|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Carrera:** | **Desarrollo de Software** | | | | **Academia:** | **Sistemas Digitales** | **Plantel:** | **Colomos** |
| **Materia:** | **Arquitectura y Organización de Computadoras** | | | | **Clave:** | **18MPBDS0514** | **Revisión:** | **A** |
| **Unidad:** | **3.** **Programación de un sistema basado en microprocesador en un lenguaje de bajo nivel.** | | | | **Tema:** | **La programación.** | | |
| **No. de Práctica:** | **11** | **Nombre de la práctica:** | | **Promedios variados** | | | | |
| **Profesor:** | **Antonio Lozano González** | | | | | | | |
| **Alumno:** | **Emmanuel Buenrostro Briseño** | | | | | | **Registro:** | **22300891** |
| **Alumno:** |  | | | | | | **Registro:** |  |
| **Semestre:** | **5** | **Grupo:** | **I1** | | **Período:** | **Ago-Dic 2024** | **Fecha:** | **10/12/24** |

**1. Objetivo.** **Hacer una suma de varios números hexadecimales con el sistema, y sacar el promedio de esos números para ir aprendiendo a manejar los datos en decimal como parte de las visualizaciones en el mundo normal.**

**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:**

Utilizando su sistema mínimo y el dip switch, además de la pantalla LCD, deberán introducir varios números (valores) por el puerto tres, su programación consistirá en introducir estos valores de 8 bits y sacar el promedio correspondiente, es importante mencionar que los valores introducidos podrán variar desde dos valores, hasta diez valores.. En la pantalla LCD, se vera cada dato introducido y  el resultado, que es el promedio, también se vera en la LCD

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

En esta practica el algoritmo que utilizamos es usando lo mismo que en la practica anterior que es leyendo los valores de 8 bits de 4 en 4.

Tambien ocupamos convertirlo los valores binarios a bcd, como ahora son valores de 8 bits lo que vamos a hacer es restarle #64H mientras podamos, y el total de veces que restemos es el número de las centenas (porque 64H es 100 en base 10), luego repetimos lo mismo restando 10 mientras podamos, ese valor es el valor de las decenas de el número, y el resto es el valor de las unidad.

Entonces hacemos n veces el siguiente proceso: leer el valor de 8 bits, convertirlo a bcd, imprimirlo en la pantalla, y agregarlo a la suma total (queda en 2 localidades).

Ahora queremos hacer la división entre n, para eso vamos a restar n de la cantidad total (tomando en cuenta las dos localidades), y la cantidad de veces que podamos restar eso sin irnos a números negativos es el resultado del promedio.

Entonces convertimos ese promedio a bcd y lo imprimimos en la pantalla.

**c) Código LST.**

promediopantalla PAGE 1

0000 C2B0 1 pant: CLR P3.0

0002 C2B1 2 CLR P3.1

0004 C2B2 3 CLR P3.2

0006 759038 4 mov P1,#38H

0009 12016B 5 lcall de5ms

000C 759038 6 mov P1,#38H

000F 12016B 7 lcall de5ms

0012 759038 8 mov P1,#38H

0015 12016B 9 lcall de5ms

0018 759038 10 mov P1,#38H

001B 12016B 11 lcall de5ms

001E 75900C 12 mov P1,#0CH

0021 12016B 13 lcall de5ms

0024 759001 14 mov P1,#01H

0027 12016B 15 lcall de5ms

002A 759006 16 mov P1,#06H

002D 12016B 17 lcall de5ms

0030 759080 18 mov P1,#80H

0033 753E00 19 mov 3EH, #00H

0036 753F00 20 mov 3FH, #00H

0039 783E 21 inicio: mov R0,#3EH

22

003B 120089 23 lcall leer

003E AA3F 24 mov R2,3FH

0040 75F00E 25 mov B,#0EH

0043 755000 26 mov 50h,#00H

0046 755100 27 MOV 51H,#00H

28

0049 120089 29 auxi: lcall leer

004C E5F0 30 mov A,B

004E 9404 31 SUBB A,#04H

0050 5009 32 JNC ncarry3

0052 7590C0 33 mov P1,#0C0H

0055 12016B 34 lcall de5ms

0058 75F0FF 35 mov B,#0FFH

005B DAEC 36 ncarry3: DJNZ R2,auxi

37

38 prom:

