# Album de las practicas realizadas en el semestre

Emmanuel Buenrostro Briseño

22300891

5°I

Arquitectura y Organización de Computadoras

Agosto-Diciembre 2024

Profesor: Antonio Lozano González

# CETI Plantel Colomos

# Índice

1. **Índice ------------------------------------------------------------ 2**
2. **Contador Binario --------------------------------------------- 3**
3. **Grabación Lectura de una EEPROM --------------------- 8**
4. **Grabación Lectura Automática de una memoria EEPROM ----------------------------------------------------- 15**
5. **Sistema mínimo ----------------------------------------------- 22**
6. **Retardo --------------------------------------------------------- 26**
7. **Secuencias Especiales ---------------------------------------- 29**
8. **Promedios ------------------------------------------------------ No**
9. **Potencia de un número -------------------------------------- 33**
10. **Manejo de la pantalla LCD --------------------------------- 36**

**10.Comparación de Letras utilizando la pantalla LCD – 41**

**11. Promedios variados ----------------------------------------- 48**

**12. Uso de los timmers internos del microcontrolador --- 55**

**13. Uso de las interrupciones internas ----------------------- 63**

Contador Binario

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Carrera:** | **Desarrollo de Software** | | | | **Academia:** | **Sistemas Digitales** | **Plantel:** | **Colomos** |
| **Materia:** | **Arquitectura y Organización de Computadoras** | | | | **Clave:** | **18MPBDS0514** | **Revisión:** | **A** |
| **Unidad:** | 1. **Fundamentos de Arquitectura y Organización de Computadoras** | | | | **Tema:** | **Los elementos de memoria y su organización jerárquica.** | | |
| **No. de Práctica:** | **1** | **Nombre de la práctica:** | | **Contador Binario** | | | | |
| **Profesor:** | **Antonio Lozano Gonzales** | | | | | | | |
| **Alumno:** | **Emmanuel Buenrostro Briseño** | | | | | | **Registro:** | **22300891** |
| **Semestre:** | **5** | **Grupo:** | **I** | | **Período:** | **Ago-Dic 2024** | **Fecha:** | **16/9/24** |

1. **Objetivo**

Realizar un conteo binario de 4 bits para su utilización como medio para incrementar direcciones de una memoria.

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

|  |  |
| --- | --- |
| **Cantidad** | **Material, Equipo y/o Herramientas** |
| **1** | **Proto** |
| **1** | **555** |
| **2** | **Capacitor** |
| **1** | **74ls191** |
| **6** | **Resistencias** |
| **4** | **Leds** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1. **Desarrollo de la práctica**
2. **Consideraciones de diseño**

Realizar un contador de 4 bits, puede contar de 0 a 9 o bien, de 0 a F. Utilizar cuatro Flip Flop para esta tarea o algún contador que tengan (74190, 74191 etc.) Pueden usar un botón pulsador para incrementar la cuenta, o un circuito, como puede ser el 555, si utilizan este último favor de usar una baja frecuencia para poder ver la cuenta.

1. **Configuración y características de componentes**

Escribir las características eléctricas obtenidas del manual del fabricante de los elementos utilizados más importantes.

**A diagram of a computer component

Description automatically generatedA diagram of a circuit

Description automatically generated**

1. **Implementación (diagrama eléctrico)**

**A diagram of a circuit

Description automatically generated**

A white electronic board with wires and wires

Description automatically generated

1. **Observaciones y Conclusiones**

|  |
| --- |
|  |
| **Esta practica fue sencilla debido a que ya hemos visto bastantes veces este circuito tanto en Digitales 2** |
| **Como en temas de electrónica, aunque siempre es algo útil para recordar como usar el 74ls191 y el 555** |
|  |

Grabación Lectura de una EEPROM

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Carrera:** | **Desarrollo de Software** | | | | **Academia:** | **Sistemas Digitales** | **Plantel:** | **Colomos** |
| **Materia:** | **Arquitectura y Organización de Computadoras** | | | | **Clave:** | **18MPBDS0514** | **Revisión:** | **A** |
| **Unidad:** | 1. **Fundamentos de Arquitectura y Organización de Computadoras** | | | | **Tema:** | **Los diferentes tipos de memoria interna de una computadora.** | | |
| **No. de Práctica:** | **2** | **Nombre de la práctica:** | | **Grabación Lectura de una EEPROM** | | | | |
| **Profesor:** | **Antonio Lozano González** | | | | | | | |
| **Alumno:** | **Emmanuel Buenrostro Briseño** | | | | | | **Registro:** | 22300891 |
| **Alumno:** |  | | | | | | **Registro:** |  |
| **Semestre:** | **5to** | **Grupo:** | **I1** | | **Período:** | **Ago-Dic 2024** | **Fecha:** | **29/08/24** |

1. **Objetivo**

Grabar una memoria EEPROM, para su posterior uso como memoria de un sistema mínimo.

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

|  |  |
| --- | --- |
| **Cantidad** | **Material, Equipo y/o Herramientas** |
| **1** | **Memoria EEPROM AT28C64** |
|  | **cable de proto** |
| **12** | **Resistencias de 330** |
| **8** | **Leds** |
| **1** | **Proto** |
| **1** | **Dip switch de 4 entradas** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1. **Desarrollo de la práctica**
2. **Consideraciones de diseño**

A white circuit board with wires and wires

Description automatically generated

Tabla:

| **Binario** | **Hexadecimal** | **Decimal** |
| --- | --- | --- |

|  |  |  |
| --- | --- | --- |
| 01011011 | 5B | 91 |

|  |  |  |
| --- | --- | --- |
| 11100010 | E2 | 226 |

|  |  |  |
| --- | --- | --- |
| 00101100 | 2C | 44 |

|  |  |  |
| --- | --- | --- |
| 10110101 | B5 | 181 |

|  |  |  |
| --- | --- | --- |
| 01100110 | 66 | 102 |

|  |  |  |
| --- | --- | --- |
| 10011101 | 9D | 157 |

|  |  |  |
| --- | --- | --- |
| 00010111 | 17 | 23 |

|  |  |  |
| --- | --- | --- |
| 11111001 | F9 | 249 |

|  |  |  |
| --- | --- | --- |
| 00111101 | 3D | 61 |

|  |  |  |
| --- | --- | --- |
| 01001010 | 4A | 74 |

|  |  |  |
| --- | --- | --- |
| 10101011 | AB | 171 |

|  |  |  |
| --- | --- | --- |
| 11000101 | C5 | 197 |

|  |  |  |
| --- | --- | --- |
| 01111011 | 7B | 123 |

|  |  |  |
| --- | --- | --- |
| 00011010 | 1A | 26 |

|  |  |  |
| --- | --- | --- |
| 10001111 | 8F | 143 |

|  |  |  |
| --- | --- | --- |
| 11011100 | DC | 220 |

1. **Configuración y características de componentes**

Escribir las características eléctricas obtenidas del manual del fabricante de los elementos utilizados

La AT28C64 es una memoria EEPROM (Electrically Erasable Programmable Read-Only Memory) fabricada por Microchip Technology. Aquí tienes algunas características eléctricas típicas de este dispositivo:

Voltaje de operación: El rango típico de voltaje de operación es de 4.5V a 5.5V.

Consumo de corriente:

Lectura: Suele consumir alrededor de 30 mA (máximo).

Escritura: El consumo puede variar, pero suele estar en el rango de 60 mA (máximo).

Standby: Consumo de corriente muy bajo en modo de espera.

Tiempo de acceso:

Tiempo de lectura: Generalmente en el rango de 150 a 450 nanosegundos (ns).

Tiempo de escritura: Depende del modo de escritura, pero suele ser en el rango de 2 a 10 milisegundos (ms).

Ciclos de escritura/lectura: La AT28C64 suele tener una alta durabilidad con una gran cantidad de ciclos de escritura/lectura, típicamente más de 10,000 ciclos.

Interfaz de comunicación: Utiliza una interfaz paralela de datos y direcciones para la lectura y escritura de datos.

Capacidad: La AT28C64 tiene una capacidad de 64 kilobits, organizada en 8 kilobytes (8K) de memoria.

