Indice

## Resumen Resolución

## Programa Principal

## Interrupciones

## LCD

## Transmisión Recepción

## Proteus

## Resumen Resolución

El programa lo he planteado con 2 interrupciones externas.

Nada más inicializarse (InicioReset) se permite la interrupción externa 0 a flanco. El LCD se inicia y la bombilla se enciende. Se quedara en un bucle, del cual saldrá tras ocurrir la interrupción EX0.

El pulsador se corresponde a la EX0, la cual nos encenderá contador de pulsos y permitirá la activación de la interrupción EX1, y desactivara la posibilidad de activarse la Interrupción EX1 Nos fijamos que es la primera vez que pulsamos la EX0. Esta interrupción también nos modificara un registro6 para que salga del bucle y avance a las líneas que apagan el LCD y dejan el LED parpadeando.

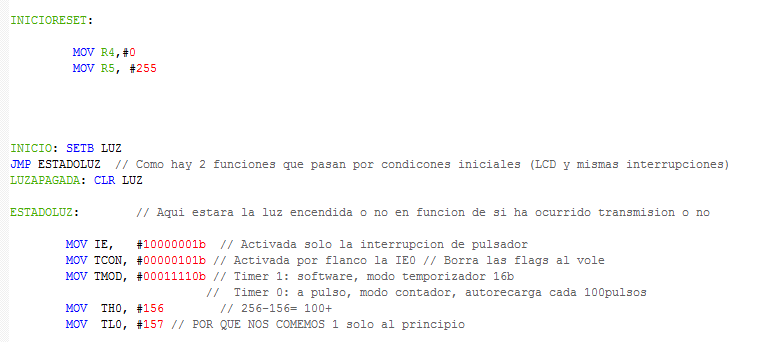
Y conectamos nuestra señal de pulsos a la interrupción externa 1, para que nos active el temporizador T1 nada más llegue el primer el pulso tras ocurrir la pulsación 1. Se desactiva EX1.

T1, no presenta interrupción, cuenta de manera ascendente hasta 65000 m, el tiempo de pulso.

T0, que programamos como contador en las primeras líneas tras inicio, se activara la flag cuando hayan llegado 100 flancos ascendentes de pulsos, en ese momento, guardara TH1 y TL1 en la posición 7000h de memoria RAM y sucesivos, y se realizara 256 veces. Tras realizarse tantas, volverá al inicio modificando el registro6, para que se quede en el bucle inicial, con la diferencia de que un registro 4, nos dejara la luz apagada pero el LCD se encenderá

Este registro 4, además, tras volver a realizarse la pulsación, saltara a la secuencia de transmisión.

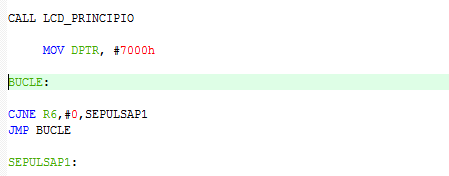
## Programa Principal



Aquí vemos un SETB Luz y un CLR Luz, dependiendo de en que momento del programa estemos, acudiremos a uno u a otro.

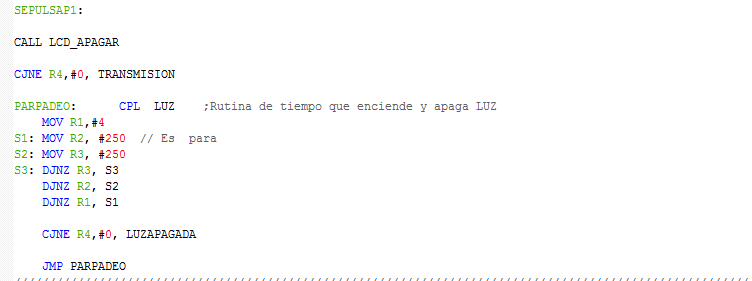
Activamos solo IE0 y preparamos 156 de autorrecarga en el contador ( como el primer flanco no se cuenta, la primera solo ha de contar 99)

Se llama a la rutina de iniciar el LCD



Como el LCD esta mapeado en la posición 4000 de memoria, nos carga el dptr con esa posición, nuestro programa guarda los datos en la posición 7000h.

