

CCLK=100MHz

ADC

Configuramos el divisor de periférico a 8:

$$\frac{CCLK}{divP} = \frac{100 \text{ MHz}}{8} = 12,5 \text{ MHz}$$

Al utilizar el modo controlado necesito 65 ciclos de reloj para establecer la muestra:

$$\frac{12,5 \text{ MHz}}{65} = 195312 \text{ Hz}$$

Siendo la frecuencia de trabajo máxima configurable.

Luego se configura el timer 0 para que de un tiempo de muestra de 1 segundo.

Teclado Matricial

Para el antirebote y el tiempo del buzzer, se utilizo una función delay que fue implementada mediante un for la cual no es la implementación mas elegante pero consideramos que era la forma mas efectiva de implementar una espera activa sin tener que encender un periférico mas o afectar la secuenciación de otros en uso.

NumOp: Numero de operaciones de 1 ciclo for

TimOp: Tiempo que tarda en ejecutarse 1 ciclo for

TimSel: El tiempo de espera activa elegida

OpSel: Numero de operaciones necesarias para obtener el tiempo de espera deseado

NumOp: Numero de operaciones de 1 ciclo for newline

TimOp: Tiempo que tarda en ejecutarse 1 ciclo for newline

TimSel: El tiempo de espera activa elegida newline

OpSel: Numero de operaciones necesarias para obtener el tiempo de espera deseado

Se estimo que un ciclo for ejecuta un NumOp (Numero de operaciones) de 11, mediante una regla de 3:

$$1 \rightarrow \frac{1}{CCLK}$$

$$NumOp \rightarrow TimOp$$

$$TimOp = \frac{NumOp}{CCLK} = \frac{11}{100 \text{ MHz}} = 1.1 * 10^{-7} \text{ s}$$

Obtenemos 1.1e-7 S por ciclo for, entonces con la siguiente relación, podemos ver cuantas operaciones serian un tiempo determinado:

$$TimOp \rightarrow 1$$

$$TimSel \rightarrow OpSel$$

$$OpSel = \frac{TimSel}{TimOp} = \frac{TimSel}{\frac{NumOp}{CCLK}} = \frac{TimSel}{NumOp} * CCLK = \frac{TimSel}{\frac{NumOp}{CCLK}} = \frac{CCLK}{NumOp} * TimSel = \frac{100\text{ Mhz}}{11} * TimSel$$

Así obtenemos la relación que se usa en la función del programa, el cual según testeo tarda el tiempo que debería y cumple perfectamente.

Timer 1

El Timer 1 es el encargado de contar 1 minuto, por lo tanto, vamos a colocar un preescaler para que la cuenta aumente cada 1 segundo. Por lo tanto usando el driver con la función UsVal, colocamos un valor de 1000000, podemos hacer el calculo inverso y ver el valor que se coloca en el preescaler:

Td: Tiempo Deseado

Pr: Prescaler

DivP: divisor de perifericos (Default)

$$Td = \frac{(Pr+1)*DivP}{CCLK} \rightarrow \frac{Td*CCLK}{DivP} = Pr+1 \rightarrow \frac{Td*CCLK}{DivP} - 1 = Pr = \frac{1000000\text{ uS} * 100\text{ MHz}}{4} - 1 = 24999999$$

Entonces, el Match al tener un valor de 60, el contador va a aumentar cada 1 segundo, entonces el match se va a dar al minuto de iniciado el timer.

Timer 0

El timer 0 es el encargado de generar un tiempo de muestreo de 1s, por lo tanto, necesito que la frecuencia de toggleo sea el doble (O que el tiempo a la mitad). Por lo tanto, el Match se debe dar cada 500mS. Con esto, se configura el preescaler para que el timer incremente cada 100mS con la función UsVal, podemos hacer el calculo inverso y ver el valor que se coloca en el preescaler:

$$Td = \frac{(Pr+1)*DivP}{CCLK} \rightarrow \frac{Td*CCLK}{DivP} = Pr+1 \rightarrow \frac{Td*CCLK}{DivP} - 1 = Pr = \frac{1000\text{ uS} * 100\text{ MHz}}{4} - 1 = 24999$$

Entonces, el Match al tener un valor de 5, el contador va a aumentar cada 100 mS, entonces el match se va a dar a medio segundo de iniciado el timer.

Systick

El Systick se usa para hacer parpadear el led en el modo 3. Para esto se utilizan ciclos de 150mS de encendidos y de apagados. En concreto, el Systick es un contador de 2^{24} , lo cual con el CCLK por defecto da:

SystickMax: Cuenta maximadel Systick

$$\frac{SystickMax}{CCLK} = \frac{2^{24} - 1}{100 \text{ MHz}} = 168 \text{ mS}$$

Por lo tanto con los 150mS estamos dentro de los tiempos posibles.

Los ciclos de prendidos y de apagados es de 5 interrupciones con el led encendido y 5 con el led apagado, por lo tanto, el led enciende y se apaga cada:

$$5 * 150 \text{ mS} = 750 \text{ mS}$$