1. **Diagramas:**

**A close-up of a test

Description automatically generated**

1. **Implementación (diagrama eléctrico)**

**A diagram of a computer circuit

Description automatically generated**

1. **Observaciones y Conclusiones**

|  |
| --- |
| **Es muy importante tomar en cuenta el modo en el que se conecta la memoria a la protoboard para no dañar la misma** |
| **Esto debido a que cualquier voltaje que entre por un lugar donde no deberia podría causar daños en los** |
| **Circuitos de la memoria ya que estas ocasionando una reacción no esperada, o un voltaje que no acepta.** |
| **Y entonces se puede quemar, teniendo que comprar una nueva.** |
| **Las protoboards pueden tener defectos** |
| **En mi caso habia algún corto circuito porque los leds que no debían prender a veces prendían parpadeando** |
| **Y entonces cada vez que tocabas la memoria se estabilizaban, pudiendo ser una muestra de que habia** |
| **Algún defecto en mi protoboard que provocaba esos cortos circuitos.** |
| **Conclusiones** |
| **Es algo bastante interesante, el como guardas la información por 10 algún tiempo para despues mostrarla en** |
| **Leds, o en general, en voltaje que ocasiona que puedas hacer mas cosas especificas porque ahora** |
| **Tienes una forma de guardar ciertos valores de información para distintos circuitos y sin que se borre.** |

Grabación Lectura automática de una memoria EEPROM

Carita Feliz -> :D

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Carrera:** | **Desarrollo de Software** | | | | **Academia:** | **Sistemas Digitales** | **Plantel:** | **Colomos** |
| **Materia:** | **Arquitectura y Organización de Computadoras** | | | | **Clave:** | **18MPBDS0514** | **Revisión:** | **A** |
| **Unidad:** | 1. **Fundamentos de Arquitectura y Organización de Computadoras** | | | | **Tema:** | **Los diferentes tipos de memoria externa de una computadora.** | | |
| **No. de Práctica:** | **3** | **Nombre de la práctica:** | | **Grabación Lectura automática de una memoria EEPROM** | | | | |
| **Profesor:** | **Antonio Lozano González** | | | | | | | |
| **Alumno:** | Emmanuel Buenrostro Briseño | | | | | | **Registro:** | **22300891** |
| **Semestre:** | **5** | **Grupo:** | **5I1** | | **Período:** | **Ago-Dic 2024** | **Fecha:** | **16/09/2024** |

1. **Objetivo**

Grabar una memoria EEPROM, y visualizar nuestro registro de forma automática, para conocer como desplazarse en la memoria.

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

|  |  |
| --- | --- |
| **Cantidad** | **Material, Equipo y/o Herramientas** |
| **2** | **protoboard** |
| **1** | **memoria at28c64** |
| **1** | **display 7 segmentos anodo comun** |
| **1** | **555** |
| **1** | **74ls191** |
| **-** | **cable para proto** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1. **Desarrollo de la práctica**
2. **Consideraciones de diseño**
3. **Configuración y características de componentes**

Escribir las características eléctricas obtenidas del manual del fabricante de los elementos utilizados más importantes.

Memoria at28c64.

A diagram of a computer circuit

Description automatically generated with medium confidenceA table with text on it

Description automatically generated

74ls191:

A diagram of a device

Description automatically generated

555:

A diagram of a device

Description automatically generated

1. **Diagramas:**

**A diagram of a wiring diagram

Description automatically generated**

**Valores en la memoria:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Ub** | **Binario** | **Hexadecimal** | **Digito en display** |
| **00** | **00100100** | **24** | **2** |
| **01** | **00100100** | **24** | **2** |
| **02** | **00110000** | **30** | **3** |
| **03** | **01000000** | **40** | **0** |
| **04** | **01000000** | **40** | **0** |
| **05** | **00000000** | **00** | **8** |
| **06** | **00010000** | **10** | **9** |
| **07** | **01111001** | **79** | **1** |
| **08** | **00100100** | **24** | **2** |
| **09** | **00100100** | **24** | **2** |
| **0A** | **00110000** | **30** | **3** |
| **0B** | **01000000** | **40** | **0** |
| **0C** | **01000000** | **40** | **0** |
| **0D** | **00000000** | **00** | **8** |
| **0E** | **00010000** | **10** | **9** |
| **0F** | **01111001** | **79** | **1** |

1. **Implementación (diagrama eléctrico)**

**A white board with wires and a digital display

Description automatically generated**

1. **Observaciones y Conclusiones**

|  |
| --- |
|  |
| Esta practica se vuelve bastante simple si tienes las dos practicas anteriores, ya que es solo juntarlas y cambiar los valores guardados en la memoria. Sin embargo, es bastante interesante el como se puede usar estas cosas ya que justo aprovechas al máximo el guardar información repitiendo un ciclo.  Tambien hay que tener cuidado al retirar la memoria porque se me rompió una pata. |

Sistema Mínimo

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Carrera:** | **Desarrollo de Software** | | | | **Academia:** | **Sistemas Digitales** | **Plantel:** | **Colomos** |
| **Materia:** | **Arquitectura y Organización de Computadoras** | | | | **Clave:** | **18MPBDS0514** | **Revisión:** | **A** |
| **Unidad:** | 1. **Elementos de un Microprocesador y su arquitectura interna.** | | | | **Tema:** | **Definición de microprocesador como parte de una computadora.** | | |
| **No. de Práctica:** | **4** | **Nombre de la práctica:** | | **Sistema mínimo** | | | | |
| **Profesor:** | **Antonio Lozano González** | | | | | | | |
| **Alumno:** | **Emmanuel Buenrostro Briseño** | | | | | | **Registro:** | **22300891** |
| **Alumno:** |  | | | | | | **Registro:** |  |
| **Semestre:** | **5** | **Grupo:** |  | | **Período:** |  | **Fecha:** | **28/10/24** |

1. **Objetivo**

Hacer un sistema mínimo, utilizando: memoria, CPU, buses, dispositivos de entrada y salida, para programarlo en lenguaje de bajo nivel, como conocimientos previos de sistemas embebidos.

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

|  |  |
| --- | --- |
| **Cantidad** | **Material, Equipo y/o Herramientas** |
| **1** | **Microcontrolador AT89S52** |
| **1** | **Memoria AT28C64B** |
| **1** | **Latch 74573** |
| **1** | **Buffer 74245** |
| **1** | **Minidip** |
| **8** | **Leds** |
| **9** | **Resistencias** |
| **1** | **Push Button** |
| **2** | **Protos** |
| **1** | **Cristal Cuarzo 12 MHz** |
| **Mucho** | **Cable para proto** |
| **2** | **Capacitores 33pF** |

1. **Desarrollo de la práctica**
2. **Consideraciones de diseño**

Utilizando una memoria EEPROM como memoria externa de un sistema computacional y un microcontrolador como un CPU y, además agregándole entradas para introducir información y, salidas para poder ver resultados; hacer un sistema mínimo interconectado por medio de buses, en el reporte de practica agregar un diagrama de las conexiones realizadas.  
PD. Se está dando mucho tiempo, no por esto se debe dejar al final, la recomendación es que se haga lo más pronto posible, por cualquier inconveniente que se pueda presentar.

1. **Configuración y características de componentes**

Escribir las características eléctricas obtenidas del manual del fabricante de los elementos utilizados más importantes.

**Microcontrolador AT89S52**

* **Voltaje de alimentación (Vcc):** 4.0V a 5.5V
* **Corriente de operación (típica):** Aproximadamente 20 mA a 5V
* **Velocidad máxima de operación:** 33 MHz
* **Corriente de entrada/salida en pines I/O:** Máximo ±10 mA (por pin)
* **Corriente total de salida de pines:** Máximo 71 mA
* **Corriente en modo Power-down:** <1 µA a 5V

**Memoria EEPROM AT28C64B**

* **Voltaje de alimentación (Vcc):** 4.5V a 5.5V
* **Corriente en modo activo:** 30 mA (lectura), 20 mA (escritura)
* **Corriente en modo standby:** 100 µA
* **Tiempo de acceso (tAA):** 150 ns (típico)
* **Resistencia mínima a borrado/escritura:** 100,000 ciclos por celda

**Latch 74573 (Latch de 8 bits)**

* **Voltaje de alimentación (Vcc):** 4.5V a 5.5V
* **Corriente de operación:** 80 µA (típico) en estado bajo, con carga mínima
* **Corriente de salida máxima:** ±35 mA (por pin)
* **Tiempo de propagación:** 15-20 ns
* **Corriente total máxima de salida:** 70 mA

**Buffer 74245 (Transceptor bidireccional de 8 bits)**

* **Voltaje de alimentación (Vcc):** 4.5V a 5.5V
* **Corriente de operación:** Aproximadamente 40 µA en modo standby
* **Corriente de salida máxima:** ±35 mA (por pin)
* **Tiempo de propagación:** 10-15 ns (típico)
* **Corriente total máxima de salida:** 70 mA

