

# **Diseño de una guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/114)**

Memoria de resultados

Convocatoria de Innovación Docente – Curso 2016-2017



**VNiVERSIDAD  
D SALAMANCA**

CAMPUS OF INTERNATIONAL EXCELLENCE

Daniel Hernández de la Iglesia (Coordinador)

Alfonso González Briones

Álvaro Lozano Murciego

Alberto López Barriuso

Jorge Revuelta Herrero

Departamento de Informática y Automática  
Universidad de Salamanca - Facultad de Ciencias  
Plaza de la Merced, s/n 37008 Salamanca



## Contenido

1. Datos del proyecto .....	5
2. Introducción .....	6
3. Objetivos .....	7
4. Desarrollo del proyecto .....	8
Issues .....	8
Pull request .....	10
Projects.....	13
Insights .....	13
Estructura de la guía.....	15
5. Resultados logrados .....	15
6. Conclusiones .....	19



## 1. Datos del proyecto

**Título:** Guía didáctica sobre dispositivos hardware de código abierto para  
estudiantes de ingeniería

**Referencia:** (ID2016/114)

**Cuantía de la subvención:** 0€

**Coordinador del proyecto:** Daniel Hernández de la Iglesia

**Organismo:** Universidad de Salamanca

**Centro:** Facultad de Ciencias

**Investigadores que forman el equipo:**

Daniel Hernández de la Iglesia (Coordinador)

Alfonso González Briones

Álvaro Lozano Murciego

Alberto López Barriuso

Jorge Revuelta Herrero

**Duración:** Octubre 2016 – junio 2017

## 2. Introducción

En la actualidad existen numerosos proyectos colaborativos basados en hardware de código abierto o hardware libre como la placa Arduino o Raspberry Pi. Éstos son a su vez, la base para miles de nuevos proyectos: desde la creación de una impresora 3D a la automatización de una explotación agrícola. Las posibilidades que ofrecen estos dispositivos son ilimitadas. Conceptos como Internet of Things (IoT) o la industria 4.0 buscan hoy en día una revolución en la manera en la que los usuarios y las industrias interactúan con los objetos y el entorno que les rodea.

El objetivo de esta guía didáctica es facilitar a todos los alumnos de ingeniería de la Universidad de Salamanca, una primera toma de contacto con el mundo del Hardware Libre. En los últimos años, son muchos los proyectos de fin de grado en ingeniería que utilizan estos dispositivos para sus proyectos. Con esta guía se pretende que cada vez sean más los alumnos que tengan acceso a este tipo de tecnologías y que sean capaces de aplicarlas a sus proyectos.

La adquisición de los conocimientos básicos en estas tecnologías, es el primer paso para todos aquellos alumnos que, tras estudiar una ingeniería, se enfrentan a la elaboración de un Trabajo Fin de Grado que haga uso de las capacidades que ofrecen estos dispositivos. En el marco académico actual, no existen asignaturas que puedan dar a los alumnos una introducción básica en esta materia. Internet es actualmente el único recurso de aprendizaje y documentación, donde la gran cantidad de información acerca de diferentes proyectos y dispositivos, puede llegar a resultar abrumadora. Gracias a esta guía didáctica, los alumnos tienen una visión global de todos estos dispositivos desde una perspectiva práctica, puede resultarles muy útil en el desarrollo de sus Trabajos de Fin de Grado.

A lo largo de los últimos años, el grupo de trabajo encargado de elaborar esta guía, ha trabajado directamente en diferentes proyectos relacionados con elementos hardware de código abierto. Esta experiencia, unida a la experiencia docente dirigiendo proyectos fin de grado en titulaciones de ingeniería en la Universidad de Salamanca, nos ha motivado a elaborar una guía que los alumnos puedan tomar como referencia a la hora de abordar sus trabajos. Esta guía es una guía abierta y con posibilidades de ser ampliada y modificada para ir recogiendo las novedades que puedan ir surgiendo ya que se trata de un área que se encuentra en continua evolución.

### 3. Objetivos

Como se ha comentado en la introducción, el **objetivo principal** de esta guía es facilitar a todos los alumnos de ingeniería que se encuentran cursando los últimos años de sus titulaciones, una primera toma de contacto con el mundo del hardware libre. Se busca facilitar a los alumnos, las herramientas básicas para que puedan empezar a desarrollar sus proyectos haciendo uso de las novedades que ofrecen estos sistemas. Se trata de proporcionar una visión de conjunto de todas las posibilidades que ofrecen los sistemas hardware abiertos que sirva como punto de partida a los alumnos. Adicionalmente, se busca facilitar las páginas web y puntos de venta donde poder adquirir estos elementos, así como ejemplos sencillos y funcionales que permitan a los estudiantes realizar sus primeras pruebas con estos dispositivos.

Como **objetivos parciales**, se destacan los siguientes:

- Ofrecer una definición actualizada del hardware libre o de código abierto, así como una definición de conceptos básicos.
- Definir la taxonomía básica del hardware libre
- Enumerar los principales sensores y actuadores existentes.
- Definir los principales protocolos de comunicación inalámbrica
- Introducción a la placa Arduino, como y donde adquirirla y todas las placas derivadas
- Introducción a la placa Raspberry Pi, como y donde adquirirla y todas las placas derivadas
- Generación de ejemplos y prácticas sencillas para aprender los conceptos básicos de la placa Arduino y de la placa Raspberry Pi
- Presentación de proyectos y trabajos realizados con sistemas de hardware de código abierto.

Otro de los objetivos conseguidos dentro de la elaboración de esta guía es dotar a la misma de un carácter didáctico. Se trata de una guía realizada en formato wiki, lo cual facilita la tarea de ser ampliada, modificada y compartida por todos aquellos usuarios que deseen ampliarla, actualizarla o saber más sobre cualquiera de sus partes.

## 4. Desarrollo del proyecto

Con el fin de crear una guía que fuera interactiva para todos aquellos alumnos y usuarios que desearan consultarla, actualizar su contenido o añadir nuevos elementos, se ha creado una sección wiki dentro de un proyecto de GitHub.

