús Montero-González, Jose María Rodríguez-Corral, Carlos Rioja-del-Rio.

Escuela Superior de Ingeniería, Universidad de Cádiz, C/ Chile nº 1, 11002, Cádiz, 956015100, 956015101, arturo.morgado@uca.es

Resumen

Se presenta una manera de aprendizaje con la plataforma Arduino y una wiki para estudiantes de distintas procedencias curriculares y equilibrar sus conocimientos de electrónica, hardware y software. Estos tutoriales refuerzan los conocimientos de procedencia, mejoran y equilibran los conocimientos complementarios y fortalecen sus relaciones. En este caso, se emplea como herramienta de aprendizaje una Wiki para que los estudiantes puedan avanzar en su aprendizaje a distintos ritmos y profundizar en los contenidos que adolecen y que emplearán más tiempo para su aprendizaje. Como plataforma hardware libre de bajo coste y alta versatilidad se emplea la placa "Arduino Leonardo" con una comunidad que respalda su uso. Con Arduino y mediante componentes básicos electrónicos, se realizan distintos montajes en formato tutorial donde se explica la parte electrónica y la parte de programación. Se incorporan audio-videotutoriales explicativos del montaje.

Palabras Clave: Plataforma Arduino Leonardo, Wiki, hardware-software, videotutoriales.

Abstract

It presents a way of learning with the Arduino platform and a wiki for students from different curricular and balances their knowledge of electronics, hardware and software. These tutorials reinforce the knowledge of origin, improve and balance the additional knowledge and strengthen their relationships. In this case, is used as a learning tool Wiki for students to advance in their learning at different rates and deepen the content that suffer and that used more time for learning. As hardware platform free of low cost and high versatility board is used "Arduino Leonardo" with a community that supports its use. With Arduino and using basic components electronic assemblies are made different tutorial format that explains the electronics and the programming part. It incorporates audio-video tutorials explaining the assembly.

Keywords: Leonardo Arduino platform, Wiki, hardware-software, video tutorials.

1. Introducción

A la hora de diseñar en las asignaturas contenidos de aprendizaje [1] para estudiantes hay que tener en cuenta un aspecto fundamental que son sus conocimientos previos. En general, un estudiante que vaya avanzando cada año en la titulación, se le presupone que tiene adquiridos las anteriores materias. En muchos de los casos, por

no tener un buen diseño el título de Grado o Máster o por razones de descoordinación entre contenidos de las asignaturas, el estudiante carece de ciertas bases que se suponen ya tienen en asignaturas de cursos mas avanzados.

También ocurre que a la hora de realizar proyectos fin de carrera (en adelante PFC) y para los futuros trabajos fin de grado, cuando se propone un tipo de proyecto, se observa que según la procedencia del título tienen ciertas carencias. De la experiencia de tutorar PFC en las titulaciones de Ingeniería Técnica Industrial y en la Ingeniería Técnica Informática así como las titulaciones de segundo ciclo, se ha notado este tipo de debilidad. En la Tabla 1 se observa de forma general las fortalezas y las debilidades de conocimientos cuando se proponen proyectos orientados con la robótica y la mecatrónica.

Tabla 1. Fortalezas y debilidades de conocimientos entre I. T. Industrial e I. T. Informática.

Grados e ingenierías	Herramientas de Programación Java, Qt, VisualStudio	Conocimientos de C	Hardware	Dispositivos electrónicos
Grados en ingenierías Informáticas	Muy alto	Muy alto	Normal	Muy bajo
Grados en ingenierías Industriales	Muy bajo	normal	Normal	Muy alto

De la experiencia de tutorar PFC relacionados con la robótica, los estudiantes procedentes de las titulaciones de informática tienen amplios conocimientos de programación y uso de herramientas software y carecen bastante de conocimientos de diseño electrónico. Este mismo razonamiento es al contrario en el caso de los estudiantes con procedencia de titulaciones de ingeniería industrial donde sus conocimientos electrónicos son bastante altos y de la parte de programación tienen conocimientos muy básicos comparados con los estudiantes de Informática. Es lógico este resultado ya que las titulaciones de Ingeniería Industrial y las de Ingeniería Informática tienen distintos perfiles, cada uno orientado a su rama de conocimientos.

Por ello, cuando queremos acometer un PFC multidisciplinar o cuando determinadas asignaturas exigen conocimientos previos, en muchos de los casos no previstos o

carentes de ellos, necesitamos que el grupo de estudiantes homogeneíen sus conocimientos a fin de avanzar progresivamente.