1. **Diagramas (puede ser la foro del pintarron)**

A white board with a diagram

Description automatically generated

1. **Implementación (foto)**

A circuit board with wires and wires

Description automatically generated

1. **Observaciones y Conclusiones**

|  |
| --- |
| **Conclusiones** |
| **Es demasiado cableado (bastante mas que cualquier otra practica que e hecho) y no se como le voy a hacer** |
| **Si tengo que encontrar algun error en esta cosa, siendo la mejor opción probablemente recablearlo (mas que** |
| **Nada porque pense que era buen acomodo pero no lo es y quedaron cables muy grandes y largos y entrelazados)** |

Retardo

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Carrera:** | **Desarrollo de Software** | | | | **Academia:** | **Sistemas Digitales** | **Plantel:** | **Colomos** |
| **Materia:** | **Arquitectura y Organización de Computadoras** | | | | **Clave:** | **18MPBDS0514** | **Revisión:** | **A** |
| **Unidad:** | 1. **Programación de un sistema basado en microprocesador en un lenguaje de bajo nivel.** | | | | **Tema:** | **La programación.** | | |
| **No. de Práctica:** | **5** | **Nombre de la práctica:** | | **Retardo** | | | | |
| **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 un retardo de x+5 segundos, para ir conociendo el lenguaje 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:**

Utilizando su sistema mínimo, deberán hacer un retardo de (x+5) segundos, Dicho retardo se mostrará con un led encendido y apagado durante esos segundos. Donde x, es el último dígito de tu registro.

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

**Lo que hacemos para hacer el retraso es en un espacio de memoria guardar 8\*(segundos) y hacer que mientras ese no sea cero (con un djnz) haga un ciclo de mover ciertos registros un montón de veces (la cantidad contada como para que 8 de esos sea un segundo) y al hacerlo (Segundos) veces se tarda los segundos que queremos que se tarde.**

**c) Código LST.**

**0000 759001 1 inicio: mov P1, #01H**

**0003 12000F 2 lcall time**

**0006 759000 3 mov P1,#00H**

**0009 12000F 4 lcall time**

**000C 020000 5 ljmp inicio**

**000F 7A30 6 time: mov R2,#30H**

**0011 79FA 7 paca: mov R1,#0FAH**

**0013 78FA 8 aca: MOV R0,#0FAH**

**0015 D8FE 9 aqui: DJNZ R0,aqui**

**0017 D9FA 10 djnz R1,aca**

**0019 DAF6 11 djnz R2,paca**

**001B 22 12 ret**

**5. Observaciones y Conclusiones**

|  |
| --- |
| **Conclusiones** |
| **Este es un codigo que vamos a utilizar un montón en el resto de las practicas para tener un control mas humano** |
| **Sobre lo que hacen los códigos ya que trabajar a la velocidad del micro (para nosotros poder hacer los cambios que ocupamos)** |
| **Es imposible, y poder pausarlo es bastante útil.** |

Secuencias especiales

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Carrera:** | **Desarrollo de Software** | | | | **Academia:** | **Sistemas Digitales** | **Plantel:** | **Colomos** |
| **Materia:** | **Arquitectura y Organización de Computadoras** | | | | **Clave:** | **18MPBDS0514** | **Revisión:** | **A** |
| **Unidad:** | 1. **Programación de un sistema basado en microprocesador en un lenguaje de bajo nivel.** | | | | **Tema:** | **La programación.** | | |
| **No. de Práctica:** | **6** | **Nombre de la práctica:** | | **Secuencias especiales** | | | | |
| **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.** Realizar desplazamientos en los leds, para ver como dichos ciclos y/o corrimientos pueden representar  
en realidad operaciones aritméticas.

**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:**

Hacer cuatro secuencias especiales con  
sus leds de salida, dichas secuencias deberán ser elegidas utilizando su puerto  
de entrada, para elegir sus secuencias, podrá utilizar bits o valores  
preestablecidos, las secuencias tienen que tener retardo para poderlas  
apreciar, use su imaginación para realizarlas, además, aunque sea obvio,  
recuerde que las veremos en los leds de su puerto de salida.**b) Algoritmo o Diagrama de Flujo.**

**En este codigo usamos el codigo de retardo que hicimos anteriormente, y mediante el uso de cjne detectamos que numero esta metido en el P3, para entonces ver que patrón vamos a hacer , en cada patrón lo que hacemos es mostrar cierto numero en el P1, ese valor es que leds queremos que se enciendan para despues agregar un retardo pequeño y que se vaya cambiando, para asi se cree el patrón y sea visible para los humanos.**

**c) Código LST.**

**disenos PAGE 1**

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

**0002 B40130 2 escal: cjne A,#01H, stair**

**0005 759001 3 mov P1,#01H**

**0008 12007D 4 lcall time**

**000B 759002 5 mov P1,#02H**

**000E 12007D 6 lcall time**

**0011 759004 7 mov P1,#04H**

**0014 12007D 8 lcall time**

**0017 759008 9 mov P1,#08H**

**001A 12007D 10 lcall time**

**001D 759010 11 mov P1,#10H**

**0020 12007D 12 lcall time**

**0023 759020 13 mov P1,#20H**

**0026 12007D 14 lcall time**

**0029 759040 15 mov P1,#40H**

**002C 12007D 16 lcall time**

**002F 759080 17 mov P1,#80H**

**0032 12007D 18 lcall time**

**19**

**0035 B40218 20 stair: cjne A,#02H, half**

**0038 759018 21 mov P1,#18H**

**003B 12007D 22 lcall time**

**003E 759024 23 mov P1,#24H**

**0041 12007D 24 lcall time**

**0044 759042 25 mov P1,#42H**

**0047 12007D 26 lcall time**

**004A 759081 27 mov P1,#81H**

**004D 12007D 28 lcall time**

**0050 B4040C 29 half: cjne A,#04H, impar**

**0053 7590F0 30 mov P1,#0F0H**

**0056 12007D 31 lcall time**

**0059 75900F 32 mov P1,#0FH**

**005C 12007D 33 lcall time**

**005F B4089E 34 impar: cjne A,#08H, inicio**

**0062 759001 35 mov P1,#01H**

**0065 12007D 36 lcall time**

**0068 759004 37 mov P1,#04H**

**006B 12007D 38 lcall time**

**006E 759010 39 mov P1,#10H**

**0071 12007D 40 lcall time**

**0074 759040 41 mov P1,#40H**

**0077 12007D 42 lcall time**

**007A 020000 43 ljmp inicio**

**007D 7A04 44 time: mov R2,#04H**

**007F 79FA 45 paca: mov R1,#0FAH**

**0081 78FA 46 aca: MOV R0,#0FAH**

**0083 D8FE 47 aqui: DJNZ R0,aqui**

**0085 D9FA 48 djnz R1,aca**

**0087 DAF6 49 djnz R2,paca**

**0089 22 50 ret**

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

**ASSEMBLY COMPLETE, NO ERRORS FOUND, 1 WARNING**

**5. Observaciones y Conclusiones**

|  |
| --- |
| **Conclusiones** |
| **Ya puedo hacer lucesitas led para navidad, creo que es una muy buena primera aplicación del retardo** |
| **Y poder hacer esos “if” con el cjne al momento de decidir el patron es bastante interesante.** |

Potencia de un Número

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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 WARNING**

**5. 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** |

Manejo de la pantalla LCD

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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:** | **9** | **Nombre de la práctica:** | | **Manejo de la pantalla LCD** | | | | |
| **Profesor:** | **Antonio Lozano González** | | | | | | | |
| **Alumno:** | **Emmanuel Buenrostro Briseño** | | | | | | **Registro:** | **22300891** |
| **Alumno:** |  | | | | | | **Registro:** |  |
| **Semestre:** | **5** | **Grupo:** | **I1** | | **Período:** | **Ago-Dic 2024** | **Fecha:** | **24/11/24** |

**1. Objetivo.** **Inicializar una `pantalla LCD, como medio de presentación de  
información, en su sistema mínimo.**

**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, deberá poner en funcionamiento una pantalla LCD, por medio de  
programación, deberá inicializar esta LCD, para después mostrar su nombre y/o apellido, el que tenga mayor numero de caracteres.    
Dicha palabra ira en la línea superior si es par su último dígito de registro  o en la línea inferior si es impar; pero cada letra deberá tardar  medio segundo en aparecer; además, después de haber completado dicha palabra, deberán volverla a imprimir en la linea contraria (con las mismas condiciones de tiempo), pero desde el lado opuesto. AL final deberán estar las dos palabras en la pantalla y se detiene o vuelve a empezar, es indistinto.