Por un lado, una wiki se define como: Sistema de trabajo informático utilizado en los sitios web que permite a los usuarios modificar o crear su contenido de forma rápida y sencilla.

GitHub es la plataforma ideal para exponer repositorios de código abierto y que éstos sean utilizados, corregidos y mejorados por la comunidad. En primer lugar, Git es un sistema de gestión de versiones descentralizado muy potente que facilita el trabajo en grupo sobre un mismo proyecto y que mantiene un histórico de las modificaciones que se han ido realizando sobre el proyecto, permitiendo volver atrás si fuese preciso. GitHub es una plataforma que permite alojar repositorios que empleen Git como gestor de versiones.

Se ha optado por Github debido a que cuenta con distintas opciones como **Issues**, **Pull requests**, **Projects**, **Wiki** y **Insights** entre otras.

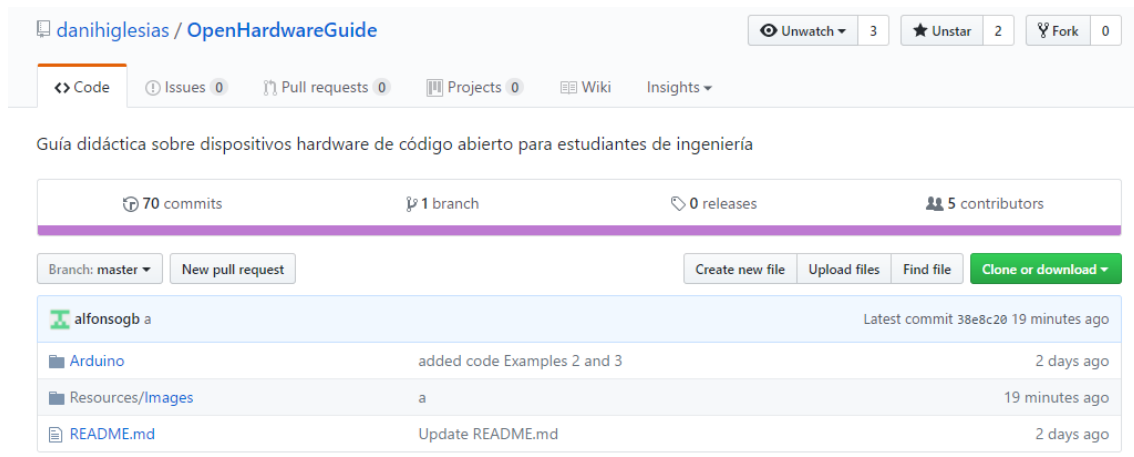


Ilustración 1 Repositorio de GitHub

### Issues

En esta sección cualquier usuario de GitHub puede abrir un *Issue* (problema, cuestión, en inglés). De tal forma que un usuario puede reportar un fallo, pedir una mejora, hacer una pregunta o simplemente tratar un tema concreto. Esta sección hace sencillo llevar un seguimiento de errores, mejoras, etc. Lo que es aún mejor es que en cada Issue se abre un hilo de discusión en el que se puede exponer el problema y los distintos usuarios pueden rebatirlo, ayudar a solucionarlo o lo que fuera preciso. Una vez el Issue queda zanjado es posible cerrarlo para que ya no aparezca en la lista de Issues abiertos, pero siempre será posible volver a acceder a él si fuera necesario.



# Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/2017)

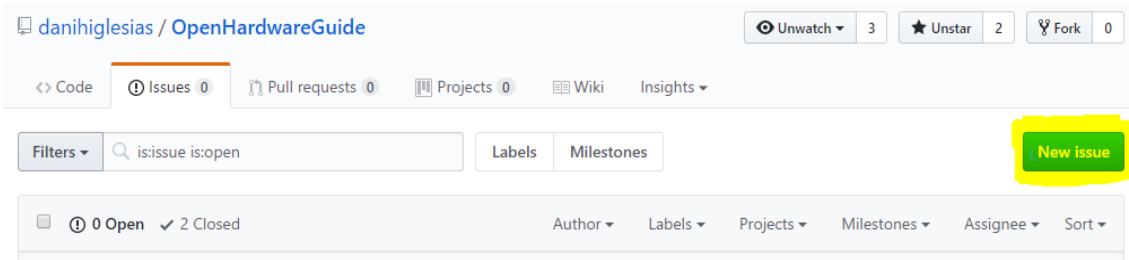


Ilustración 2 Nuevo Issue

En la creación de un nuevo Issue es posible especificar si se trata de un error, una mejora e incluso asignarlo a uno de los colaboradores del proyecto.