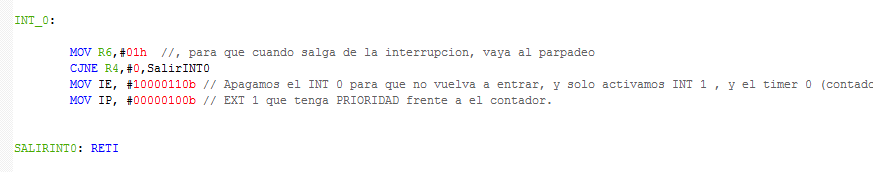
Entra en un bucle del que no saldrá hasta que ocurra la interrupción que modificiara el registro 6.



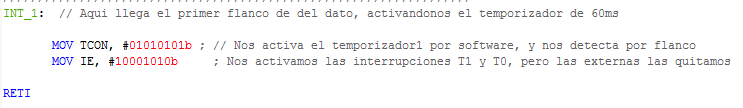
Tras pulsarse, se apaga el LCD y mira que no este encendido R4, que se activa al guardar en memoria los 256 datos. En el bucle la luz se complementa cada 0,5 segundos con esa rutina.

Cuando R4 se active, ira a la rutina luz apagada que tenemos en las primeras líneas de código. Donde se encenderá el LCD, esta vez con la luz apagada y se quedara en nuestro primer bucle esperando a que se vuelva a pulsar el pulsador 1. Esta segunda vez, R4 valdra 1 e iremos a la subrutina de transmisión.

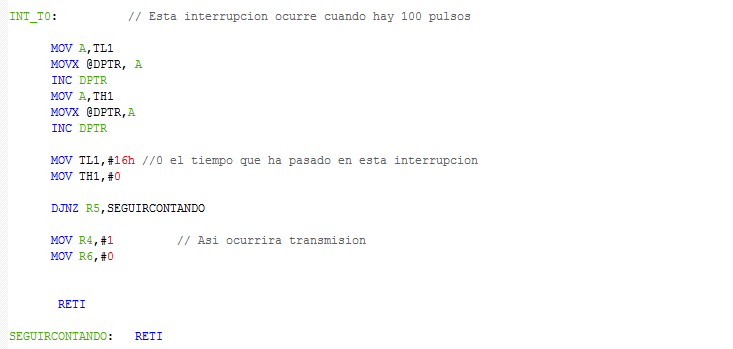
## Interrupciones



Esta interrupción activa la interrupción 1 y prepara el contador de flancos, además de desactivarse a si misma. Cuando tenga que ocurrir por segunda vez, no activará esta interrupción, ira directamenta a salirint0. (En la nueva versión , salirinto 0 tiene un Reti encima, y en la línea desactiva la interrupción IE)



Solo se activa tras P1 y el primer flanco del pulso. Incia el Temporizador 1 y se desactiva a si misma.

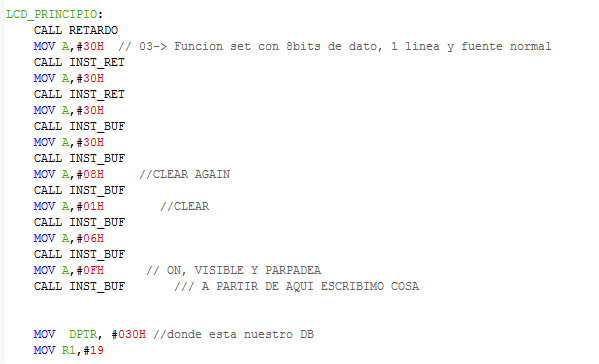


Esta interrupción ocurre al haber 100 pulsos , manda TL1 y TH1 a la posición del DPTR, y al acabar reinicia\* el temporizador. Si R5, que hemos cargado con 255 se vuelve 0, R4 se encenderá, el registro que indica que ha ocurrido el guardado de datos. Además R6 volverá a ser cero para quedarse en el bucle con el LCD encendido.

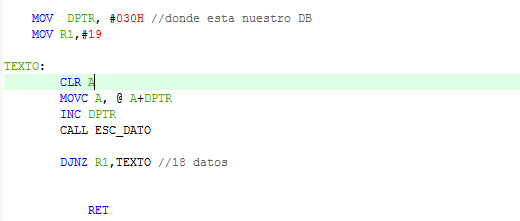
\*No lo reincia, lo deja en 16 que es el tiempo que no ha contado mientras se realizaba esta instruccion

# LCD

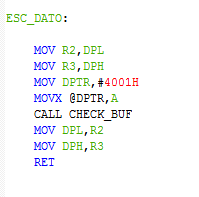
16x2, 1 Linea , sin que avance el cursor y 8 bits de dato (30H). Realizamos la secuencia de inicialización del LCD tras llamarlo.



