

## Objetivo de la práctica

Diseñar el sistema de control para una máquina expendedora de un único producto. El funcionamiento en líneas generales es el siguiente.

- Se introduce el importe (entero entre 0€ y 15€) a través de un potenciómetro y un botón (botón A).
- Se debe poder saber en todo momento el importe que va a introducirse, representándolo en binario en LEDs.
- Se efectúa la compra mediante otro botón (botón B).
- Se indica la actuación de la entrega mediante un LED.
- Se indica la falta de existencias o el saldo insuficiente mediante un LED de error (rojo).
- El producto está en varios (tres) compartimentos con existencias independientes. Se deben mantener equilibradas las existencias en los distintos compartimentos seleccionando el más lleno en cada compra.
- Se detecta la recogida del producto mediante un final de carrera (cable) que actúa como detector de la puerta del compartimento.
- Se indica el saldo sobrante por el puerto serie.
- Por defecto, el saldo sobrante se mantiene en la máquina, pero se podrá retirar (eliminar) manteniendo pulsado el botón A durante algunos segundos.
- Se debe poder cambiar el precio del producto (botón A) y añadir existencias (botón B) a través de un modo *técnico* al que se accedera manteniendo ambos botones pulsados durante varios segundos.

## Opcional

- Añadir un indicador de recogida disponible, que indicará además si ha pasado demasiado tiempo sin que se efectúe.
- Calcular la cantidad necesaria de cada tipo de moneda para la devolución.
- Sustituir las funciones `digitalWrite` y `digitalRead` por funciones propias.
- Sustituir la función `analogRead` por una versión propia y no-bloqueante.

## Recomendaciones

- Uso de una máquina de estados.
- Uso de funciones sencillas y reutilizables.
- Parametrización mediante variables, constantes y macros de la mayoría de parámetros del programa.
- Uso de múltiples archivos para estructurar y organizar el programa.
- Uso de arrays y punteros donde sea adecuado, al ser potenciales modificaciones para la defensa.

## Condiciones

- El programa debe ser no bloqueante (como todos los programas de la asignatura).
- Las únicas funciones predefinidas permitidas son:
  - Control de puertos: `digitalWrite`, `digitalRead` y `analogRead`.
  - Control de tiempo: `millis`.
  - Control de comunicaciones: `Serial.begin`, `.read`, `.print/ln` y `.available`.
- El uso de la clase `String` solo está permitido dentro de los métodos `Serial.print/ln`.
- El uso del tipo de dato referencia `&` no está permitido en ningún caso.
  - No debe confundirse con el operador de dirección `&`, que sí está permitido.
- El uso de clases solo está permitido tras consultarlo con el profesor y en las condiciones que se estipulen con cada alumno individualmente.

## Defensa

- La fecha límite para la defensa es el viernes 9 de Mayo de 2025.
- La defensa consistirá en una defensa oral del código, explicando su funcionamiento.
- Durante la defensa se harán preguntas acerca del programa y las soluciones adoptadas por el estudiante.
- Al terminar la explicación se solicitará una modificación que debe realizarse, y defenderse, de forma individual durante la propia sesión.
- La defensa se dará por terminada y suspendida en cualquiera de los siguientes casos:
  - El programa no lleva a cabo las funciones requeridas.
  - Incumplir cualquiera de las condiciones mencionadas anteriormente.
  - Incapacidad para explicar o explicaciones erróneas acerca del funcionamiento de alguna parte del programa.
  - Uso de asistencia para realizar la modificación.
- Cuando la defensa finalice, se subirá el proyecto final al buzón correspondiente de moodle.