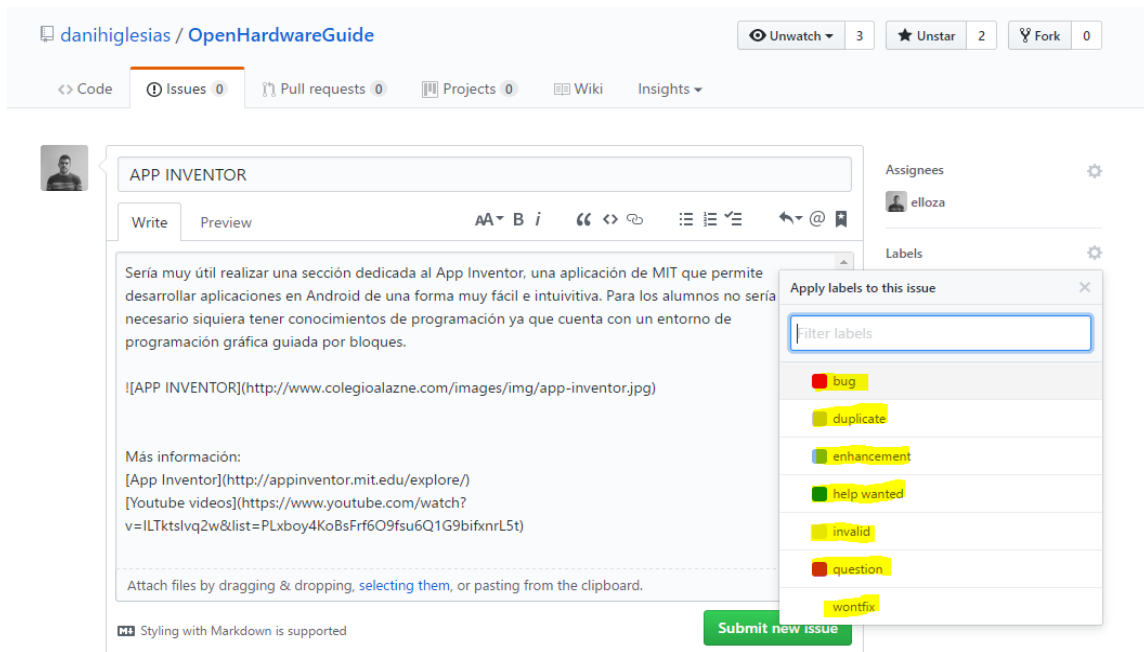


Ilustración 3 Nuevo Issue: título y contenido

Una vez creado aparecerá en la lista de *Issues* abiertos:

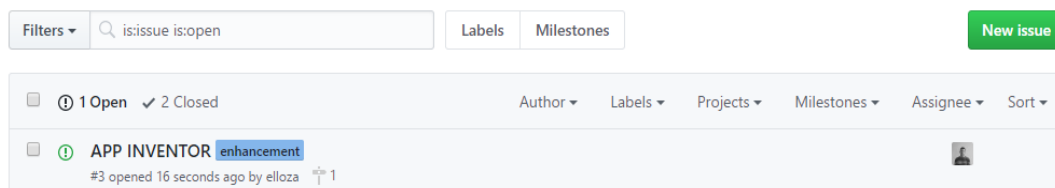


Ilustración 4 Listado de Issues

Una vez en la lista este Issue podrá ser comentado por cualquier usuario de GitHub:

# Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/2017)

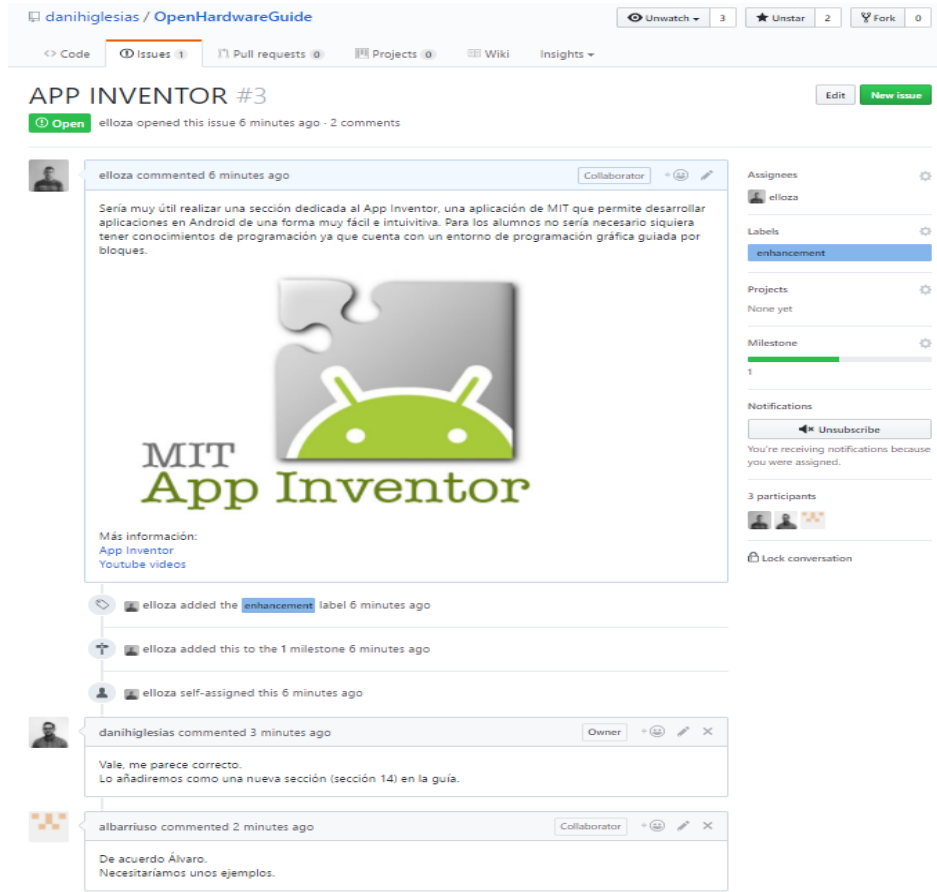


Ilustración 5 Hilo de discusión de un Issue

Esta es una característica muy enriquecedora de GitHub porque a menudo se muestran debates entre miembros de la comunidad para encontrar la mejor solución a un problema planteado. Los colaboradores del proyecto decidirán si cerrarlo o no y darle la oportuna contestación.

## Pull request

Puede darse el caso de que un usuario de GitHub quiera contribuir con nuevo contenido al proyecto como por ejemplo con un ejemplo más de Arduino. Para ello se podrá utilizar esta funcionalidad y serán los colaboradores del repositorio de GitHub quienes aceptarán o denegarán la *pull request* demandada.

Esta funcionalidad es una de las más potentes ya que hace que el proyecto se mantenga vivo y que, por ejemplo, un *Issue* abierto por una persona en particular sea corregido con una *pull request* de otra persona totalmente distinta, quizás en dos partes completamente opuestas del planeta.

Con un ejemplo ejemplo muy sencillo:

En primer lugar, deberemos hacer un *fork* del proyecto de tal forma que tendremos una copia del repositorio en nuestro *GitHub*.

# Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/2017)

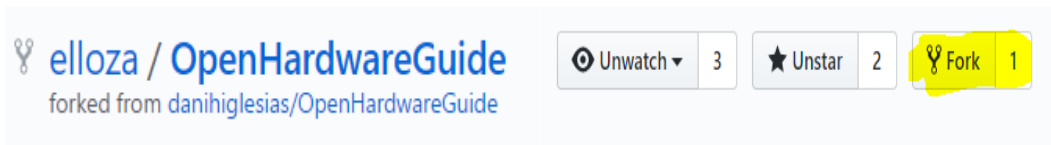


Ilustración 6 Botones de seguimiento estrellas y fork (derecha), título del fork del proyecto (izquierda)

Posteriormente realizaremos los cambios oportunos y realizaremos una *pull request* que pedirá que nuestros cambios se fusionen con la rama que elijamos del proyecto.



Ilustración 7 Botón para crear un nuevo Pull Request

## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

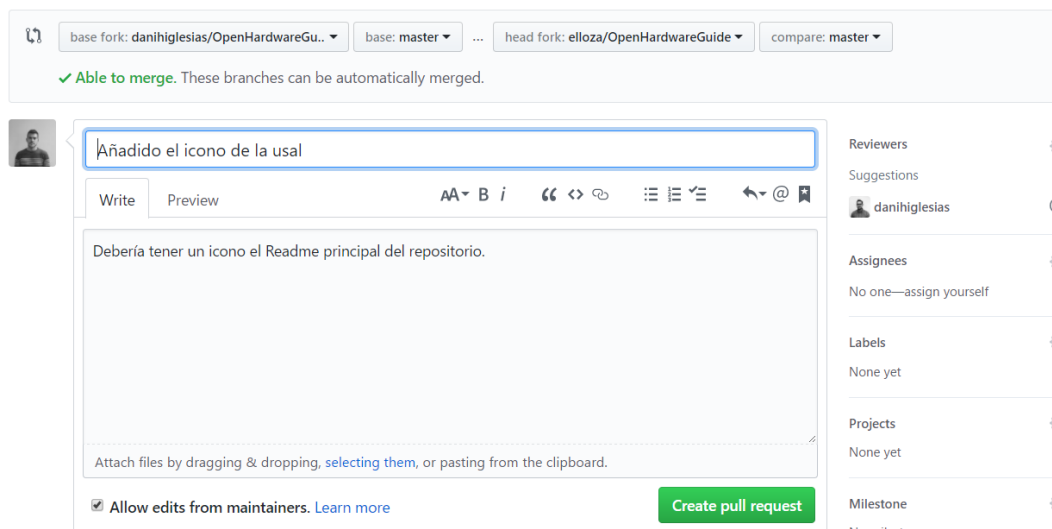


Ilustración 8 Creación de un nuevo pull request

Aparecerá en el listado de *pull request* la nueva *pull request* realizada.

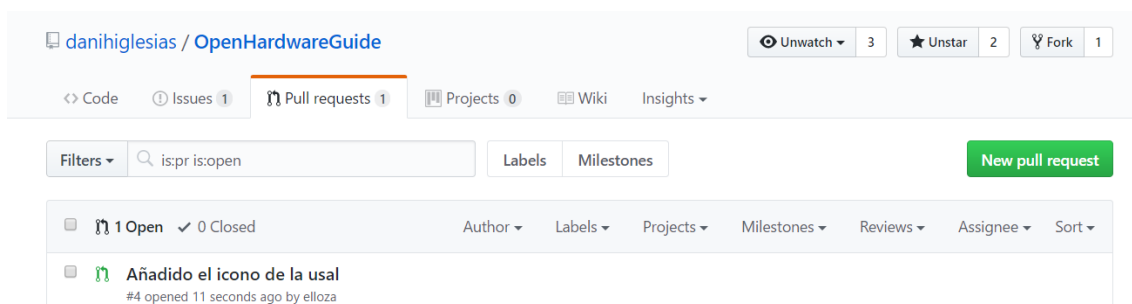
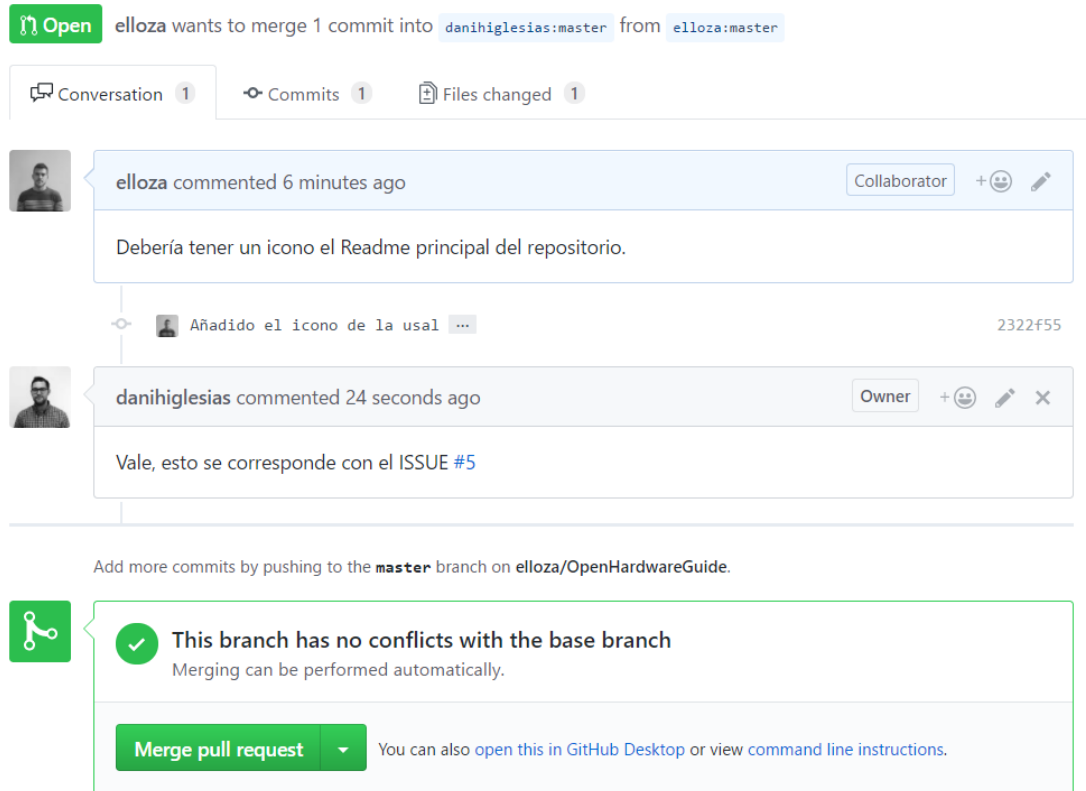


