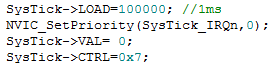
***Practica 4:***

***Interrupciones***

***Hecho por Vladislav Kravchenko***

Durante la realización de la practica 4, he trabajado con interrupciones, para realizar diversas modificaciones a la practica 3. Para su correcto funcionamiento he modificado la practica 3, eliminando varias funciones (void muestrabin\_o\_bcd (); void isp () ;), dentro de ellas la que realizaba la temporización de la practica 3, la función **delay ().** Esta función la he sustituido por la función **SysTick\_Handler ();** la cual he configurado de tal forma que nos interrumpa cada 1ms.

Pero antes de todo esto, lo primero que he realizado es la configuracion de los pines, para definir como entrada los nuevos botones de la placa LPC1768 que se utilizan en esta practica. Despues de esto, con el registro LPC\_PINCON->PINSELX, he configurado las tres interrupciones que se nos piden y las he configurado, con los registros que se presentan en el guion de la practica 4. Estos registros son:

* NVIC\_EnableIRQ(IRQn\_Type IRQn);
* NVIC\_SetPriorityGrouping(uint32\_t PriorityGroup);
* NVIC\_SetPriority(IRQn\_Type IRQn,uint32\_t priority);
* EXTMODE, EXTPOLAR y EXTINT

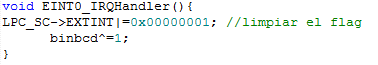
Para la correcta representacion de los numero primos, con el tiempo asignado al SysTick, he implementado en el codigo del **SysTick\_Handler ();** contadores. Con ellos represento la temporizacion prevista de 125ms para los LEDS de la placa LD2 y LD3; y 400ms para el display de 7 segmentos. Toda la repsentacion del numero primo dura 2 segundos, como en la practica anterior.

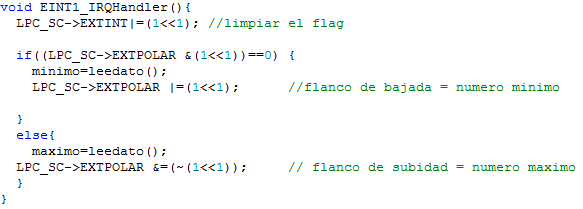
En esta practica 4 he partido de la practica anterior,y he añadido dos funciones adicionales para calcular y representar los numeros primos. Para calcular los numeros primos tengo la función **calculaprimo ()**;. A esta funcion le he pasado el codigo que tenia en la practica 3, en el main. Para realizar el calculo de numeros primos de arriba - abajo o viceversa he necesitado duplicar este codigo, separados en diferentes for, que tienen diferentes limites de funcionamiento, del minimo al maximo, y del maximo al minimo. Dentro de esta función, cuando se obtiene el número primo, el siguiente paso es representar ese número. Para eso he creado una funcion que hace esta tarea, con el nombre de **representa ();.**

En esta función **representa ();** lo primero que se mira es si el número que se va a reprsentar lo va a hacer en binario o BCD, esto es mirar el valor de la variable asociada a la interrupcion que realiza el cambio de representacion entre binario y BCD, la interrupcion EINT1.

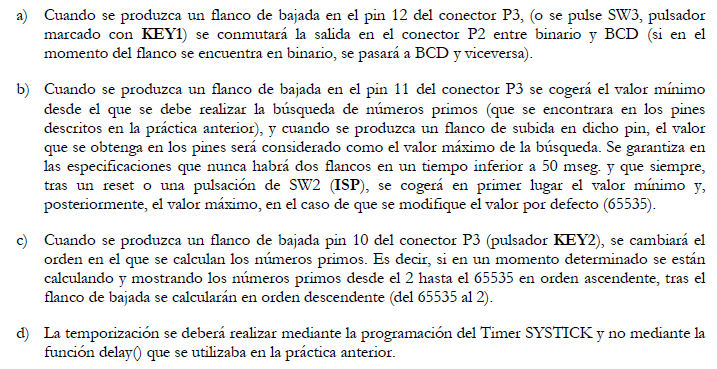
En esta practica se nos pide añadir la funcionalidad a los botones de la Placa MiniDK2. En ella tenemos los botones ISP, KEY1, KEY2 y RESET.

* El pulsador ISP, asociado a la ISR EINT0, debe realizar la funcion de cambiar entre la representacion en binario y BCD. Para eso he escrito el siguiente codigo:



* El pulsador KEY1, asociado a la ISR EINT1, debe realizar la funcion de coger el limite inferior cuando se produzca un flanco de bajada y el limite superior cuando se produzca un flanco de bajada. Para ello he escrito el siguiente codigo:
* El pulsador KEY2, asociado a la ISR EINT2, debe realizar la funcion de cambiar el sentido de la representacion de los numeros primos, en orden ascendente o descendente. Para ello he escrito el siguiente codigo:



Durante la realizacion de la practica he tenido un problema el cual consiste en que habia descargado de la Blackboard la practica 4, y trabajaba con ella desde el escritorio de mi ordenador. Y en los ultimos dias antes de entregar la practica me di cuenta de que mi practica 4, no era la misma que la practica 4 subida en la BlackBoard. Le adjunto un pantallazo de la practica 4 erronea:

La cual se distingue de la practica 4 subida en la BlackBoard.