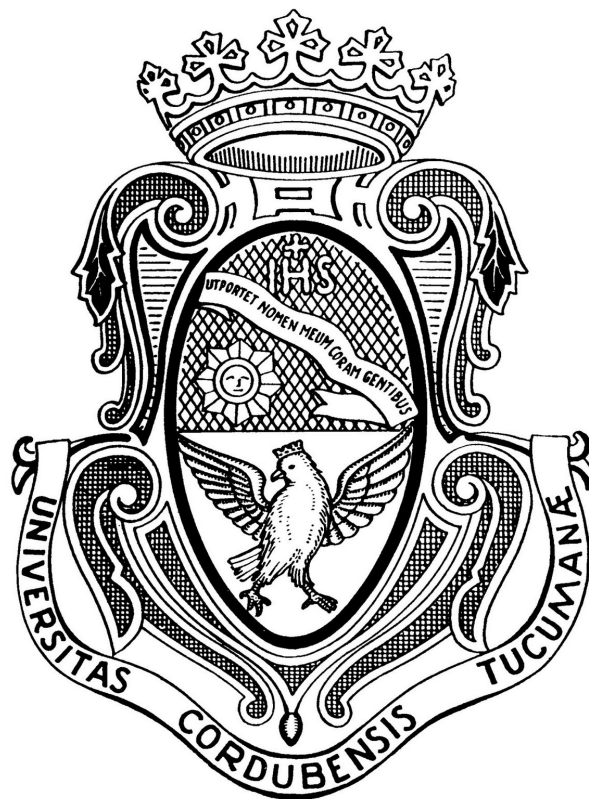


Universidad Nacional de Córdoba
Facultad de Ciencias Exatas, Físicas y Naturales



Trabajo Práctico Final
Electrónica Digital II
Docente: Ing. Martín Del Barco

Losano Quintana, Juan Cruz
Piñero, Tomás Santiago
Ingeniería en Computación
Año 2019

Índice

Índice	1
1. Introducción	2
Enunciado	2
2. Desarrollo	2
Cálculos realizados	2
Resistencias para los segmentos	2
Resistencias para la multiplexación	2
3. Implementación	3
Diagramas de flujo	3
Programa principal	3
Interrupciones	3
Interrupción por RB0	3
Interrupción por <i>Timer0</i>	4
Interrupción por <i>Timer1</i>	4
Interrupción por <i>ADC</i>	5
4. Esquema del circuito	5

1. Introducción

Este informe trata sobre el trabajo práctico final de la materia Electrónica Digital II. El tema es a elección de los estudiantes, por lo que se eligió realizar un sensor de temperatura para propósitos generales.

Enunciado

El sistema a realizar consiste en sensar la temperatura ambiente. La medición comienza cuando se presiona un botón. Una vez presionado se muestra el valor sensado en 4 (cuatro) *displays* de siete segmentos con el formato “XX°C” y se realiza un *log* en la computadora cada cinco segundos.

Para su realización se utilizaron los siguientes materiales:

- Microcontrolador PIC16F887,
- Cristal de 4 MHz,
- *Bridge USB to UART* CP2102,
- Sensor de temperatura LM35,
- Cuatro *displays* de siete segmentos cátodo común,
- Resistencias de 1 k Ω , 330 Ω y 220 Ω ,
- Capacitor de 22 pF,
- Un pulsador.

2. Desarrollo

Cálculos realizados

Resistencias para los segmentos

Los *displays* son rojos, usan X voltios y del pic salen X mA, entonces la formula para la resistencia es:

$$algo = 0$$

$$R = \frac{algo}{algomas}$$

$$R = tantosOhmios \tag{1}$$

Resistencias para la multiplexación

Al tratarse de *displays* de cátodo común, se necesitan transistores NPN para su multiplexación. La fórmula para el tiempo de encendido de cada *display* es la siguiente:

$$Tiempo = \frac{50ms}{N} \quad (2)$$

Siendo N la cantidad de *displays* a utilizar.

3. Implementación

El PIC tiene como frecuencia de reloj un cristal de 4 MHz, por lo que el ciclo de instrucción se realiza con una frecuencia de 1 MHz debido a que cada 4 ciclos del reloj se realiza una instrucción. Esto se debe a que para ejecutar la instrucción indicada, el PIC debe ejecutar cuatro acciones:

1. Buscar la instrucción en la memoria principal.
2. Decodificar la instrucción.
3. Ejecutar la instrucción.
4. Almacenar los resultados.