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

Primero hacemos toda la inicialización de la pantalla LCD, para despues poner el cursos en la parte de abajo y poner mi nombre B U E N R O S T R O (en ASCII) y ya despues poner el cursor en la ultima de la primer linea pero cambiar que ahora se vaya para atrás y pongo mi nombre al reves O R T S O R N E U B

**c) Código LST.**

**nombre PAGE 1**

**0000 C2B0 1 inicio: CLR P3.0**

**0002 C2B1 2 CLR P3.1**

**0004 759038 3 mov P1,#38h**

**0007 1200FD 4 lcall de120us**

**000A 759038 5 mov P1,#38h**

**000D 1200FD 6 lcall de120us**

**0010 759038 7 mov P1,#38h**

**0013 1200FD 8 lcall de120us**

**0016 759038 9 mov P1,#38h**

**0019 1200FD 10 lcall de120us**

**001C 75900C 11 mov P1,#0CH**

**001F 1200FD 12 lcall de120us**

**0022 759001 13 mov P1,#01H**

**0025 1200FD 14 lcall de5ms**

**0028 759006 15 mov P1,#06H**

**002B 1200FD 16 lcall de120us**

**002E 7590C0 17 mov P1,#0C0H**

**0031 1200FD 18 lcall de120us**

**0034 D2B0 19 setb P3.0**

**0036 759042 20 mov P1,#42H**

**0039 1200FD 21 lcall de120us**

**003C 12010A 22 lcall time**

**003F 759055 23 mov P1,#55H**

**0042 1200FD 24 lcall de120us**

**0045 12010A 25 lcall time**

**0048 759045 26 mov P1,#45H**

**004B 1200FD 27 lcall de120us**

**004E 12010A 28 lcall time**

**0051 75904E 29 mov P1,#4EH**

**0054 1200FD 30 lcall de120us**

**0057 12010A 31 lcall time**

**005A 759052 32 mov P1,#52H**

**005D 1200FD 33 lcall de120us**

**0060 12010A 34 lcall time**

**0063 75904F 35 mov P1,#4FH**

**0066 1200FD 36 lcall de120us**

**0069 12010A 37 lcall time**

**006C 759053 38 mov P1,#53H**

**006F 1200FD 39 lcall de120us**

**0072 12010A 40 lcall time**

**0075 759054 41 mov P1,#54H**

**0078 1200FD 42 lcall de120us**

**007B 12010A 43 lcall time**

**007E 759052 44 mov P1,#52H**

**0081 1200FD 45 lcall de120us**

**0084 12010A 46 lcall time**

**0087 75904F 47 mov P1,#4FH**

**008A 1200FD 48 lcall de120us**

**008D 12010A 49 lcall time**

**0090 C2B0 50 CLR p3.0**

**0092 759004 51 mov P1,#04H**

**0095 1200FD 52 lcall de120us**

**0098 75908F 53 mov P1,#8FH**

**009B 1200FD 54 lcall de120us**

**009E D2B0 55 setb P3.0**

**00A0 75904F 56 mov P1,#4FH**

**00A3 1200FD 57 lcall de120us**

**00A6 12010A 58 lcall time**

**00A9 759052 59 mov P1,#52H**

**00AC 1200FD 60 LCALL de120us**

**00AF 12010A 61 lcall time**

**00B2 759054 62 mov P1,#54H**

**00B5 1200FD 63 lcall de120us**

**00B8 12010A 64 lcall time**

**00BB 759053 65 mov P1,#53H**

**00BE 1200FD 66 lcall de120us**

**00C1 12010A 67 lcall time**

**00C4 75904F 68 mov P1,#4FH**

**00C7 1200FD 69 lcall de120us**

**00CA 12010A 70 lcall time**

**00CD 759052 71 mov P1,#52H**

**00D0 1200FD 72 lcall de120us**

**00D3 12010A 73 lcall time**

**00D6 75904E 74 mov P1,#4EH**

**00D9 1200FD 75 lcall de120us**

**00DC 12010A 76 lcall time**

**00DF 759045 77 mov P1,#45H**

**00E2 1200FD 78 lcall de120us**

**00E5 12010A 79 lcall time**

**00E8 759055 80 mov P1,#55H**

**00EB 1200FD 81 lcall de120us**

**00EE 12010A 82 lcall time**

**00F1 759042 83 mov P1,#42H**

**00F4 1200FD 84 lcall de120us**

**00F7 12010A 85 lcall time**

**00FA 020000 86 ljmp inicio**

**87 de120us:**

**00FD D2B1 88 de5ms: SETB P3.1**

**00FF 790A 89 mov R1, #0AH**

**0101 78FA 90 aca: mov R0,#0FAH**

**0103 D8FE 91 aqui2: DJNZ R0, aqui2**

**0105 D9FA 92 DJNZ R1, aca**

**0107 C2B1 93 CLR P3.1**

**0109 22 94 RET**

**95**

**96**

**010A 7F04 97 time: mov R7,#04H**

**010C 7EFA 98 paca: mov R6,#0FAH**

**010E 7DFA 99 aca2: MOV R5,#0FAH**

**0110 DDFE 100 aqui3: DJNZ R5,aqui3**

**0112 DEFA 101 djnz R6,aca2**

**0114 DFF6 102 djnz R7,paca**

**0116 22 103 ret**

**104**

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

**ASSEMBLY COMPLETE, NO ERRORS FOUND, 1 WARNING**

**ERROR SUMMARY:**

**Line 104, WARNING:Missing`END'directive**

**5. Observaciones y Conclusiones**

|  |
| --- |
| **Conclusiones** |
| **La pantalla es algo interesante, al inicio pense que iba a ser mas complicado pero me salio rápido todo** |
| **Y te sientes con poder cuando usas una pantalla, creo que es la primer cosa como “grafica” que hacemos** |

Comparación de letras utilizando la pantalla LCD

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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:** | **10** | **Nombre de la práctica:** | | **Comparación de letras utilizando la pantalla LCD** | | | | |
| **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.** **Utilizar operaciones aritméticas, para encontrar la forma de saber cuál letra  
es mayor, menor o igual a otra e ir aprendiendo a utilizar el carry como  
elemento diferenciador.**

**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, deberá introducir por el dip switch, una frase que le sera proporcionada por su maestro, entonces por medio de su programación, deberá acomodar de mayor a menor cada letra si su ultimo dígito del registro es par y de menor a mayor cada letra si su ultimo dígito del registro es impar. Los datos introducidos irán en la parte superior de la pantalla y los ordenados en la parte inferior.

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

Lo que hacemos es leer los valores ascii de las letras de 8 bits (leemos de 4 en 4 y los juntamos con un orl y un swap), entonces ahora vamos a aplicar bubble sort, para eso vamos a hacer un “recorrido” de los 16 valores y haciendo una comparación entre dos consecutivos y haciendo los swaps necesarios para que quede el arreglo en orden.

Ese recorrido lo hacemos 16 veces para asegurarnos que el arreglo quede debidamente ordenado y con eso terminamos imprimiendo todos los valores en el orden en que nos dejo el bubble sort.