Ilustración 9 Listado de Pull Requests

## Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/2017)

Los colaboradores podrían hacer comentarios y objeciones acerca de la *pull request* realizada. En el mismo hilo de conversación se pueden ver los cambios realizados en cada *commit* adicional de la *pull request*.

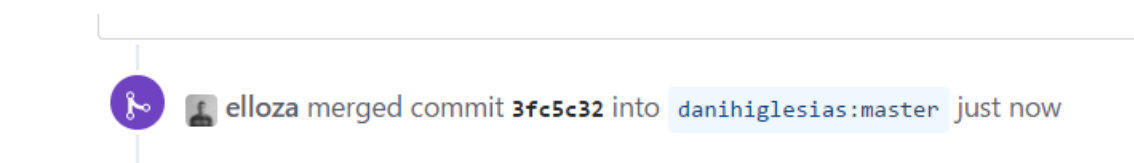
### Añadido el icono de la usal #4



The screenshot shows a GitHub pull request interface. At the top, it says "Open" and "elloza wants to merge 1 commit into danihiglesias:master from elloza:master". Below this, there are tabs for "Conversation 1", "Commits 1", and "Files changed 1". The main content is a discussion thread. The first comment is from "elloza" (Collaborator) posted 6 minutes ago, stating: "Debería tener un icono el Readme principal del repositorio." Below the comment is a commit titled "Añadido el icono de la usal" with a commit hash of "2322f55". The second comment is from "danihiglesias" (Owner) posted 24 seconds ago, replying: "Vale, esto se corresponde con el ISSUE #5". At the bottom of the screenshot, there is a green box indicating that the branch has no conflicts with the base branch and that merging can be performed automatically. A green button labeled "Merge pull request" is visible, along with a link to view command line instructions.

Ilustración 10 Hilo de discusión de un *pull request*

Si los colaboradores aceptan este *pull request* los cambios serán incorporados a la rama *master*.



The screenshot shows a GitHub merge notification. It features a purple merge icon on the left. The text reads: "elloza merged commit 3fc5c32 into danihiglesias:master just now".

Ilustración 11 Merge realizado tras aceptación del *pull request*

Una vez hecho el *merge* (la fusión de los cambios) estos pueden verse en la rama del proyecto. Si existiese algún conflicto con alguno de los archivos, estos deberían ser resueltos antes de realizar el *merge*.

# Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/2017)

<> Code 1 Issues 2 0 Pull requests 0 0 Projects 0 Wiki Insights

Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería

73 commits 1 branch 0 releases 5 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

elloza committed on GitHub Merge pull request #4 from elloza/master Latest commit 3fc5c32 25 seconds ago

Arduino	added code Examples 2 and 3	2 days ago
Resources/Images	a	an hour ago
README.md	Añadido el icono de la usal	10 minutes ago

README.md

**Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería**



**VNiVERSiDAD  
D SALAMANCA**

CAMPUS DE EXCELENCIA INTERNACIONAL

Ilustración 12 Resultado del pull request: portada del repositorio con un icono de la USAL

## Projects

Este apartado añadido recientemente por GitHub permite crear una pizarra tipo Kanban que permita ir añadiendo tareas y que a los desarrolladores principales les sea más fácil utilizar una metodología Kanban Flow.

## Insights

Esta parte permite tener estadísticas de las contribuciones al proyecto y hace posible ver como se trata de un proyecto vivo y con qué frecuencia se incorporan nuevos cambios en él. En esta parte se ven cuando se han realizado *commits* a la rama principal por lo que no necesariamente es cuando se ha realizado el trabajo si no cuando se ha subido el trabajo al repositorio. En esta parte se puede tener un seguimiento de los *commits* realizados por cada usuario, el tráfico del repositorio etc.

# Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/2017)

