|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Carrera:** | **Desarrollo de Software** | | | | **Academia:** | **Sistemas Digitales** | **Plantel:** | **Colomos** |
| **Materia:** | **Arquitectura y Organización de Computadoras** | | | | **Clave:** | **18MPBDS0514** | **Revisión:** | **A** |
| **Unidad:** | **2. Elementos de un Microprocesador y su arquitectura interna.** | | | | **Tema:** | **La programación.** | | |
| **No. de Práctica:** | **8** | **Nombre de la práctica:** | | **Potencia de un número** | | | | |
| **Profesor:** | **Antonio Lozano González** | | | | | | | |
| **Alumno:** | **Emmanuel Buenrostro Briseño** | | | | | | **Registro:** | **22300891** |
| **Alumno:** |  | | | | | | **Registro:** |  |
| **Semestre:** | **5** | **Grupo:** | **I1** | | **Período:** | **Ago-Dic 2024** | **Fecha:** | **25/10/24** |

**1. Objetivo.** **Hacer una potencia de un número con el sistema, para seguir aprendiendo como  
introducir datos al sistema en ensamblador.**

**2. Material, Equipo y/o Herramientas**

|  |  |
| --- | --- |
| **Cantidad** | **Material, Equipo y/o Herramientas** |
| **1** | **Microcontrolador** |
| **1** | **Memoria** |
| **1** | **Latch** |
| **1** | **Buffer** |
| **1** | **Minidip** |
| **8** | **Leds** |

**3. Desarrollo de la práctica**

**a) Condiciones de la práctica:**

Introducir  un numero comprendido entre uno y 255 por el puerto de entrada del  
sistema, este numero sera elevado al cubo. Mostrar el resultado por el puerto de salida.

**b) Algoritmo o Diagrama de Flujo.**

**En este codigo lo que hacemos es primero multiplicar el numero por si mismo para tenerlo al cuadrado, despues multiplicar ambos valores de 8 bits por el valor que queremos elevar al cubo (porque al elevarlo al cuadrado te quedan 8 bits), y los valores de “en medio” (es decir al multiplicar el menos significativo tomamos el mas significativo y viceversa) y los sumamos y si hay carry se lo sumamos al mas significativo que sale del multiplicar el mas significativo.**

**c) Código LST.**

**Cubo PAGE 1**

**0000 E5B0 1 cubo: mov A,P3**

**0002 85B0F0 2 mov B,P3**

**0005 A4 3 mul AB**

**0006 A9F0 4 mov R1,B**

**0008 85B0F0 5 mov B,P3**

**000B A4 6 mul AB**

**000C F8 7 mov R0,A**

**000D AAF0 8 mov R2,B**

**000F E9 9 mov A,R1**

**0010 85B0F0 10 mov B,P3**

**0013 A4 11 mul AB**

**0014 ABF0 12 mov R3,B**

**0016 8AF0 13 mov B,R2**

**0018 25F0 14 add A,B**

**001A F9 15 mov R1,A**

**001B EB 16 mov A,R3**

**001C 3400 17 addc A,#00h**

**001E FA 18 mov R2,A**

**001F 8890 19 mov P1,R0**

**0021 120031 20 lcall time**

**0024 8990 21 mov P1,R1**

**0026 120031 22 lcall time**

**0029 8A90 23 mov P1,R2**

**002B 120031 24 lcall time**

**002E 020000 25 ljmp cubo**

**0031 7F10 26 time: mov R7,#10H**

**0033 7EFA 27 paca: mov R6,#0FAH**

**0035 7DFA 28 aca: MOV R5,#0FAH**

**0037 DDFE 29 aquí: DJNZ R5,aquí**

**0039 DEFA 30 djnz R6,aca**

**003B DFF6 31 djnz R7,paca**

**003D 22 32 ret**

**\*\*\*\*WARNING:Missing`END’directive**

**ASSEMBLY COMPLETE, NO ERRORS FOUND, 1 WARNING5. Observaciones y Conclusiones**

|  |
| --- |
| **Conclusiones** |
| **Se me hizo curioso pensar en “expandir” al momento de hacer la parte de sumar los dos valores de en medio** |
| **Estuvo medio sencillo el codigo (aunque siempre se expande por mover todo al acumulador)** |
| **Aunque mi circuito estuvo fallando bastante (es decir, bastantes cortos o algo asi) se pudo entregar** |
|  |
|  |
|  |
|  |
|  |
|  |