Esto es importante para el cálculo de la subrutina de retardo, ya que depende de la frecuencia de reloj que se utilice.

Diagramas de flujo

En esta sección se muestran los diagramas de flujo del programa principal y las subrutinas que utiliza.

Programa principal

Primero se realiza la configuración de los puertos A y B como salida y entrada digitales, respectivamente.

Una vez configurados los puertos se toman los datos del puerto de entrada y a esa lectura se la invierte, ya que cuando los Dip Switch estén bajos las entradas estarán con un valor de uno lógico debido a la presencia de las resistencias pull-up.

Consecuentemente el programa almacena los últimos cuatro bits en la variable *numero1* y se lo suma a los primeros cuatro bits leídos, mostrando el resultado en los LEDs de salida. Si el resultado es de cinco bits, el LED del *digit carry* parpadeará y no se podrá realizar otra suma hasta que se resetee el PIC.

Interrupciones

Interrupción por RB0 Esta interrupcion hace esto

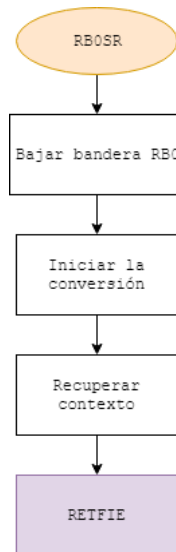


Figura 1: Diagrama de flujo de la interrupción por RB0.

Interrupción por *Timer0* Esta interrupcion hace esto

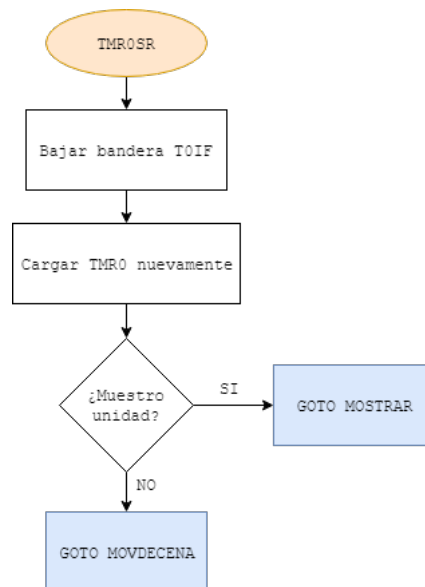


Figura 2: Diagrama de flujo de la interrupción por *Timer0*.

Interrupción por *Timer1* Esta interrupcion hace esto

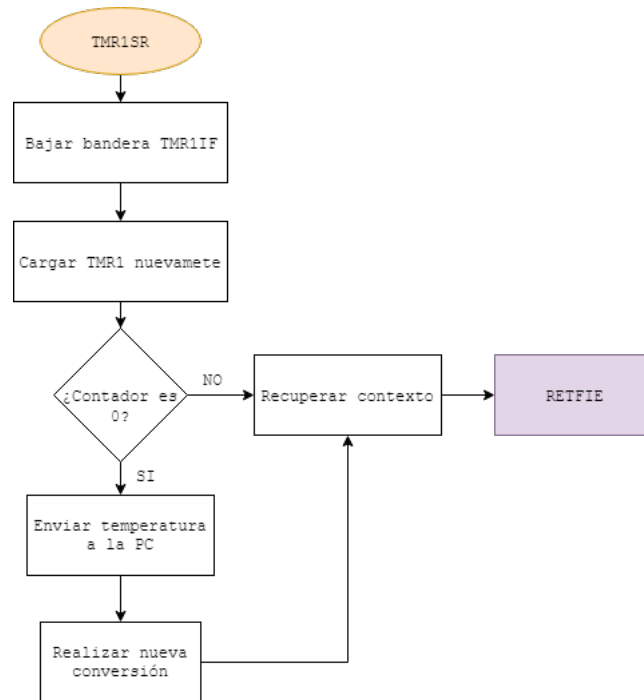


Figura 3: Diagrama de flujo de la interrupción por *Timer1*.

Interrupción por *ADC* Esta interrupcion hace esto

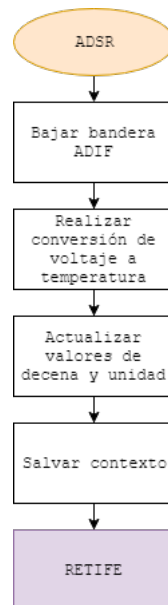


Figura 4: Diagrama de flujo de la interrupción por *ADC*.

4. Esquema del circuito