005D E551 39 ayuda1: mov A,51H

005F C3 40 CLR C

0060 C2D6 41 CLR AC

0062 953F 42 subb A,3FH

0064 500B 43 JNC ncarry4

0066 F5F0 44 mov B,A

0068 E550 45 mov A,50H

006A 600F 46 JZ fin

006C 14 47 DEC A

006D F550 48 mov 50H,A

006F E5F0 49 mov A,B

0071 F551 50 ncarry4: mov 51H,A

0073 E53E 51 mov A,3EH

0075 04 52 INC A

0076 F53E 53 mov 3EH,A

0078 02005D 54 ljmp ayuda1

55

007B 853E60 56 fin: mov 60H,3EH

007E 12012C 57 lcall bitabcd

0081 D2B0 58 setb P3.0

0083 1200B8 59 lcall coso1

60

0086 020086 61 ciclo: ljmp ciclo

0089 12017B 62 leer: lcall time

008C E5B0 63 mov A,P3

008E 54F0 64 anl A,#0F0H

0090 F570 65 mov 70H,A

0092 D2B2 66 setB P3.2

0094 12017B 67 lcall time

0097 C2B2 68 CLR P3.2

0099 E5B0 69 mov A,P3

009B 54F0 70 anl A,#0F0H

009D C4 71 swap A

009E 4570 72 orl A,70H

00A0 08 73 INC R0

00A1 F6 74 mov @R0,A

00A2 F560 75 mov 60H,A

00A4 E551 76 mov A,51H

00A6 C3 77 CLR C

00A7 C2D6 78 CLR AC

00A9 2560 79 ADD A,60H

00AB F551 80 MOV 51H,A

00AD E550 81 MOV A,50H

00AF 3400 82 ADDC A,#00H

00B1 F550 83 MOV 50H,A

00B3 12012C 84 lcall bitabcd

00B6 D2B0 85 setb P3.0

00B8 E552 86 coso1: mov A,52H

00BA 602A 87 JZ coso2

00BC 15F0 88 DEC B

00BE 4430 89 orl A,#30H

00C0 F590 90 mov P1,A

00C2 12016B 91 lcall de5ms

00C5 E553 92 mov A,53H

00C7 15F0 93 DEC B

00C9 4430 94 orl A,#30H

00CB F590 95 mov P1,A

00CD 12016B 96 lcall de5ms

00D0 E554 97 mov A,54H

00D2 15F0 98 DEC B

00D4 4430 99 orl A,#30H

00D6 F590 100 mov P1,A

00D8 12016B 101 lcall de5ms

00DB 759020 102 mov P1,#20H

00DE 15F0 103 DEC B

00E0 12016B 104 lcall de5ms

00E3 C2B0 105 clr P3.0

00E5 22 106 ret

107

00E6 E553 108 coso2: mov A,53H

00E8 601F 109 JZ coso3

00EA 15F0 110 DEC B

00EC 4430 111 orl A,#30H

00EE F590 112 mov P1,A

00F0 12016B 113 lcall de5ms

00F3 E554 114 mov A,54H

00F5 15F0 115 DEC B

00F7 4430 116 orl A,#30H

00F9 F590 117 mov P1,A

00FB 12016B 118 lcall de5ms

00FE 759020 119 mov P1,#20H

0101 15F0 120 DEC B

0103 12016B 121 lcall de5ms

0106 C2B0 122 clr P3.0

0108 22 123 ret

124

0109 E554 125 coso3: mov A,54H

010B 6014 126 JZ coso4

010D 15F0 127 DEC B

010F 4430 128 orl A,#30H

0111 F590 129 mov P1,A

0113 12016B 130 lcall de5ms

0116 759020 131 mov P1,#20H

0119 15F0 132 DEC B

011B 12016B 133 lcall de5ms

011E C2B0 134 clr P3.0

0120 22 135 ret

136

0121 759020 137 coso4: mov P1,#20H

0124 15F0 138 DEC B

0126 12016B 139 lcall de5ms

0129 C2B0 140 clr P3.0

012B 22 141 ret

142

012C 755200 143 bitabcd:mov 52H,#00H

012F 755300 144 mov 53H,#00H

0132 755400 145 mov 54H,#00H

0135 E560 146 cien: mov A,60H

0137 C3 147 CLR C

0138 C2D6 148 CLR AC

013A 9464 149 subb A,#64H

013C 5005 150 JNC ncarry1

013E 2464 151 ADD A,#64H

0140 02014E 152 LJMP diez

0143 F560 153 ncarry1: MOV 60H,A

0145 E552 154 MOV A,52H

0147 2401 155 ADD A,#01H

0149 F552 156 MOV 52H,A

014B 020135 157 LJMP cien

158

014E E560 159 diez: mov A,60H

0150 C3 160 CLR C

0151 C2D6 161 CLR AC

0153 940A 162 subb A,#0AH

0155 5005 163 JNC ncarry2

0157 240A 164 ADD A,#0AH

0159 020167 165 LJMP uno

015C F560 166 ncarry2: MOV 60H,A

015E E553 167 MOV A,53H

0160 2401 168 ADD A,#01H

0162 F553 169 MOV 53H,A

0164 02014E 170 LJMP diez

0167 856054 171 uno: MOV 54H,60H

016A 22 172 ret

173

174 de5ms:

016B D2B1 175 SETB P3.1

016D 7F0A 176 mov R7, #0AH

016F 7EFA 177 aca: mov R6,#0FAH

0171 DEFE 178 aqui2: DJNZ R6, aqui2

0173 DFFA 179 DJNZ R7, aca

0175 C2B1 180 CLR P3.1

0177 22 181 RET

182

183

0178 020178 184 time2: ljmp time2

185 time:

017B 7F40 186 mov R7,#40H

017D 7EFA 187 paca: mov R6,#0FAH

017F 7DFA 188 aca2: MOV R5,#0FAH

0181 DDFE 189 aqui3: DJNZ R5,aqui3

0183 DEFA 190 djnz R6,aca2

0185 DFF6 191 djnz R7,paca

0187 22 192 ret

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

ASSEMBLY COMPLETE, NO ERRORS FOUND, 1 WARNING

**5. Observaciones y Conclusiones**

|  |
| --- |
| **Conclusiones** |
| *Creo que al principio me costo abstante er el como codear esto, ya que la división no se veía tan trivial* |
| *Además de que tenia un poco de miedo porque el parcial pasado no me salio la practica de promedios aunque* |
| *era específicamente de 8 bits, pero despues de ver como hacer la división todo quedo bastante mas claro, y .* |
| con la explicación de como hacer el binario a bcd, se me hizo una practica bastante divertida. |