2. Wiki y Arduino

Entre los recursos que se usan generalmente para diseñar contenidos de aprendizaje están las wiki [2] que además de presentar la información en el orden que se establezca, el estudiante puede ir parando y saltando de un contenido a otro según si conocen o no la materia que se esté tratando. Esto nos lleva al paradigma del aprendizaje asíncrono [3] en el cual un estudiante determinado puede estar el tiempo que necesite en algunas partes y saltarse o ir más rápido en otras. Pensamos que este es el formato ideal para nivelar conocimientos a grupos de estudiantes con distintas procedencias de conocimientos en determinadas asignaturas o cuando se tutoren PFC al comienzo de su impartición o tutoración.

Con el formato de wiki se ha diseñado un tutorial básico para adquirir conocimientos de hardware y de software. Este puede ser el objetivo de asignaturas relacionadas con conceptos de diseño electrónico que deban tener conocimientos previos de programación y viceversa y sobre todo para la tutoración de PFC que abarquen mayores conceptos de los impartidos en las asignaturas de la carrera a fin de realizar y acometer proyectos de mayor envergadura. Para este caso se ha empleado la plataforma de hardware libre Arduino [4] y el entorno Sketch Open Project [5] que tiene asociado.

La decisión de emplear las placas Arduino es debido a que es una plataforma de hardware libre donde se conoce su esquema electrónico, los componentes empleados y su disposición en la placa y que además está documentada totalmente como para poder emplearla en múltiples soluciones como cerebro de procesado junto a sensores y activadores. Además, queda avalada por la comunidad [6] que tiene detrás de ella y que continuamente aporta mayores alternativas de uso en los campos [7] de la electrónica, robótica, mecatrónica, informática, control de procesos, sensado de señales, interfaz entre sistemas electrónicos y PCs, sistemas empujados, comunicaciones entre sistemas, desarrollo de instrumentación, diseño de prototipos hardware y software, almacenamiento de información, etc. Se aportan tutoriales [8] de montajes, librerías para control de dispositivos, interfaces con software de alto nivel, etc. Se tiene en el mercado distintos kits básicos para empezar este tipo de tutoriales a

un coste bajo. El empleado para las prácticas en la asignatura es el Starter Kits – Arduino de la web de Cooking hacks [9] modificado con la placa Arduino Leonardo.

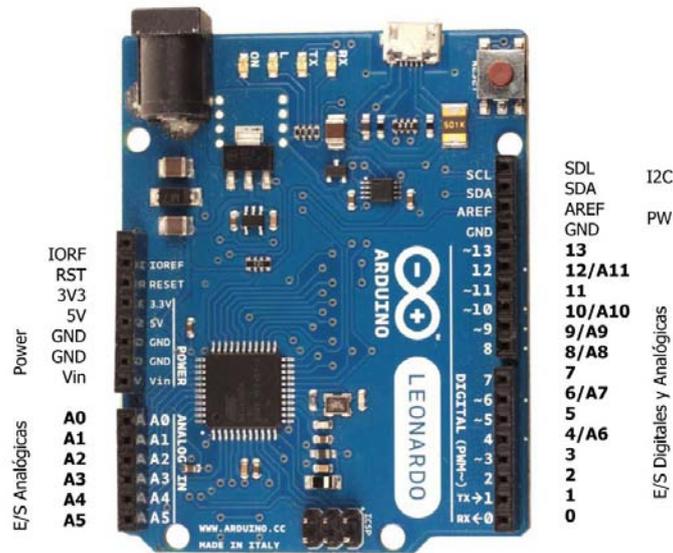


Figura 1. Placa Arduino Leonardo y asignaciones de pines de E/S

Junto a la placa, se emplea el entorno de programación libre Sketch Open Project donde se programa en lenguaje C [10] y el cual sirve para abrir, guardar y verificar código C y compilarlo y subirlo a la placa Arduino mediante un cable USB.