Cargamos en el DPTR el valor 30H , que es donde hemos escrito nuestro DB, y en R1 el numero de letras que tiene DB.

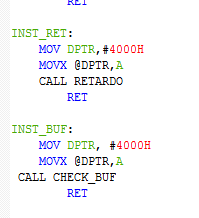


Tras ponerse con el cursor, tenemos que mandarle dato, cargara el valor guardado en 30H en A, y moverá el dato A la posición de memoria 4001H, que tal y como tenemos mapeado el LCD , se corresponde a un 01 en el RS RW del LCD, o escribir dato.

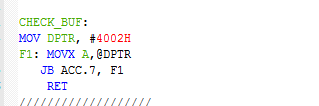


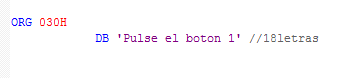
R2 y R3 guardan en nuestro caso #30H para que no se borre al mover el 4001 al dptr

Estas son las instrucción de inicialización

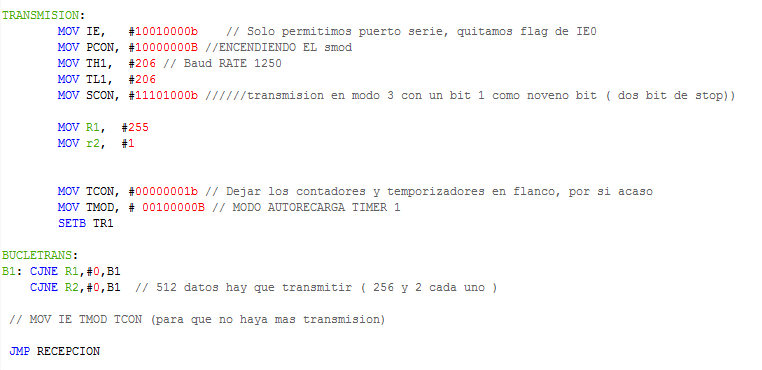


Al cargar el 30H le decimos como ha de inicializarse el LCD. Por como esta diseñado, hay que realizar esto 2 veces con un retardo de al menos 1,6ms cada una, y después otras dos veces más esta vez comprobando que la flag busy del controlador no este encendida antes de enviar el siguiente dato/instrucción.





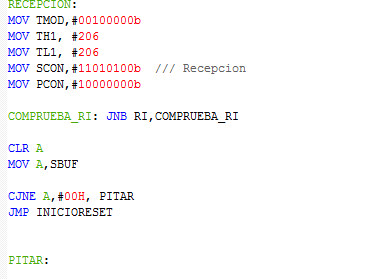
## Transmisión - Recepción



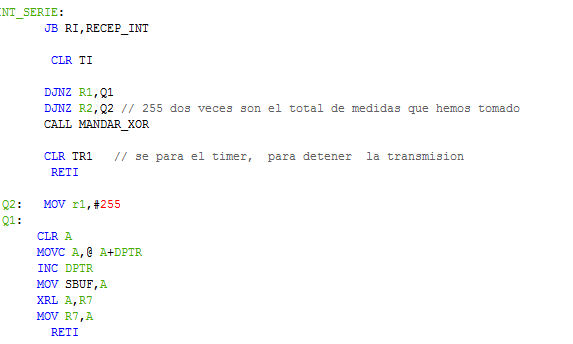
Preparamos nuestro puerto serie a 1250 baudios, modo con 2 bit de stop. como nos dice el enunciado

Se quedara en el bucle hasta que se hayan transmitido los 512 datos ( por eso usamos R1 y R2)

Al acabar el bucle ira a recepción.



En recepción, comprueba que A sea distinto de cero y salta nuestra interrupción pitar, o vuelve al inicioRESET ( con los valores R4,R6… como cero)



La interrupción correspondiente al puerto serie, cuando carga los 512 valores en el sbuf, ha de cargar un valor CHECK SUM, una XOR de cada uno de los datos, con el valor guardado ya pasado por la XOR. ( se guarda en R7)