**c) Código LST.**

comparacion PAGE 1

0000 C2B0 1 pant: CLR P3.0

0002 C2B1 2 CLR P3.1

0004 759038 3 mov P1,#38H

0007 12021C 4 lcall de5ms

000A 759038 5 mov P1,#38H

000D 12021C 6 lcall de5ms

0010 759038 7 mov P1,#38H

0013 12021C 8 lcall de5ms

0016 759038 9 mov P1,#38H

0019 12021C 10 lcall de5ms

001C 75900C 11 mov P1,#0CH

001F 12021C 12 lcall de5ms

0022 759001 13 mov P1,#01H

0025 12021C 14 lcall de5ms

0028 759006 15 mov P1,#06H

002B 12021C 16 lcall de5ms

002E 759080 17 mov P1,#80H

0031 782F 18 inicio: mov R0,#2FH

0033 793F 19 e mov R1,#3FH

0035 120068 20 lcall readc

0038 120068 21 lcall readc

003B 120068 22 lcall readc

003E 120068 23 lcall readc

0041 120068 24 lcall readc

0044 120068 25 lcall readc

0047 120068 26 lcall readc

004A 120068 27 lcall readc

004D 120068 28 lcall readc

0050 120068 29 lcall readc

0053 120068 30 lcall readc

0056 120068 31 lcall readc

0059 120068 32 lcall readc

005C 120068 33 lcall readc

005F 120068 34 lcall readc

0062 120068 35 lcall readc

0065 020089 36 ljmp ord

37

0068 12022C 38 readc: lcall time

006B E5B0 39 mov A,P3

006D 54F0 40 anl A,#0F0H

006F F570 41 mov 70H,A

0071 12022C 42 lcall time

0074 E5B0 43 mov A,P3

0076 54F0 44 anl A,#0F0H

0078 C4 45 swap A

0079 4570 46 orl A,70H

007B 08 47 INC R0

007C 09 48 INC R1

007D F7 49 mov @R1,A

007E F6 50 mov @R0,A

007F D2B0 51 setb P3.0

0081 F590 52 mov P1,A

0083 12021C 53 lcall de5ms

0086 C2B0 54 clr P3.0

0088 22 55 ret

0089 12013A 56 ord: lcall bubble

008C 12013A 57 lcall bubble

008F 12013A 58 lcall bubble

0092 12013A 59 lcall bubble

0095 12013A 60 lcall bubble

0098 12013A 61 lcall bubble

009B 12013A 62 lcall bubble

009E 12013A 63 lcall bubble

00A1 12013A 64 lcall bubble

00A4 12013A 65 lcall bubble

00A7 12013A 66 lcall bubble

00AA 12013A 67 lcall bubble

00AD 12013A 68 lcall bubble

00B0 12013A 69 lcall bubble

00B3 12013A 70 lcall bubble

00B6 12013A 71 lcall bubble

72

00B9 7590C0 73 imp: mov P1,#0C0H

00BC 12021C 74 lcall de5ms

00BF D2B0 75 setb P3.0

00C1 854090 76 mov P1,40H

00C4 12021C 77 lcall de5ms

00C7 854190 78 mov P1,41H

00CA 12021C 79 lcall de5ms

00CD 854290 80 mov P1,42H

00D0 12021C 81 lcall de5ms

00D3 854390 82 mov P1,43H

00D6 12021C 83 lcall de5ms

00D9 854490 84 mov P1,44H

00DC 12021C 85 lcall de5ms

00DF 854590 86 mov P1,45H

00E2 12021C 87 lcall de5ms

00E5 854690 88 mov P1,46H

00E8 12021C 89 lcall de5ms

00EB 854790 90 mov P1,47H

00EE 12021C 91 lcall de5ms

00F1 854890 92 mov P1,48H

00F4 12021C 93 lcall de5ms

00F7 854990 94 mov P1,49H

00FA 12021C 95 lcall de5ms

00FD 854A90 96 mov P1,4AH

0100 12021C 97 lcall de5ms

0103 854B90 98 mov P1,4BH

0106 12021C 99 lcall de5ms

0109 854C90 100 mov P1,4CH

010C 12021C 101 lcall de5ms

010F 854D90 102 mov P1,4DH

0112 12021C 103 lcall de5ms

0115 854E90 104 mov P1,4EH

0118 12021C 105 lcall de5ms

011B 854F90 106 mov P1,4FH

011E 12021C 107 lcall de5ms

0121 C2B0 108 clr P3.0

0123 02022C 109 ljmp time

0126 E570 110 comp: mov A, 70H

0128 8571F0 111 mov B, 71H

012B C3 112 CLR C

012C C2D6 113 ClR AC

012E 95F0 114 subb A,B

0130 5001 115 JNC ncarry

0132 22 116 ret

0133 E571 117 ncarry: mov A,71H

0135 C570 118 xch A,70H

0137 C571 119 xch A,71H

0139 22 120 ret

121

122

013A 854070 123 bubble: mov 70H, 40H

013D 854171 124 mov 71H, 41H

0140 120126 125 lcall comp

0143 857040 126 mov 40H,70H

0146 857141 127 mov 41H,71H

0149 854170 128 mov 70H, 41H

014C 854271 129 mov 71H, 42H

014F 120126 130 lcall comp

0152 857041 131 mov 41H,70H

0155 857142 132 mov 42H,71H

0158 854270 133 mov 70H, 42H

015B 854371 134 mov 71H, 43H

015E 120126 135 lcall comp

0161 857042 136 mov 42H,70H

0164 857143 137 mov 43H,71H

0167 854370 138 mov 70H, 43H

016A 854471 139 mov 71H, 44H

016D 120126 140 lcall comp

0170 857043 141 mov 43H,70H

0173 857144 142 mov 44H,71H

0176 854470 143 mov 70H, 44H

0179 854571 144 mov 71H, 45H

017C 120126 145 lcall comp

017F 857044 146 mov 44H,70H

0182 857145 147 mov 45H,71H

0185 854570 148 mov 70H, 45H

0188 854671 149 mov 71H, 46H

018B 120126 150 lcall comp

018E 857045 151 mov 45H,70H

0191 857146 152 mov 46H,71H

0194 854670 153 mov 70H, 46H

0197 854771 154 mov 71H, 47H

019A 120126 155 lcall comp

019D 857046 156 mov 46H,70H

01A0 857147 157 mov 47H,71H

01A3 854770 158 mov 70H, 47H

01A6 854871 159 mov 71H, 48H

01A9 120126 160 lcall comp

01AC 857047 161 mov 47H,70H

01AF 857148 162 mov 48H,71H

01B2 854870 163 mov 70H, 48H

01B5 854971 164 mov 71H, 49H

01B8 120126 165 lcall comp

01BB 857048 166 mov 48H,70H

01BE 857149 167 mov 49H,71H

01C1 854970 168 mov 70H, 49H

01C4 854A71 169 mov 71H, 4AH

01C7 120126 170 lcall comp

01CA 857049 171 mov 49H,70H

01CD 85714A 172 mov 4AH,71H

01D0 854A70 173 mov 70H, 4AH

01D3 854B71 174 mov 71H, 4BH

01D6 120126 175 lcall comp

01D9 85704A 176 mov 4AH,70H

01DC 85714B 177 mov 4BH,71H

01DF 854B70 178 mov 70H, 4BH

01E2 854C71 179 mov 71H, 4CH

01E5 120126 180 lcall comp

01E8 85704B 181 mov 4BH,70H

01EB 85714C 182 mov 4CH,71H

01EE 854C70 183 mov 70H, 4CH

01F1 854D71 184 mov 71H, 4DH

01F4 120126 185 lcall comp

01F7 85704C 186 mov 4CH,70H

01FA 85714D 187 mov 4DH,71H

01FD 854D70 188 mov 70H, 4DH

0200 854E71 189 mov 71H, 4EH

0203 120126 190 lcall comp

0206 85704D 191 mov 4DH,70H

0209 85714E 192 mov 4EH,71H

020C 854E70 193 mov 70H, 4EH

020F 854F71 194 mov 71H, 4FH

0212 120126 195 lcall comp

0215 85704E 196 mov 4EH,70H

0218 85714F 197 mov 4FH,71H

021B 22 198 ret

199

200

201

021C D2B1 202 de5ms: SETB P3.1

021E 7F0A 203 mov R7, #0AH

0220 7EFA 204 aca: mov R6,#0FAH

0222 DEFE 205 aqui2: DJNZ R6, aqui2

0224 DFFA 206 DJNZ R7, aca

0226 C2B1 207 CLR P3.1

0228 22 208 RET

209

210

0229 020229 211 time2: ljmp time2

022C 7F40 212 time: mov R7,#40H

022E 7EFA 213 paca: mov R6,#0FAH

0230 7DFA 214 aca2: MOV R5,#0FAH

0232 DDFE 215 aqui3: DJNZ R5,aqui3

0234 DEFA 216 djnz R6,aca2

0236 DFF6 217 djnz R7,paca

0238 22 218 ret

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

ASSEMBLY COMPLETE, NO ERRORS FOUND, 1 WARNING

**5. Observaciones y Conclusiones**

|  |
| --- |
| **Conclusiones** |
| **Definitivamente es el bubble sort mas dificil que e programado, aunque se me hizo bastante divertido** |
| **El uso de los apuntadores e ir recorriendo en las direcciones para recorrer los 16 valores e ir haciendo** |
| **Los swaps, en los 16 recorrido que hacemos en total.** |
|  |
|  |

Promedios variados

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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. |

Uso de los timmers internos del micro controlador

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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:** | **12** | **Nombre de la práctica:** | | **Uso de los timmers internos del microcontrolador** | | | | |
| **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.** **Utilizar un temporizador interno del microcontrolador como contador o temporizador y así utilizar más funciones del microcontrolador**

**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, deberá utilizar un temporizador interno del  
microcontrolador y configurarlo en su formato de contador, para que de esta  
manera podamos contar eventos externos y mostrar esta cuenta en la pantalla LCD;  
en dicha pantalla se deberá mostrar una especie de proceso. En dicho proceso simulado,  
se estarán contando botellas, y cada que sean (x+10) botellas, se contara una caja. Donde x es el ultimo dígito de su registro. Todo lo anterior es para las personas que su ultimo dígito del registro sea par.  
Para las personas cuyo ultimo del registro sea impar deberá configurar el temporizador que medirá el tiempo que un botón este presionado. La temporalización se deberá ver el la lcd.