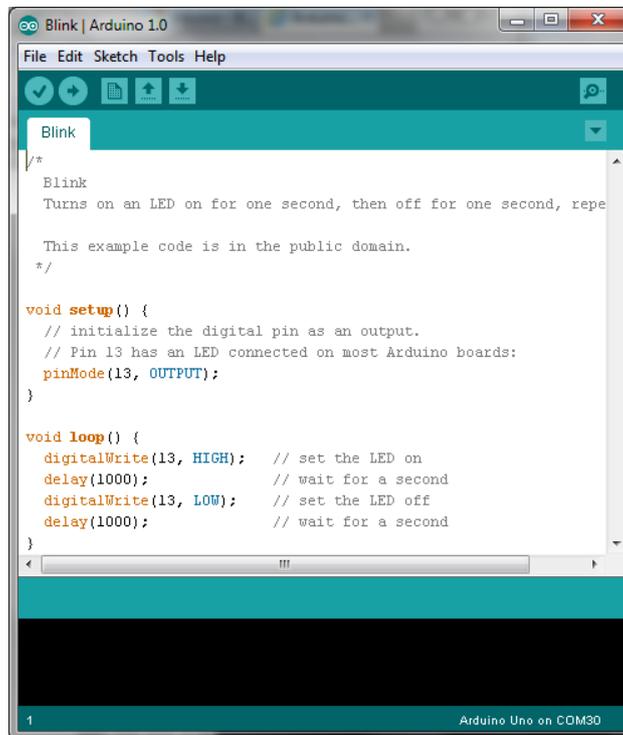


Figura 2. Entorno de programación Sketch Open Project de Arduino

El entorno de programación Sketch Open Project se puede instalar en cualquier sistema operativo de cualquier PC.

3. Contenido de tutorial con Arduino

La página principal de la wiki es la que aparece en la figura siguiente. En ella se comienza explicando qué es Arduino, y los aspectos básicos del hardware.



Figura 3. Entrada wiki del tutorial de Arduino

La Figura 4 contiene la forma de instalar el software para programar y compilar los programas que se cargarán en la placa Arduino.



Figura 4. Instalación del software de Arduino

3.1. Tutoriales con Arduino

Se han propuesto 10 tutoriales básicos comenzando desde el más básico que es haciendo parpadear el led de la placa, el uso de potenciómetros, LDR, NTC y pulsadores hasta el control de servos. La figura siguiente contiene los tutoriales propuestos.



Figura 5. Tutoriales con Arduino

Todos los tutoriales tienen la misma estructura:

1. Objetivos
2. Material necesario
3. Montaje del hardware
4. Videotutorial
5. Código del programa
6. Explicación del código
7. Prácticas

3.2. Descripción de los apartados del tutorial

Se muestra en las siguientes figuras el cuarto tutorial donde se emplea un pulsador y un diodo. Aparecen todos los apartados indicados anteriormente.

En la Figura 6 está la tabla de contenido y los objetivos marcados en esta práctica. A través de los hiperenlaces se accede a las distintas partes del tutorial.

Proyecto cuatro: Añadimos un pulsador con resistencia

Contenido [ocultar]
1 Objetivos
2 Material necesario
3 Montaje del hardware
4 Videotutorial
5 Código del programa
6 Explicación del código
7 Prácticas
7.1 Práctica 1
7.2 Práctica 2

Objetivos

[\[editar\]](#)

En este caso, solamente haremos una modificación del anterior proyecto, puesto que hemos añadido una resistencia a la conexión entre el botón y la conexión del botón con la placa.

Figura 6. Tabla de contenido y objetivos de la práctica

La siguiente figura contiene las fotos de los componentes electrónicos que se van a emplear en la práctica. De esta manera es mas facil de identificar cada componente, sobre todo orientado a los estudiantes que no han empleado habitualmente componentes electrónicos.

Material necesario

[\[editar\]](#)

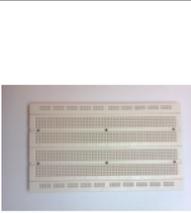
		
Placa Base Arduino Uno R3	Led Rojo	Led Verde
		
Resistencia de 150Ω	Jumper wires	Breadboard
		
Pulsador		

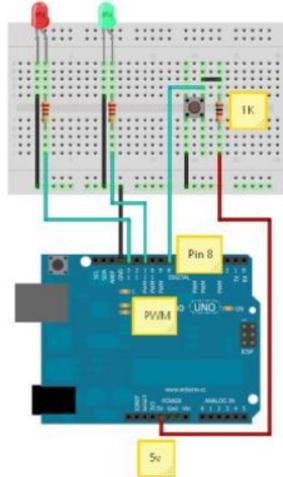
Figura 7. Componentes electrónicos empleados en la práctica

La Figura 8 tiene la distribución de los componentes electrónicos en la placa protoboard y la interconexión de ella con la placa Arduino. Esta imagen es muy útil para los estudiantes que no tienen destreza en el montaje y diseño de sistema electrónico ya que con el Esquema de interconexión que aparece mas abajo, sería suficiente para estudiantes con conocimientos en electrónica.

