



PROYECTO. SISTEMAS EMPOTRADOS

Contenido

<u>1.</u> <u>Introducción</u>	3
<u>2.</u> <u>El equipo del laboratorio</u>	4
<u>3.</u> <u>Reglas del proyecto.</u>	6
<u>4.</u> <u>Ejecución del proyecto.</u>	8

1. Introducción

Llegados a este punto, el alumno se habrá familiarizado con muchos de los periféricos que están disponibles en cualquier microcontrolador moderno. Es hora de hacer algo práctico con ellos.

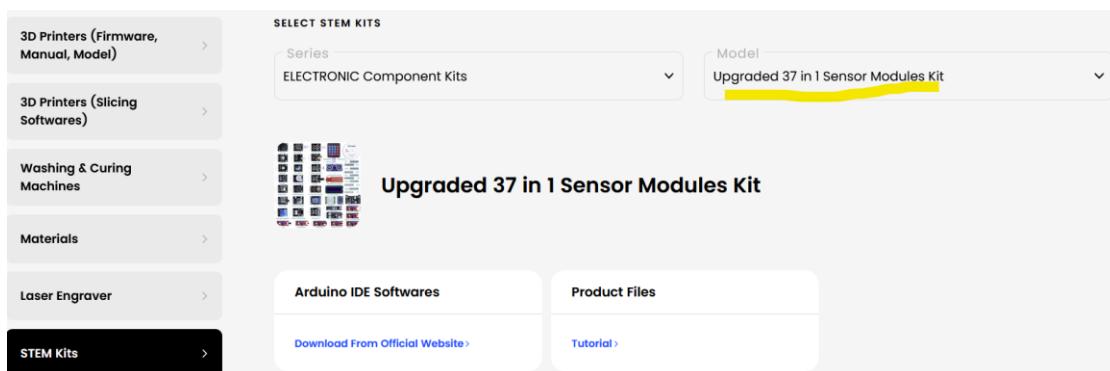
Este proyecto es un ejercicio obligatorio de ejecución libre donde se pondrá en práctica lo aprendido durante el curso sobre el ESP32S3, el kit de desarrollo y los sensores que se disponen en el laboratorio u otros que el alumno pueda proponer.

2. El equipo del laboratorio

En el laboratorio se disponen de 4 kits de componentes de la marca ELEGOO que incluyen 37 sensores y actuadores cada uno.

- Los sensores son de diferente naturaleza, aunque todos proporcionan una salida en forma de tensión/voltaje variable.
- Los actuadores son de tipo digital (ej. LED)
- Alguno de ellos requiere un protocolo particular de comunicación a través de pulsos ya que la señal que proporciona está internamente acondicionada, por ejemplo 'TEMP AND HUMIDITY'.

Información del sensor / actuador y ejemplos básicos de funcionamiento están disponibles en: <https://us.elegoo.com/pages/download> > STEM KITS >



The screenshot shows a website interface for selecting STEM Kits. On the left, there are links for 3D Printers (Firmware, Manual, Model), 3D Printers (Slicing Softwares), Washing & Curing Machines, Materials, Laser Engraver, and STEM Kits (which is highlighted with a black background). In the center, there's a 'SELECT STEM KITS' section with dropdown menus for Series ('ELECTRONIC Component Kits') and Model ('Upgraded 37 in 1 Sensor Modules Kit'). Below this, there's a thumbnail image of the kit components and the text 'Upgraded 37 in 1 Sensor Modules Kit'. At the bottom, there are links for 'Arduino IDE Softwares' (with a 'Download From Official Website' button) and 'Product Files' (with a 'Tutorial' link).





3. Reglas del proyecto.

Para poder echar a andar un proyecto lo primero que se debe tener en cuenta es la imaginación. A partir de los sensores y lo que hemos visto en clase imagínate qué aplicación se puede desarrollar e implementar con un microcontrolador.

Reglas:

La dificultad del proyecto se mide en estrellas. Un proyecto debe tener **al menos 8 estrellas**. Las estrellas se consiguen sumando funcionalidades:

El proyecto debe interactuar con información al usuario por algún medio:

- | | |
|---|------|
| - Visual con led externo | ★ |
| - Visual con led externo bicolor/tricolor | ★★ |
| - Visual con led integrado en la placa y diferentes colores | ★★ |
| - Auditivo | ★★★ |
| - A través del puerto serie (bidireccional) | ★★★ |
| - A través de Bluetooth | ★★★★ |
| - A través de Wifi y un navegador | ★★★★ |

El proyecto debe tomar medidas del mundo exterior

- | | |
|--|------|
| - Interruptor o pulsador externo | ★ |
| - Interruptor o pulsador interno | ★ |
| - ADC y sensor de tipo simple resistivo (ej. joystick) | ★★ |
| - ADC y sensor de tipo LED (ej IR) | ★★★ |
| - ADC y sensor de tipo Tilt | ★★★ |
| - Sensores que requieren protocolo de comunicación | ★★★★ |

Si utiliza temporizadores para sincronizar eventos, medidas, etc

★

Si utiliza mensajes de depuración por el puerto serie

★

Si es capaz de ahorrar energía y entrar-salir de modo *sleep*

★★

Si utiliza la SPDIF para almacenar parámetros

★★★

Otras funcionalidades no vistas en clase (ej, OTA ,...)

★★★★

Nota: Entra en la web de ELEGOO e investiga la naturaleza de los sensores, qué señal proporcionan, cuáles son sus limitaciones, etc. Esto te puede dar una idea para empezar.

Ejemplos de proyectos:

- **Medidor de reacción de usuario.** Una función genera un pulso en un instante de tiempo aleatorio que enciende un led y el usuario tiene que pulsar un botón en menos de 300ms. Si lo consigue, se enciende un led verde, sino un led rojo (alternativamente puede dar una señal acústica). Almacenará la estadística para dar un reporte.
- **Detector de parking.** Con un sensor IR, se detecta la presencia de un coche aparcado o no. La disponibilidad se muestra con un led verde o rojo (de forma complementaria: la información se envía por BLE).
- **Gestión de bombas de agua.** A través de *interface* serie y un menú de selección, el usuario define el funcionamiento de las bombas de agua (LEDs que parpadean de forma intermitente) a diferentes velocidades, dirección de giro, marcha o paro de seguridad.

4. Ejecución del proyecto.

Se pueden utilizar todos los recursos vistos en clase u otros: entradas analógicas y digitales y (salidas analógicas 😊) y digitales, *timers*, memoria no volátil, etc.

Dispones de 4-5 horas aproximadamente en clase para desarrollar el proyecto (hardware y software). Empieza con la parte hardware y vete añadiendo funciones software poco a poco.

Estas 4-5 horas comprenden 2h de la cuarta sesión de laboratorio y la mitad de la quinta sesión. Podrás también libremente adelantar trabajo en casa, sobre todo de la parte software.

En la 5^a sesión, escucharás al resto de compañeros, en tu turno dispondrás de 5-7 minutos¹ para exponerlo al resto de la clase. Para ello, explicarás:

- El motivo del proyecto.
- Elementos del proyecto (sensores, actuadores y funciones básicas)
- Demostración básica de funcionamiento del proyecto.

¹El tiempo se controlará de manera rigurosa para que todos los grupos dispongan de su oportunidad.

La nota tendrá en cuenta todos estos elementos, así como la dificultad del proyecto realizado. Para evaluar el código, tendrás que subir las fuentes de tu proyecto al Campus Virtual.