

## **Práctica 1: Microcontrolador MC9S08QG8**

Objetivo: Familiarizar al alumnos con el dispositivo microcontrolador MC9S08QG8 y la tarjeta de evaluación DEMO9S08 así como con el ambiente de programación CODEWARRIOR de Freescale.

Material y Equipo:

Tarjeta de evaluación DEMO9S08

Programa CODEWARRIOR v5.1

Computadora personal

Desarrollo:

- 1.- Hacer una descripción de los recursos del microcontrolador MC9S08QG8, tales como puertos, temporizadores, entre otros.
- 2.- Hacer una descripción la tarjeta demo en relación a sus recursos para prueba del micro controlador MC9S08QG8.
- 3.- Descripción del ambiente de programación en el programa CODEWARRIOR (en lengua ensamblador y lenguaje C).
- 4.- Realizar pruebas para verificar el funcionamiento de la tarjeta de evaluación con los recursos de la tarjeta.

## **Practica 2: Manipulación de registros y de datos en memoria**

Objetivo1: Elaborar un programa en ensamblador, para el microcontrolador MC9S08QG8, que haga uso de los registros internos del CPU para manipular datos en memoria. Este debe consistir en el desplazamiento de datos contenidos en un segmento previamente establecido a otro igualmente establecido por el programador.

Objetivo2: Haciendo uso del programa generado en el inciso anterior , realice los cambios pertinentes al programa de manera que los datos transferidos al segmento destino queden almacenados en forma ascendente.

Material y Equipo:

Tarjeta de evaluación DEMO9S08

Programa CODEWARRIOR v5.1

Computadora personal

Desarrollo:

- 1.- Establecer un segmento fuente de mínimo 20 localidades de memoria y un segmento destino del mismo tamaño. Los datos contenidos en dichas localidades pueden ser de 01h a FFh.

- 2.- El programa debe finalizar el proceso de transferencia de datos al encontrar un cero.
- 3.- Analice aspectos relacionados con el tamaño de los segmentos fuente y destino, y con ausencia de ceros.
- 4.- Realizar los cambios pertinentes para que los datos del segmento fuente queden almacenados en orden ascendente en el segmento destino, ya sea durante la transferencia de éstos o después de que los datos ya hayan sido transferidos.

### **Practica 3: Puertos de entrada y salida**

Objetivo1: Elaborar un programa en ensamblador para el microcontrolador MC9S08QG8, que haga lectura de un dato presente en los de 4 bits menos significativos del puerto A, dicho valor se debe presentar en los dos nibbles que forman los 8bits del puerto B.

Objetivo2: Haciendo uso del arreglo de leds del programa anterior, elaborar un programa que realice un retardo de tiempo controlado y haga uso de este para generar una secuencia (definida por el programador) de encendido y apagado de en los leds.

Material y Equipo:

- 8 Resistencias de 330 ohms.
- 4 Resistencias de 4.7K ohms.
- 8 Leds.
- 4 Interruptores.
- Protoboard
- Fuente de alimentación.
- Tarjeta de evaluación DEMO9S08.
- Conector hembra de 32 terminales
- Programa CODEWARRIOR v5.1.
- Computadora personal.

Desarrollo:

- 1.- Durante la elaboración del programa, debe considerar puntos relacionados con:
  - Ubicar la dirección de los puertos en el mapa de memoria.
  - Configuración de registros de control de puertos.
- 2.- Realizar el armado y conexión de los 4 interruptores en las terminales PTA1,... y PTA4, estos deben ser conectados de manera que aseguren los niveles lógicos de 0 y 1 en el microcontrolador.
- 3.- Realizar el armado y conexión de los leds con su respectivas resistencias limitadoras de corriente en las terminales PTB0 a la PTB7.

4.- Desconecte los interruptores y elabore una subrutina de retardo de tiempo y elabore una rutina que haga uso del retardo para crear una secuencia de encendido y apagado a conveniencia del programador pero diferente de los demás equipos.

## **Practica 4: Aritmética de punto fijo y punto flotante.**

Objetivo: Elaborar dos programas en lenguaje C para el microcontrolador MC9S08QG8, que calculen la raíz cuadrada de un número positivo. El primer programa debe hacer uso sólo de variables y datos de tipo entero y el segundo hará uso de flotantes. el resultado debe ser almacenado en una dirección de memoria definida por el usuario.