Montaje del hardware

[editar]

Ahora, comprobaremos que nuestra placa **Arduino** estará desconectada y sin energía, puesto que de no ser así podría dañarse tanto la placa, como el equipo. Una vez hemos realizado esta comprobación, pasaremos a realizar el montaje hasta tener un diseño tal y como este:



Montaje del hardware del proyecto tres: Añadimos un pulsador

No es importante el mero hecho de conectar todo tal y como está en esa imagen, pero si respetar el esquema que pondremos a continuación, puesto que sin eso, el proyecto no funcionaría. Pueden cambiar los colores de los cables, e incluso los huecos de la Breadboard donde conectan, pero solo eso.

Lo primero que haremos será colocar la placa **Arduino** de forma cómoda con la **Breadboard**, para así facilitar las conexiones que tengamos que realizar entre ambas. Luego pasaremos a colocar cada pieza en su lugar.

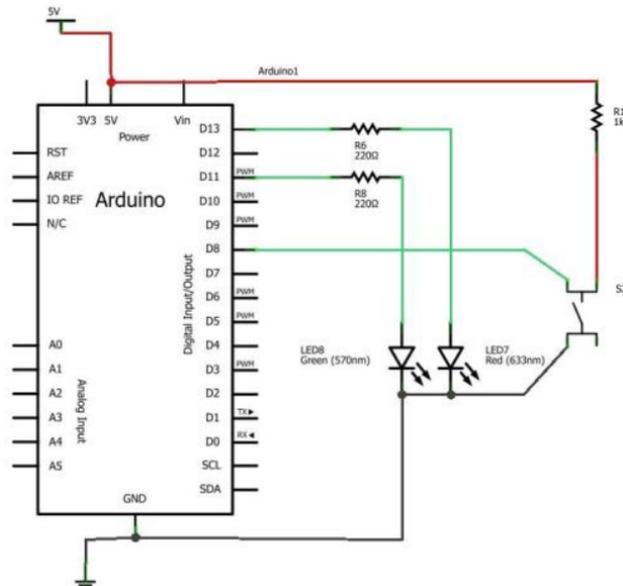


Figura 8. Colocación de los componentes y esquema electrónico

La Figura 9 tiene el videotutorial completo de montaje hardware de todo el sistema paso a paso y el funcionamiento de la práctica. El video tiene incrustado el audio donde se explica paso a paso el montaje de la práctica. Además, se explica conceptos básicos de electrónica conforme se va realizando el montaje. Al final se conecta la placa Arduino al ordenador para ver el funcionamiento de la práctica.

Videotutorial

[editar]

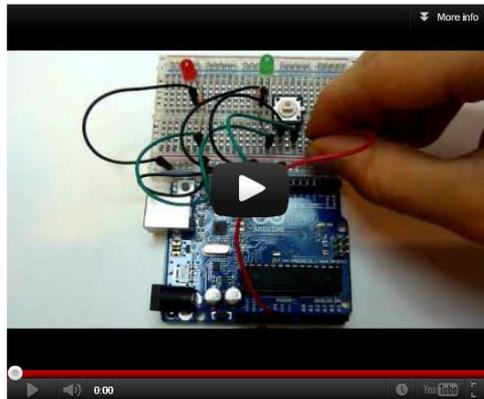


Figura 9. Audio-video tutorial del montaje de la práctica con Arduino

La Figura 10 tiene el código del programa. Está realizado de forma que copiando el código y pegándolo en el entorno de programación Sketch (ver Figura 2) está listo para poderlo compilar y programar la placa Arduino.

Código del programa

[editar]

```
int buttonPin =7; // Definimos el pin en que se encuentra el botón.
int ledPin2 = 11; // Definimos el pin en que se encuentra un led.
int ledPin = 12; // Definimos el pin en que se encuentra un led.
int buttonState = 0; // Esta variable guardará el estado del botón.
void setup() {

    pinMode(ledPin, OUTPUT);

    pinMode(buttonPin, INPUT); // Definimos el pin del botón como entrada de datos.

}
void loop(){

    buttonState = digitalRead(buttonPin); // Leemos el estado del botón con la función digitalRead().

    if (buttonState == HIGH) { // Comprobamos si el botón está HIGH, que sería apagado, o LOW, que sería encendido.
        digitalWrite(ledPin, HIGH); // Se enciende un Led
        digitalWrite(ledPin, LOW); // Se apaga el otro Led
    }
    else {
        digitalWrite(ledPin, LOW); // Se apaga un Led
        digitalWrite(ledPin, HIGH); // Se enciende el otro Led
    }

}
```

Figura 10. Código del programa de la práctica

La Figura 11 contiene la explicación del código orientado fundamentalmente a estudiantes con escasos conocimientos de programación y donde práctica a práctica se van fortaleciendo los conceptos de software.