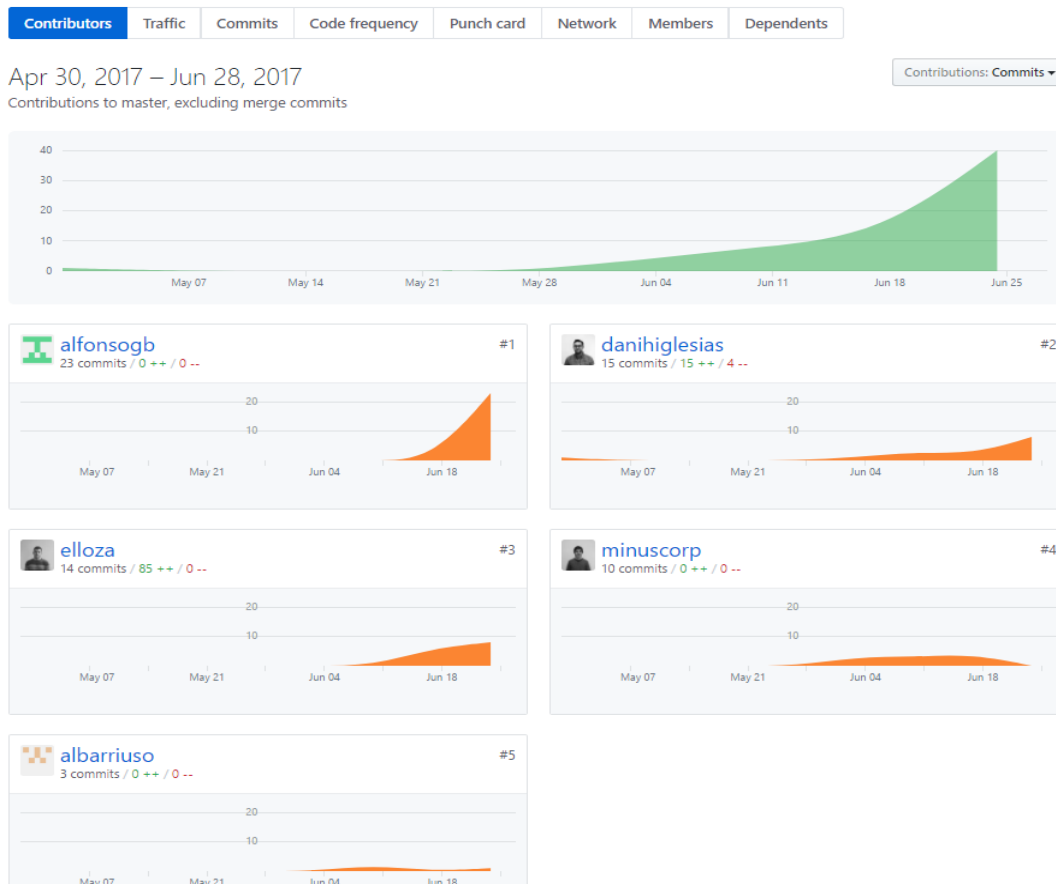


Ilustración 13 Commits realizados por cada colaborador al repositorio

## Estadísticas de los documentos del repositorio:

Referring sites			Popular content		
Site	Views	Unique visitors	Content	Views	Unique visitors
github.com	67	4	Home	139	10
			09. Entornos y lenguajes de programaci...	91	5
			danihiglesias/OpenHardwareGuide: Guía...	87	7
			07. Sensores y actuadores	86	4
			08. Sistemas de comunicación inalámbr...	63	5
			Editing 07. Sensores y actuadores	51	1
			OpenHardwareGuide/Resources/Images...	50	5
			03. Introducción a la placa arduino	49	4
			01. Introducción al hardware de código ...	33	8
			10. Ejemplos de aplicaciones de Hardwa...	31	7

Ilustración 14 Estadísticas de visitantes de cada fichero del repositorio

Por último, pero no menos importante, cada usuario de GitHub puede concederle una estrella al repositorio haciéndolo así uno de sus favoritos e incluso seguir sus cambios con los siguientes botones:

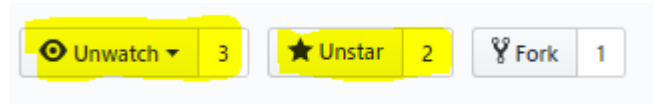


Ilustración 15 Botones de seguimiento, hacer favorito y realizar fork, de izquierda a derecha con sus cantidades.

Esto sirve en la comunidad como unidad de medida de la fama y calidad de un repositorio y puede servir para ver el alcance que puede tener dicho proyecto.

## Estructura de la guía

Se ha estructurado la guía en diferentes unidades con el fin de agrupar los distintos elementos que la componen de una manera sencilla e intuitiva para el alumno. La estructura es la siguiente:

1. **Introducción al hardware de código abierto, definiciones básicas y conceptos teóricos**
2. **Taxonomía del hardware libre**
3. **Introducción a la placa Arduino**
4. **Shields y placas derivadas de Arduino**
5. **Introducción a la placa Raspberry pi**
6. **Shields y placas derivadas de Raspberry pi**
7. **Sensores y actuadores**
8. **Sistemas de comunicación inalámbricas entre dispositivos**
9. **Entornos y lenguajes de programación**
10. **Ejemplos de aplicaciones de Hardware de código abierto en el campo de la ingeniería**
11. **Proyectos relevantes realizados a partir de dispositivos de Hardware de código abierto**
12. **Ejemplos prácticos básicos realizados con la placa Arduino**
13. **Ejemplos prácticos básicos realizados con la placa Raspberry Pi**

## 5. Resultados logrados

La guía didáctica resultante de este proyecto de innovación docente se puede consultar en la dirección:

<https://github.com/danihiglesias/OpenHardwareGuide/wiki>

Como ya se ha mencionado anteriormente, se trata de una wiki creada dentro de un proyecto en GitHub. En la ilustración 16 se observa la página principal de la wiki desarrollada.

# Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/2017)

## Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/114)

Convocatoria de Innovación Docente – Curso 2016-2017



**VNIVERSIDAD  
DSALAMANCA**

CAMPUS OF INTERNATIONAL EXCELLENCE

- Daniel Hernández de la Iglesia (Coordinador)
- Alfonso González Briones
- Álvaro Lozano Murciego
- Alberto López Barriuso
- Jorge Revuelta Herrero

▼ Pages 13

- Home
- 01. Introducción al hardware de código abierto
- 02. Taxonomía del Hardware Libre
- 03. Introducción a la placa arduino
- 04. Shields y placas derivadas de Arduino
- 05. Introducción a la placa Raspberry Pi
- 06. Shields y placas derivadas de Raspberry pi
- 07. Sensores y actuadores
- 08. Sistemas de comunicación inalámbrica entre dispositivos
- 09. Entornos y lenguajes de programación
- 10. Ejemplos de aplicaciones de Hardware de código abierto en el campo de la Ingeniería
- 11. Proyectos de hardware de código abierto
- 12. Ejemplos prácticos básico realizados con la placa Arduino
- 13. Ejemplos prácticos básicos realizados con Raspberry Pi

Ilustración 16 Pantalla Inicial de la guía

En la parte derecha de la ilustración 16, se observa el menú de la guía con las diferentes secciones de las que se compone. En la ilustración 17 se muestra el detalle de la primera sección que describe el concepto de hardware de código abierto.