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

En esta pracitca me toco hacer un timmer, para eso activo todos los bits que tengo que prender para que el timmer 0 empiece a contar, y entonces pongo el timmer en 15535, y cada vez que se llena prende la flag, esto es cada 50000 ciclos maquinas, osea 50 ms, asi que ciclo para cada vez que se active la flag aumento un contador. El cual cuando llegue a 20 aumenta 1 segundo el tiempo que se debe mostrar (para eso usamos el Hexa a bcd), y entonces cuando llega a 60 segundos aumentamos en uno los minutos y ponemos en 0 los segundos.

**c) Código LST.**

temp PAGE 1

1

0000 020084 2 ljmp inicio

3

0003 208D03 4 ciclo: JB TF0,temp2

0006 020003 5 ljmp ciclo

6

0009 C28D 7 temp2: clr TF0

000B E8 8 mov A,R0

000C 14 9 DEC A

000D F8 10 mov R0,A

000E 6009 11 JZ cero

0010 758C3C 12 mov TH0, #3CH

0013 758AAF 13 mov TL0, #0AFH

0016 020003 14 ljmp ciclo

15

0019 E9 16 cero: mov A,R1

001A 04 17 inc A

001B F9 18 mov R1,A

001C 943B 19 subb A,#3BH

001E 500E 20 JNC nocarry

0020 120041 21 lcall imprimir

0023 7814 22 mov R0,#14H

0025 758C3C 23 mov TH0, #3CH

0028 758AAF 24 mov TL0, #0AFH

002B 020003 25 ljmp ciclo

26

002E EA 27 nocarry: mov A,R2

002F 04 28 inc A

0030 FA 29 mov R2,A

0031 7900 30 mov R1,#00H

0033 120041 31 lcall imprimir

0036 7814 32 mov R0,#14H

0038 758C3C 33 mov TH0, #3CH

003B 758AAF 34 mov TL0, #0AFH

003E 020003 35 ljmp ciclo

36

37

0041 759080 38 imprimir: mov P1,#80H

0044 120133 39 lcall de5ms

0047 8A60 40 mov 60H,R2

0049 1200F4 41 lcall bitabcd

004C E553 42 MOV A,53H

004E 4430 43 ORL A,#30H

0050 D2B0 44 setb P3.0

0052 F590 45 MOV P1,A

0054 120133 46 lcall de5ms

0057 E554 47 MOV A,54H

0059 4430 48 ORL A,#30H

005B F590 49 mov P1,A

005D 120133 50 lcall de5ms

0060 75903A 51 mov P1,#3AH

0063 120133 52 lcall de5ms

0066 C2B0 53 clr P3.0

0068 8960 54 MOV 60H,R1

006A 1200F4 55 lcall bitabcd

006D E553 56 MOV A,53H

006F 4430 57 ORL A,#30H

0071 D2B0 58 setb P3.0

0073 F590 59 MOV P1,A

0075 120133 60 lcall de5ms

0078 E554 61 MOV A,54H

007A 4430 62 ORL A,#30H

007C F590 63 mov P1,A

007E 120133 64 lcall de5ms

0081 C2B0 65 clr P3.0

0083 22 66 ret

67

0084 758909 68 inicio: mov TMOD, #09H

0087 D28C 69 setb TCON.4

70

0089 7814 71 pant: mov R0,#14H

008B C2B0 72 CLR P3.0

008D C2B1 73 CLR P3.1

008F C2B2 74 CLR P3.2

0091 756000 75 mov 60H, #00H

0094 7900 76 mov R1,#00H

0096 7A00 77 MOV R2,#00H

0098 759038 78 mov P1,#38H

009B 120133 79 lcall de5ms

009E 759038 80 mov P1,#38H

00A1 120133 81 lcall de5ms

00A4 759038 82 mov P1,#38H

00A7 120133 83 lcall de5ms

00AA 759038 84 mov P1,#38H

00AD 120133 85 lcall de5ms

00B0 75900C 86 mov P1,#0CH

00B3 120133 87 lcall de5ms

00B6 759001 88 mov P1,#01H

00B9 120133 89 lcall de5ms

00BC 759006 90 mov P1,#06H

00BF 120133 91 lcall de5ms

00C2 759080 92 mov P1,#80H

00C5 120133 93 lcall de5ms

00C8 D2B0 94 SETB P3.0

00CA 759030 95 mov P1,#30H

00CD 120133 96 lcall de5ms

00D0 759030 97 mov P1,#30H

00D3 120133 98 lcall de5ms

00D6 75903A 99 mov P1,#3AH

00D9 120133 100 lcall de5ms

00DC 759030 101 mov P1,#30H

00DF 120133 102 lcall de5ms

00E2 759030 103 mov P1,#30H

00E5 120133 104 lcall de5ms

00E8 C2B0 105 clr p3.0

00EA C3 106 clr C

00EB 758C3C 107 mov TH0, #3Ch

00EE 758AAF 108 mov TL0, #0AFh

00F1 020003 109 LJMP CICLO

110

111

00F4 755200 112 bitabcd: mov 52H,#00H

00F7 755300 113 mov 53H,#00H

00FA 755400 114 mov 54H,#00H

00FD E560 115 cien: mov A,60H

00FF C3 116 CLR C

0100 C2D6 117 CLR AC

0102 9464 118 subb A,#64H

0104 5005 119 JNC ncarry1

0106 2464 120 ADD A,#64H

0108 020116 121 LJMP diez

010B F560 122 ncarry1: MOV 60H,A

010D E552 123 MOV A,52H

010F 2401 124 ADD A,#01H

0111 F552 125 MOV 52H,A

0113 0200FD 126 LJMP cien

127

0116 E560 128 diez: mov A,60H

0118 C3 129 CLR C

0119 C2D6 130 CLR AC

011B 940A 131 subb A,#0AH

011D 5005 132 JNC ncarry2

011F 240A 133 ADD A,#0AH

0121 02012F 134 LJMP uno

0124 F560 135 ncarry2: MOV 60H,A

0126 E553 136 MOV A,53H

0128 2401 137 ADD A,#01H

012A F553 138 MOV 53H,A

012C 020116 139 LJMP diez

012F 856054 140 uno: MOV 54H,60H

0132 22 141 ret

142

143

144 de5ms:

0133 D2B1 145 SETB P3.1

0135 7F0A 146 mov R7, #0AH

0137 7EFA 147 aca: mov R6,#0FAH

0139 DEFE 148 aqui2: DJNZ R6, aqui2

013B DFFA 149 DJNZ R7, aca

013D C2B1 150 CLR P3.1

013F 22 151 RET

152

153

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

ASSEMBLY COMPLETE, NO ERRORS FOUND, 1 WARNING

**5. Observaciones y Conclusiones**

|  |
| --- |
| **Conclusiones** |
| *La verdad esta practica me costo bastante mas de lo que deberia, creo que mayormente debido a una confusión* |
| *que tuve, entre las interrupciones y el como conectar el botón para el temporizador* |
| *Pero fuera de eso se me hizo una practica interesante, una nueva función del microcontrolador.* |
|  |

Uso de las interrupciones internas

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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:** | **13** | **Nombre de la práctica:** | | **Uso de las interrupciones internas** | | | | |
| **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.** **Utilizar las interrupciones internas del microcontrolador como método de selección  
de programas diferentes para situaciones que se presentan de forma aleatoria**

**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, deberá realizar su practica utilizando interrupciones,  
las cuales serán, interrupciones externas. El programa que se deberá entregar  
será como sigue:  
  
  
Deberá tener un programa principal, el cual deberá mostrar su nombre en la pantalla LCD, pero  
de tal manera que cada letra de su nombre tarde un segundo en aparecer, y una  
vez que el nombre se haya completado, deberá borrar la pantalla y comenzar de  
nuevo a escribir su nombre, (con las reglas dadas anteriormente) y así  
sucesivamente por todo el tiempo que dure corriendo el programa.  
  
  
La interrupción externa cero, deberá mostrar su apellido paterno, dicho apellido a diferencia  
del nombre, deberá aparecer de una sola vez y, durara x segundos en la pantalla  
LCD, después, como es lógico, volverá al programa principal, pero si su nombre,  
en el programa principal, ya tenia algunas letras, estas deberán estar, una vez  
que se vuelva al programa principal, ya que el programa continuara donde se  
había quedado.  
  