Explicación del código

[\[editar\]](#)

Ahora pasaremos a explicar el código línea a línea, puesto que se trata del primer proyecto no se dejará pasar nada, ni lo más básico. No nos pararemos a explicar la sintaxis, ni la estructura del código C, puesto que podéis visitar la [lección de C](#) antes de comenzar este proyecto.

Las partes básicas del código, que se explicaron en el proyectos anteriores, serán omitidas.

digitalRead

Se trata de una función de Arduino, donde se lee el pin que, anteriormente, se ha inicializado como pin de entrada de datos. Generalmente suele guardarse el resultado en una variable, como hemos hecho en este proyecto.

```
if (buttonState == HIGH) { } else { }
```

Se trata de una estructura de control de programación, if...else. En if, se condiciona si la acción es verdadera, es decir, la realización del if, se realizaría cuando la opción que contiene sea real, y si no lo fuera, se realizaría el else.

Prácticas

[\[editar\]](#)

Si una vez hecho esto deseas seguir avanzando, puedes realizar algunas de las prácticas que se propondrán a continuación:

Práctica 1

[\[editar\]](#)

Se trata de realizar cambios en el código para que ambas bombillas se enciendan a la vez cuando se pulse el botón, y viceversa.

Práctica 2

[\[editar\]](#)

Realizar un blink (con tres parpadeos) tras pulsar el botón, en las dos bombillas a la vez.

Figura 11. Explicación del código del programa

Además se proponen prácticas alternativas sobre el mismo esquema electrónico donde requieren modificaciones del programa para su resolución.

4. Conclusiones

Se observa que con este tutorial basado en wiki, los estudiantes pueden adquirir conocimientos a distintas velocidades debido sobre todo a las carencias que en partes de estas materias tengan o quieran profundizar y avancen más rápidamente cuando estos conocimientos ya son conocidos. Este tipo de materias que contienen aspectos relacionados con el hardware y con el software, el empleo de la herramienta wiki de forma adecuada hace unificar conocimientos al nivel que sea necesario en las asignaturas a fin de tener a todos los estudiante con los conocimientos necesarios para avanzar de forma unificada en los siguientes apartados.

5. Agradecimientos

Los autores agradecen a la Unidad de Innovación Docente de la Universidad de Cádiz a través del proyecto concedido AAA_12_054, "Adaptación audiovisual del material

docente de asignaturas del área de Arquitectura y Tecnología de Computadores" el apoyo para la realización de este trabajo. Además, mencionar nuestro agradecimiento a los estudiantes Sergio Casas Rodríguez, Jose Antonio Marín López y Francisco Javier Sánchez Barrena por su excelente trabajo y colaboración.

6. Referencias

1. Cabral Vargas, Brenda. *Biblioteca digital: contenidos y aprendizaje*. Revista Interamericana de Bibliotecología. Vol. 29, No. 2, p. 119-140 (jul.-dic. 2006).
2. Augar, Naomi, Raitman, Ruth and Zhou, Wanlei, *Teaching and learning online with wikis, in Beyond the comfort zone: Proceedings of the 21st Australasian Society for Computers in Learning in Tertiary Education*. Conference ASCILITE Conference, Perth, 5-8 December, ASCILITE, {Perth, W.A.}, pp. 95-104 (2004).
3. Bourne, J.R.; McMaster, E.; Rieger, J.; Campbell, J.O. *Paradigms for on-line learning: a case study in the design and implementation of an asynchronous learning networks (ALN) course*, Frontiers in Education Conference, 1997. 27th Annual Conference. 'Teaching and Learning in an Era of Change'. Proceedings. , vol. 1, no., pp. 245-255, doi: 10.1109/FIE.1997.644851, (5-8 Nov 1997)
4. <http://arduino.cc/en/Main/Hardware> visitado el 30-05-2012.
5. <http://arduino.cc/en/Main/Software> visitado el 30-05-2012.
6. <http://arduino.cc/en/> visitado el 30-05-2012.
7. <http://arduino.cc/playground/> visitado el 30-05-2012.
8. <http://arduino.cc/en/Tutorial/HomePage> visitado el 30-05-2012.
9. <http://www.cooking-hacks.com/index.php/shop/arduino/starter-kits/arduino-starter-kit.html> visitado el 30-05-2012.
10. <http://arduino.cc/en/Reference/HomePage> visitado el 30-05-2012.