## Sección 1 – Introducción al hardware de código abierto

En esta primera sección explicaremos que es el hardware de código abierto y cómo surgió. Hablaremos también de los proyectos más importantes dentro del mundo del hardware de código abierto y de conceptos nuevo surgidos en torno al hardware libre como el fenómeno del internet de las cosas (Internet of Things - IoT) o el movimiento maker. Definiremos también otros conceptos básicos que son necesarios para comprender mejor todo lo que rodea al hardware libre y que serán útiles a lo largo de esta guía.

### 1.1 ¿Qué es el hardware de código abierto?

En primer lugar, debemos tener claro el concepto "hardware" y todo lo que engloba su definición. La Real Academia Española de la lengua define el hardware como: El conjunto de los componentes que conforman la parte material (física) de una computadora, a diferencia del software que refiere a los componentes lógicos (intangibles). Sin embargo, el concepto suele ser entendido de manera más amplia y se utiliza para denominar a todos los componentes físicos de una tecnología. En el caso de la informática y los ordenadores, el hardware permite definir no sólo a los componentes físicos internos (disco duro, placa base, microprocesador, circuitos, cables, etc.), sino también a los periféricos (escáner, impresora, teclado...). Podemos decir por lo tanto que el hardware es toda aquella pararte física que hay en una tecnología.



Una vez introducido el concepto hardware, vamos a definir entonces que es el hardware de código abierto. Según se define en la Wikipedia, el hardware libre es de código abierto y define...

▼ Pages 13

- Home
- 01. Introducción al hardware de código abierto
- 02. Taxonomía del Hardware Libre
- 03. Introducción a la placa arduino
- 04. Shields y placas derivadas de Arduino
- 05. Introducción a la placa Raspberry Pi
- 06. Shields y placas derivadas de Raspberry pi
- 07. Sensores y actuadores
- 08. Sistemas de comunicación inalámbrica entre dispositivos
- 09. Entornos y lenguajes de programación
- 10. Ejemplos de aplicaciones de Hardware de código abierto en el campo de la Ingeniería
- 11. Proyectos de hardware de código abierto
- 12. Ejemplos prácticos básico realizados con la placa Arduino
- 13. Ejemplos prácticos básicos realizados con Raspberry Pi

Ilustración 17 Sección de introducción

Se han detallado todas las placas y versiones, así como los diferentes elementos que las componen con la intención de facilitar al alumno una idea clara y sencilla que sirva como punto de partida en sus proyectos. En la ilustración 18 se muestra una captura de la sección dedicada a la placa Arduino dentro de la guía.



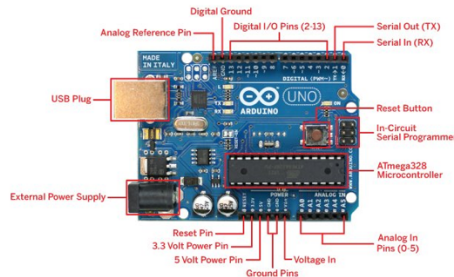
# Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/2017)

Evolución placas Arduino (Fuente: [Blog Arduino](#))

A continuación se enumeran las placas más famosas de Arduino

## Arduino UNO

Una de las versiones más famosas de la placa Arduino es la versión Arduino UNO. Posee un puerto USB a Serie que permite que se utilice el Arduino UNO como cualquier otro dispositivo USB cotidiano. Posee 14 pines digitales de entrada salida de los cuales 6 pueden actuar de salidas PWM. Posee 6 entradas analógicas y posee un microcontrolador ATmega328 a 16MHz con 32KB de memoria Flash.



## Arduino MEGA

Otra versión muy famosa es la conocida como Arduino Mega. Esta versión es más grande y posee muchos más pines analógicos y digitales por lo que es la opción ideal en el caso de necesitar conectar un gran número de sensores, LEDs etc. Aunque esta placa es más grande, es compatible con la mayoría de *Shields* disponibles en el mercado, estos *Shields* los veremos más adelante en una sección dedicada a ellos. La última versión de Arduino Mega utiliza el ATmega2560 que posee 256 KB de memoria Flash.

*Ilustración 18 Sección dedicada a la placa Arduino*

Se facilita también información adicional sobre los elementos descritos que se han considerado útiles para los alumnos. Datos como el precio que puede tener una placa Arduino básica o un sensor de gas y sobre todos, el sitio web o punto de venta donde adquirirlos. En la ilustración 19, se muestra una captura de la guía donde se detalla el precio y la tienda online donde es posible adquirir un sensor detector de movimiento.

## DetECCIÓN DE MOVIMIENTO

### PIR



El sensor de movimiento PIR tiene algunas ventajas sobre otros sensores similares. Tiene un precio bajo y únicamente se activa si algo se mueve. Además, es posible variar la sensibilidad para que se active sólo cuando se detecta un movimiento cerca o aumentar su rango de acción. Es posible utilizar este sensor tanto en entornos interiores como exteriores.

- PIR:
  - Precio: 2,14€
  - Url: [goo.gl/qqkqjx](http://goo.gl/qqkqjx)

*Ilustración 19 Ejemplo de un sensor incluido en la guía*

Se ha incluido también una extensa sección introductoria a la programación. Se trata de una serie de conocimientos básicos que, si bien los alumnos de ingeniería informática conocen gracias a las asignaturas de su titulación, alumnos de otras ingenierías podrían no manejar. Con esta primera toma de contacto con la programación, todos estos alumnos que no conocían previamente esta materia, podrán adquirir los conocimientos básicos necesarios para trabajar con las placas de hardware libre descritas. En la ilustración 20, se recoge una captura de la guía donde se introducen conceptos básicos de programación en lenguaje C.

# Guía didáctica sobre dispositivos hardware de código abierto para estudiantes de ingeniería (ID2016/2017)

## Variables

Una variable es una determinada ubicación de almacenamiento en memoria que está vinculada a un nombre simbólico. La funcionalidad de una variable es la de almacenar un valor, que podrá ir cambiando en tiempo de ejecución (de ahí su nombre, variables), en contraposición con las *constantes*, que también almacenan valores, pero su valor permanecerá inmutable durante la ejecución del programa. Tanto las variables como las constantes están asociadas con un tipo de dato, que determina la naturaleza de los datos que va a poder almacenar.

Antes de utilizar una variable, debe ser declarada previamente. Al realizar la declaración de una variable, estamos reservando un espacio en memoria correspondiente a lo que ocupa el tipo de dato de la variable. A la hora de realizar la declaración, podemos o no asignar un valor inicial a la variable.

La sintaxis para realizar la declaración de una variable es la siguiente:

```
tipo_de_dato nombre_de_variable = valor_inicial;
```

## Tipos de datos

En la siguiente tabla se pueden ver, a modo de resumen, cuáles son los tipos de datos más habituales, su definición y el tamaño que ocupan en memoria.

Tipo	Definición	Tamaño en memoria
void	La palabra reservada void se emplea en declaraciones de funciones para indicar que la función en cuestión no devolverá ningún valor de retorno a la función desde la que se ha invocado.	
boolean	Almacena el valor false (falso) o true (verdadero)	1 byte (8 bits)

Ilustración 20 Sección de introducción al lenguaje C

Con el objetivo de facilitar a los alumnos el aprendizaje en las dos principales placas de hardware libre (Arduino y Raspberry Pi), se han elaborado una serie de ejemplos prácticos bien detallados. El código de estos ejemplos, así como todas las explicaciones y pasos necesarios para reproducirlos, se encuentran disponibles en las últimas unidades de la guía. En la imagen 21, se observa un ejercicio de ejemplo para la placa Arduino, con el código y las explicaciones necesarias para llevarlo a cabo.

Ahora utilizaremos el siguiente *sketch* para leer tanto la entrada del módulo digital como la analógica. Además imprimiremos los valores por el puerto serie para que sea posible verlos por la consola de IDE.

Veamos el código:

```
/*
Aquí se definen algunas constantes para ser utilizadas a lo largo del código.
En este caso los pines analógico y digital utilizados para las entradas del programa
*/
#define ANALOG_SENSOR_PIN A0
#define DIGITAL_SENSOR_PIN 7

int digitalValue; /* Variable para el ultimo valor leído de la entrada digital */
int lightAnalogValue; /* Variable para el último valor analógico leído */

void setup()
{
  // Inicializamos el puerto serie del arduino (en este caso el USB al PC) a 9600 baud
  Serial.begin(9600);
  Serial.println("Lectura de Sensores Ejemplo 2"); //Imprimimos un mensaje de bienvenida
  //Ponemos el pin digital en modo entrada
  pinMode(DIGITAL_SENSOR_PIN, INPUT);
}

void loop(){

digitalValue = digitalRead(DIGITAL_SENSOR_PIN);

//Escribimos el valor de la señal
Serial.println("Valor digital:"+String(digitalValue)); //El valor digital lo pasamos

lightAnalogValue = analogRead(ANALOG_SENSOR_PIN); //Leemos el voltaje del sensor
Serial.print("Valor analógico de (0 a 1023)");

Serial.println(lightAnalogValue,DEC); // Escribimos en el puerto serie
delay(500); //Esperamos
}
```

Una vez analizado el código detenidamente y comprendiendo que hace cada parte

Ilustración 21 Código de ejemplo de una práctica para la placa Arduino

En cuanto a los datos obtenidos de las medidas de evaluación de resultados y mejora del aprendizaje, se han realizado una serie de actividades para evaluar el impacto del material desarrollado en los estudiantes que se encuentran desarrollando su proyecto final.

Tras analizar los trabajos presentados en los últimos 3 años en la titulación de Grado en Ingeniería Informática, quedó constatado que en los últimos meses los trabajos que emplean algún tipo de elemento de hardware libre han ido en aumento. Esto refuerza la tesis inicial de que los alumnos cada vez se interesan más por este tipo de tecnologías a pesar de la falta de docencia en este ámbito.

Se le ha facilitado esta guía a varios alumnos que se encuentran actualmente en las primeras fases de desarrollo de sus trabajos fin de grado en ingeniería. En general las valoraciones de estos alumnos han sido buenas, con una gran aceptación de los contenidos que se han expuesto. Han destacado la sencillez con la que se introducen algunos conceptos que pueden resultar abrumadores y han valorado de manera muy positiva los ejemplos y prácticas facilitadas.

### 6. Conclusiones

Este proyecto de innovación docente surgió con el ánimo de proporcionar nuevos recursos materiales a los alumnos de diferentes disciplinas de ingeniería, de forma que les sirviera como un primer acercamiento con el mundo del Hardware Libre.

Dado el carácter multidisciplinario de los alumnos a los que está orientada la guía didáctica, no se ha presupuesto conocimiento de ningún tipo por su parte. Por tanto, ha sido de vital importancia tener este factor en cuenta a la hora de elaborar la guía, de manera que siempre se mantenga en ésta un espíritu introductorio, brindando a los alumnos los conceptos básicos necesarios acerca de las diferentes tecnologías que se presentan.

Debido a la rápida evolución de la tecnología, los materiales didácticos en esta área de conocimiento suelen tener una vida útil corta. Por este motivo se ha optado por utilizar una plataforma wiki en lugar de elaborar un libro, ya que ésta favorece que el contenido proporcionado en los materiales pueda ser modificado, facilitando de esta forma la vigencia de la guía didáctica.

Consideramos que, tras finalizar la elaboración de los materiales didácticos, se ha conseguido una guía que cumple los objetivos marcados inicialmente, que servirá como punto de partida para todos aquellos alumnos de grado de diferentes disciplinas a la hora de realizar sus primeros proyectos basados en Hardware Libre. Con ello, se pretende impulsar un espíritu de innovación por parte de los alumnos, al facilitarles el acceso a diferentes tecnologías que pueden tener una aplicación directa en sus áreas de conocimiento.