  
La interrupción externa uno, deberá mostrar su apellido materno, dicho apellido a diferencia  
del nombre, deberá aparecer de una sola vez y, durará x segundos en la  
pantalla LCD, pero esta interrupción deberá tener mas prioridad que la  
interrupción cero; por lo tanto, deberá entrar a ejecutarse, aun cuando este en  
funcionamiento la interrupción cero. Por lo tanto, aquí surge la complejidad de  
tener que checar, el contexto de la solicitud de la interrupción uno, ya que,  
si entro después de la interrupción cero, va a tener que devolverse a  
terminarla, en otras palabras, deberá poner el apellido paterno, el tiempo  
restante que le quedaba a la interrupción cero, pero si entro desde el programa  
principal, solo deberá volver al programa principal, con las condiciones dichas  
en la interrupción cero.  
  
  
La calificación de cien se obtendrá, si se cumplen todas las condiciones; si solo se devuelve,  
sin observar ninguna condición, la calificación será de 60; si cumple con alguna de las condiciones, será de 80.  
  
  
NOTA 1: Solo se usará una fila para este producto integrador  
en la LCD  
  
  
NOTA2. Las personas que su ultimo dígito del registro sean  
par, escribirán de izquierda a derecha, las que terminen impar de derecha a  
izquierda, además las pares usaran el renglón inferior y las impares el renglón  
superior.

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

Para las interrupciones hacemos ORG 0003 H y ORG 0013 H y los mandamos a funciones separadas del apellido paterno y el apellido materno.

Para el estado por defecto de el codigo, que es mostrando el nombre lo que hago es mostrar las primeras n letras de mi nombre (en orden inverso porque me toca de derecha a izquierda) y luego me espero, limpio la pantalla y vuelvo a hacerlo con n+1, asi hasta que reinicio. El ir mostrando todo el prefijo es para que cuando haya una interrupción, vuelva a mostrar todo donde se quedo.

Para las interrupciones prácticamente lo mismo, nomas que en vez de mostrar un prefijo muestro toda la palabra, nada mas hay que tener cuidado de usar registros distintos para los contadores ya que asi cuando termine Briseño puede aparecer Buenrostro el tiempo que le faltaba.

