**PRÁCTICA 3: Adding Interrupts**

**Finalidad**

El objetivo de esta práctica es la continuación de la práctica 2. El objetivo principal es cambiar el control del pulsador introduciendo interrupciones.

Interfaz de usuario gráfica

Descripción generada automáticamente**Esquema STM32 CubeX**

Tabla

Descripción generada automáticamente

En esta ocasión, hemos configurado el pin del botón como EXTI (External Interrupt Mode). Además, hemos habilitado las excepciones en esa línea (sección NVIC).

No entraremos en más detalle, ya que tanto la configuración de los pines y los componentes del circuito han sido explicados en la práctica anterior.

Imagen que contiene Esquemático

Descripción generada automáticamente**Esquema del circuito**

**Solución propuesta**

En esta ocasión, a diferencia de la práctica anterior, hemos organizado la secuencia del programa mediante un switch para una mayor limpieza y organización. Hemos mantenido el acceso a registros ya que nos resulta más práctico poder modificar distintos pines simultáneamente, a pesar de ser “menos legible”.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamenteEn primer lugar, declaramos todos los posibles estados fuera de la sección main. Esto lo realizamos ya que serán utilizados en distintas funciones como la de callback.

Texto

Descripción generada automáticamente con confianza mediaDeclaramos además las dos funciones que utilizaremos para cumplir la lógica del programa.

Texto

Descripción generada automáticamenteEn la parte principal del programa únicamente se llama a la función de secuencia, a la que pasamos como parámetro el siguiente estado.

Dicha función consiste en lo siguiente:

Texto

Descripción generada automáticamente

El funcionamiento es el siguiente: el estado inicial se corresponde con el semáforo de vehículos verde y el de peatones rojo. Esto se mantendrá hasta que se habilite la interrupción (pulsar el botón).

Cuando se pulsa el botón, accedemos a la función callback:

Texto

Descripción generada automáticamente

La pulsación del botón conlleva a la activación de la interrupción. Aquí, se modifica el siguiente estado al estado amarillo.

Tras esto, se continua con la ejecución del método secuencia, cuyo funcionamiento es idéntico al de la práctica anterior.

En el estado parpadeo se llama a la propia función (explicada en la práctica 2):

Texto

Descripción generada automáticamente

En esta sección del código se declara las distintas interrupciones así como su prioridad.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente