

Universidad Autónoma de Baja California



Microcontroladores y microprocesadores

Practica 7. Uso de puertos y retardos mediante Software

Alumno: Caudillo Sanchez Diego

Matricula: 1249199

Docente: Jesus Adan Garcia Lopez

Fecha de entrega: 16 de abril de 2020

Teoria

Retardo por software (calculos)

Los retardos por software puede ser creados con ciclos (for, while). En cualquiera de los casos estos dependen de

- **Cantidad de ciclos a realizar:** viendo las hojas de especificaciones del ensamblador que se encuentre utilizando, se puede saber cuantos ciclos se tarda una instrucción en ejecutarse. Este factor lo decide el programador, y tiene total control de él.
- **Código que genera el compilador:** debido a que una vez que se está compilando el programa, el mismo compilador optimiza el código, dependiendo de la configuración de este. En estos casos no es posible controlar este factor. Una manera de evitar esto, es escribiendo el programa en ensamblador, así el compilador pasa el código tal como está.
- **La velocidad del procesador:** este punto es de suma importancia, debido a que es necesario conocer la velocidad en la que el procesador ejecuta la instrucciones. A partir de aquí podemos hacer los cálculos necesarios junto al primer punto para poder ejecutar los ciclos necesarios para el Delay deseado.

Para saber cuánto tiempo consume la ejecución de un código, es necesario hacer un análisis de este último, para determinar cuántos ciclos de reloj consume en toda la ejecución.

Tomando en cuenta que un procesador tiene una velocidad de 16 MHz, analizaremos el siguiente ejemplo.

Nota: 16MHz significa que realiza 16 millones de ciclos por segundo, por tanto $62.5\mu S$ debido a que $\frac{1}{16000000} = 62.5\mu S$.

J1:	LDI	A, 190	→ 1 = 1
	LDI	B, 122	→ [190](1) = 190
J2:	LDI	C, 11	→ [190][122](1) = 23,180
J3:	NOP		→ [190][122][11](1) = 254,980
	NOP		→ [190][122][11](1) = 254,980
	NOP		→ [190][122][11](1) = 254,980
	DEC	C	→ [190][122][11](1) = 254,980
	BRNE	J3	→ [190][122][11](2) - [190][122](1) = 509,960 - 23,180
	DEC	H	→ [190][122](1) = 23,180
	BRNE	J2	→ [190][122](2) - [190](1) = 46,360 - 190
	DEC	A	→ [190](1) = 190
	BRNE	J1	→ [190](2) - 1 = 380 - 1
	RET		→ [1](5) = 5

Nota: Como el código tiene una instrucción RET sabemos que se manda a llamar con una instrucción CALL, por lo que hay que considerar la cantidad de ciclos que consume la instrucción (5).

Resultado = 5+1+190+23,180+4(254,980)+509,960-23,180+23,180+46,360-190+ 190+380-1+5

Resultado = 1,600,000 **instrucciones**

Comentarios

En un principio, tuve dudas con el cálculo de las instrucciones cuando estaban dentro de un ciclo. Debido a que no sabía el alcance que tenía cada pedazo de código cuando este se encontraba dentro de un ciclo. Otro problema al cual me estaba enfrentando constantemente era el de encontrar numeros que al dividir dieran números enteros, normalmente la última variable era la que más fallaba en esos aspectos.

Conclusiones

Con la finalización de esta práctica se logró aprender a cómo analizar códigos para determinar el tiempo que tardan en ejecutarse, y hacer los cálculos pertinentes con las instrucciones que ejecutan los códigos.

Bibliografía

Presentación de Retardos mediante Software -Leocundo Aguilar

Muhammad A., Sarmad N., & Sepehr, N. (2011). *The AVR microcontroller and embedded systems: using Assembly and C*. Pearson