**interrupcion PAGE 1**

**1 ORG 0000H**

**0000 020016 2 LJMP PANT**

**3 ORG 0003H**

**0003 02023B 4 LJMP caso1**

**5 ORG 0013H**

**0013 02028C 6 LJMP caso2**

**7**

**0016 C2B0 8 pant: CLR P3.0 ; inicializar la pantalla**

**0018 C2B1 9 CLR P3.1**

**10**

**001A 756000 11 mov 60H, #00H**

**001D 7900 12 mov R1,#00H**

**001F 7A00 13 MOV R2,#00H**

**0021 759038 14 mov P1,#38H**

**0024 1202D5 15 lcall de5ms**

**0027 759038 16 mov P1,#38H**

**002A 1202D5 17 lcall de5ms**

**002D 759038 18 mov P1,#38H**

**0030 1202D5 19 lcall de5ms**

**0033 759038 20 mov P1,#38H**

**0036 1202D5 21 lcall de5ms**

**0039 75900C 22 mov P1,#0CH**

**003C 1202D5 23 lcall de5ms**

**24**

**25 ;Habilitar interrupciones**

**003F 75A885 26 mov IE,#85H**

**0042 75B804 27 mov IP,#04H**

**28**

**29 inicio: ; Todo lo de sin interrupciones**

**30 ;E**

**0045 759001 31 mov P1,#01H**

**0048 1202D5 32 lcall de5ms**

**004B 759004 33 mov P1,#04H**

**004E 1202D5 34 lcall de5ms**

**0051 75908F 35 mov P1,#8FH**

**0054 1202D5 36 lcall de5ms**

**0057 D2B0 37 SETB P3.0**

**0059 759045 38 mov P1,#45H**

**005C 1202D5 39 lcall de5ms**

**005F C2B0 40 CLR P3.0**

**0061 12029C 41 lcall de1s**

**42**

**43 ; ME**

**44**

**0064 759001 45 mov P1,#01H**

**0067 1202D5 46 lcall de5ms**

**006A 759004 47 mov P1,#04H**

**006D 1202D5 48 lcall de5ms**

**0070 75908F 49 mov P1,#8FH**

**0073 1202D5 50 lcall de5ms**

**0076 D2B0 51 SETB P3.0**

**0078 75904D 52 mov P1,#4DH**

**007B 1202D5 53 lcall de5ms**

**007E 759045 54 mov P1,#45H**

**0081 1202D5 55 lcall de5ms**

**0084 C2B0 56 CLR P3.0**

**0086 12029C 57 lcall de1s**

**58 ;MME**

**59**

**0089 759001 60 mov P1,#01H**

**008C 1202D5 61 lcall de5ms**

**008F 759004 62 mov P1,#04H**

**0092 1202D5 63 lcall de5ms**

**0095 75908F 64 mov P1,#8FH**

**0098 1202D5 65 lcall de5ms**

**009B D2B0 66 SETB P3.0**

**009D 75904D 67 mov P1,#4DH**

**00A0 1202D5 68 lcall de5ms**

**00A3 75904D 69 mov P1,#4DH**

**00A6 1202D5 70 lcall de5ms**

**00A9 759045 71 mov P1,#45H**

**00AC 1202D5 72 lcall de5ms**

**00AF C2B0 73 CLR P3.0**

**00B1 12029C 74 lcall de1s**

**75**

**76 ;AMME**

**77**

**00B4 759001 78 mov P1,#01H**

**00B7 1202D5 79 lcall de5ms**

**00BA 759004 80 mov P1,#04H**

**00BD 1202D5 81 lcall de5ms**

**00C0 75908F 82 mov P1,#8FH**

**00C3 1202D5 83 lcall de5ms**

**00C6 D2B0 84 SETB P3.0**

**00C8 759041 85 mov P1,#41H**

**00CB 1202D5 86 lcall de5ms**

**00CE 75904D 87 mov P1,#4DH**

**00D1 1202D5 88 lcall de5ms**

**00D4 75904D 89 mov P1,#4DH**

**00D7 1202D5 90 lcall de5ms**

**00DA 759045 91 mov P1,#45H**

**00DD 1202D5 92 lcall de5ms**

**00E0 C2B0 93 CLR P3.0**

**00E2 12029C 94 lcall de1s**

**95**

**96 ;NAMME**

**97**

**00E5 759001 98 mov P1,#01H**

**00E8 1202D5 99 lcall de5ms**

**00EB 759004 100 mov P1,#04H**

**00EE 1202D5 101 lcall de5ms**

**00F1 75908F 102 mov P1,#8FH**

**00F4 1202D5 103 lcall de5ms**

**00F7 D2B0 104 SETB P3.0**

**00F9 75904E 105 mov p1,#4EH**

**00FC 1202D5 106 lcall de5ms**

**00FF 759041 107 mov P1,#41H**

**0102 1202D5 108 lcall de5ms**

**0105 75904D 109 mov P1,#4DH**

**0108 1202D5 110 lcall de5ms**

**010B 75904D 111 mov P1,#4DH**

**010E 1202D5 112 lcall de5ms**

**0111 759045 113 mov P1,#45H**

**0114 1202D5 114 lcall de5ms**

**0117 C2B0 115 CLR P3.0**

**0119 12029C 116 lcall de1s**

**117**

**118 ;UNAMME**

**119**

**011C 759001 120 mov P1,#01H**

**011F 1202D5 121 lcall de5ms**

**0122 759004 122 mov P1,#04H**

**0125 1202D5 123 lcall de5ms**

**0128 75908F 124 mov P1,#8FH**

**012B 1202D5 125 lcall de5ms**

**012E D2B0 126 SETB P3.0**

**0130 759055 127 mov P1,#55H**

**0133 1202D5 128 lcall de5ms**

**0136 75904E 129 mov p1,#4EH**

**0139 1202D5 130 lcall de5ms**

**013C 759041 131 mov P1,#41H**

**013F 1202D5 132 lcall de5ms**

**0142 75904D 133 mov P1,#4DH**

**0145 1202D5 134 lcall de5ms**

**0148 75904D 135 mov P1,#4DH**

**014B 1202D5 136 lcall de5ms**

**014E 759045 137 mov P1,#45H**

**0151 1202D5 138 lcall de5ms**

**0154 C2B0 139 CLR P3.0**

**0156 12029C 140 lcall de1s**

**141**

**142 ;EUNAMME**

**143**

**0159 759001 144 mov P1,#01H**

**015C 1202D5 145 lcall de5ms**

**015F 759004 146 mov P1,#04H**

**0162 1202D5 147 lcall de5ms**

**0165 75908F 148 mov P1,#8FH**

**0168 1202D5 149 lcall de5ms**

**016B D2B0 150 SETB P3.0**

**016D 759045 151 mov P1,#45H**

**0170 1202D5 152 lcall de5ms**

**0173 759055 153 mov P1,#55H**

**0176 1202D5 154 lcall de5ms**

**0179 75904E 155 mov p1,#4EH**

**017C 1202D5 156 lcall de5ms**

**017F 759041 157 mov P1,#41H**

**0182 1202D5 158 lcall de5ms**

**0185 75904D 159 mov P1,#4DH**

**0188 1202D5 160 lcall de5ms**

**018B 75904D 161 mov P1,#4DH**

**018E 1202D5 162 lcall de5ms**

**0191 759045 163 mov P1,#45H**

**0194 1202D5 164 lcall de5ms**

**0197 C2B0 165 CLR P3.0**

**0199 12029C 166 lcall de1s**

**167**

**168 ;LEUNAMME**

**019C 759001 169 mov P1,#01H**

**019F 1202D5 170 lcall de5ms**

**01A2 759004 171 mov P1,#04H**

**01A5 1202D5 172 lcall de5ms**

**01A8 75908F 173 mov P1,#8FH**

**01AB 1202D5 174 lcall de5ms**

**01AE D2B0 175 SETB P3.0**

**01B0 75904C 176 mov P1,#4CH**

**01B3 1202D5 177 lcall de5ms**

**01B6 759045 178 mov P1,#45H**

**01B9 1202D5 179 lcall de5ms**

**01BC 759055 180 mov P1,#55H**

**01BF 1202D5 181 lcall de5ms**

**01C2 75904E 182 mov p1,#4EH**

**01C5 1202D5 183 lcall de5ms**

**01C8 759041 184 mov P1,#41H**

**01CB 1202D5 185 lcall de5ms**

**01CE 75904D 186 mov P1,#4DH**

**01D1 1202D5 187 lcall de5ms**

**01D4 75904D 188 mov P1,#4DH**

**01D7 1202D5 189 lcall de5ms**

**01DA 759045 190 mov P1,#45H**

**01DD 1202D5 191 lcall de5ms**

**01E0 C2B0 192 CLR P3.0**

**01E2 12029C 193 lcall de1s**

**194**

**195 ; reiniciar**

**196**

**01E5 020045 197 LJMP inicio**

**198**

**199**

**200**

**201 app: ; Imprime el apellido paterno (BUENROSTRO)**

**01E8 759001 202 mov P1,#01H**

**01EB 1202D5 203 lcall de5ms**

**01EE 759004 204 mov P1,#04H**

**01F1 1202D5 205 lcall de5ms**

**01F4 75908F 206 mov P1,#8FH**

**01F7 1202D5 207 lcall de5ms**

**01FA D2B0 208 SETB P3.0**

**209**

**01FC 75904F 210 mov P1,#4FH**

**01FF 1202D5 211 lcall de5ms**

**0202 759052 212 mov P1,#52H**

**0205 1202D5 213 lcall de5ms**

**0208 759054 214 mov P1,#54H**

**020B 1202D5 215 lcall de5ms**

**020E 759053 216 mov P1,#53H**

**0211 1202D5 217 lcall de5ms**

**0214 75904F 218 mov P1,#4FH**

**0217 1202D5 219 lcall de5ms**

**021A 759052 220 mov P1,#52H**

**021D 1202D5 221 lcall de5ms**

**0220 75904E 222 mov P1,#4EH**

**0223 1202D5 223 lcall de5ms**

**0226 759045 224 mov P1,#45H**

**0229 1202D5 225 lcall de5ms**

**022C 759055 226 mov P1,#55H**

**022F 1202D5 227 lcall de5ms**

**0232 759042 228 mov P1,#42H**

**0235 1202D5 229 lcall de5ms**

**0238 C2B0 230 CLR P3.0**

**023A 22 231 ret**

**232**

**233**

**234 caso1: ;Aqui hacemos el tiempo de 1s de la interrupcion 0**

**023B 756607 235 mov 66h, #07H**

**023E D56601 236 paca3: DJNZ 66h, aqui3**

**0241 32 237 reti**

**0242 1201E8 238 aqui3: lcall app**

**0245 1202C2 239 lcall de100ms2**

**0248 02023E 240 ljmp paca3**

**241**

**242**

**243 apm: ; Imprime el apellido materno (BRISENO)**

**024B 759001 244 mov P1,#01H**

**024E 1202D5 245 lcall de5ms**

**0251 759004 246 mov P1,#04H**

**0254 1202D5 247 lcall de5ms**

**0257 75908F 248 mov P1,#8FH**

**025A 1202D5 249 lcall de5ms**

**025D D2B0 250 SETB P3.0**

**025F 75904F 251 mov P1, #4FH**

**0262 1202D5 252 lcall de5ms**

**0265 75904E 253 mov P1, #4EH**

**0268 1202D5 254 lcall de5ms**

**026B 759045 255 mov P1, #45H**

**026E 1202D5 256 lcall de5ms**

**0271 759053 257 mov P1, #53H**

**0274 1202D5 258 lcall de5ms**

**0277 759049 259 mov P1, #49H**

**027A 1202D5 260 lcall de5ms**

**027D 759052 261 mov P1, #52H**

**0280 1202D5 262 lcall de5ms**

**0283 759042 263 mov P1, #42H**

**0286 1202D5 264 lcall de5ms**

**0289 C2B0 265 CLR P3.0**

**028B 22 266 ret**

**267**

**268 caso2: ;Aqui hacemos el tiempo de 1s de la interrupcion 1**

**028C 756507 269 mov 65h, #07H**

**028F D56501 270 paca4: DJNZ 65h, aqui4**

**0292 32 271 reti**

**0293 12024B 272 aqui4: lcall apm**

**0296 1202AF 273 lcall de100ms**

**0299 02028F 274 ljmp paca4**

**275**

**276**

**277 de1s:**

**029C 754208 278 mov 42h,#08H**

**029F 7541FA 279 paca1: mov 41h,#0FAH**

**02A2 7540FA 280 aca1: MOV 40h,#0FAH**

**02A5 D540FD 281 aqui1: DJNZ 40h,aqui1**

**02A8 D541F7 282 djnz 41h,aca1**

**02AB D542F1 283 djnz 42h,paca1**

**02AE 22 284 ret**

**285**

**286 de100ms:**

**02AF 756201 287 mov 62h,#01H**

**02B2 7561FA 288 paca5: mov 61h,#0FAH**

**02B5 7560FA 289 aca5: MOV 60h,#0FAH**

**02B8 D560FD 290 aqui5: DJNZ 60h,aqui5**

**02BB D561F7 291 djnz 61h,aca5**

**02BE D562F1 292 djnz 62h,paca5**

**02C1 22 293 ret**

**294**

**295 de100ms2:**

**02C2 757201 296 mov 72h,#01H**

**02C5 7571FA 297 paca6: mov 71h,#0FAH**

**02C8 7570FA 298 aca6: MOV 70h,#0FAH**

**02CB D570FD 299 aqui6: DJNZ 70h,aqui6**

**02CE D571F7 300 djnz 71h,aca6**

**02D1 D572F1 301 djnz 72h,paca6**

**02D4 22 302 ret**

**303**

**304**

**305 de5ms:**

**02D5 D2B1 306 SETB P3.1**

**02D7 7F0A 307 mov R7, #0AH**

**02D9 7EFA 308 aca2: mov R6,#0FAH**

**02DB DEFE 309 aqui2: DJNZ R6, aqui2**

**02DD DFFA 310 DJNZ R7, aca2**

**02DF C2B1 311 CLR P3.1**

**02E1 22 312 RET**

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

**ASSEMBLY COMPLETE, NO ERRORS FOUND, 1 WARNING**

**5. Observaciones y Conclusiones**

|  |
| --- |
| **Conclusiones** |
| Esta practica, me salio bastante rápido, nada más tenia un error que usando algo que utilizaba en practicas anteriores |
| (un led auxiliar) prendia el 3.2 lo que hacia que entrara siempre en la interrupción de BUENROSTRO, pero de ahí |
| En más siento que es algo muy útil de los microcontroladores para poder tener mas interacción con el mundo |
| Exterior. |