Material y Equipo:

Tarjeta de evaluación DEMO9S08

Programa CODEWARRIOR v5.1

Computadora personal

Desarrollo:

Un método utilizado para el cálculo de la raíz cuadrada de un número positivo es el que a continuación se presenta:

$$y(n) = \frac{1}{2} \left[ y(n-1) + \frac{x(n)}{y(n-1)} \right]$$

Donde  $n$  = es el número de iteración (0, 1, 2, 3, 4, 5, ...).

$y(n)$  = valor presente de la raíz cuadrada.

$x(n)$  = valor al que se le quiere extraer la raíz cuadrada (radicando).

$y(n-1)$  = valor pasado de la raíz cuadrada.

$y(-1)$  = condición inicial o estima inicial de la raíz cuadrada.

Este algoritmo es un método iterativo cuyo valor  $y(n)$  converge a la raíz cuadrada de  $x(n)$  a medida que  $n$  crece. El valor  $y(-1)$  es una condición inicial que debe cumplir con ser un número mayor que cero; existen algunos criterios para establecerla, pero para fines de esta práctica tomaremos por condición inicial  $y(-1) = x(n)/2$ .

1.- El primer programa debe considerar los siguientes aspectos:

- Uso de datos y variables de tipo entero.
- Aplicar cálculos con aritmética de punto fijo.
- Precisión de los cálculos en centésimas de unidad.

2.- El usuario escribirá directamente en una localidad de memoria del MC, definida por el programador, el valor del radicando. Se evaluará y observará el resultado de cada iteración,

a través del uso de condiciones de "Breakpoints" durante la prueba y depuración del programa.

3.- Se harán pruebas con diversas cantidades hasta llegar al límite máximo de cálculo que provoque alguna condición de error.

4.- El segundo programa debe considerar los siguientes aspectos:

- Uso de datos y variables de tipo flotante.

5.- Al igual que en el programa anterior, el usuario escribirá el radicando en una localidad de memoria del MC. Se evaluará y observará el resultado de cada iteración, a través del uso de condiciones de "Breakpoints" durante la prueba y depuración del programa.

6.- Se harán pruebas con diversas cantidades hasta llegar al límite máximo de cálculo que provoque alguna condición de error.

7.- Hacer una comparación entre el código maquina generado en el primer y segundo programa y observar las diferencias entre el tamaño del código ensamblado, así mismo resaltar las ventajas y desventajas entre ambos programas.

## **Practica 5: Comparador analógico.**

Objetivo: Elaborar un programa en el ensamblador del microcontrolador MC9S08QG8, que configure y pruebe el funcionamiento del comparador analógico con que cuenta este dispositivo, a través de la comparación de dos voltajes aplicados en las terminales PTA0 y PTA1.

Material y Equipo:

Protoboard

Conector hembra de 32 terminales

2 Potenciómetros

Fuente de alimentación.

Tarjeta de evaluación DEMO9S08

Programa CODEWARRIOR v5.1

Computadora personal

Desarrollo:

1.- El programa a elaborar, debe realizar en primer lugar una configuración del registro de control y estado del módulo comparador. Dicho registro activa o desactiva el módulo, almacena el estado de la salida del comparador en función de la relación de los voltajes aplicados a éste.

2.- El programa debe verificar constante mente el estado de la salida del comparador y el valor de esta debe ser presentado en los leds de la tarjeta de evaluación. Cuando el voltaje sea mayor se deben encender ambos leds y al ser menor se deben apagar.

## **Practica 6: Convertidor A/D.**

Objetivo: Elaborar un programa en lenguaje C del microcontrolador MC9S08QG8, que configure y pruebe el funcionamiento del convertidos A/D de dispositivo a través de la digitalización de una señal analógica y de la visualización de sus muestras.

Material y Equipo:

Protoboard

Conector hembra de 32 terminales

8 Resistencias de 330 ohms.

8 Leds.

Generador de funciones.

Osciloscopio.

Fuente de alimentación.

Tarjeta de evaluación DEMO9S08

Programa CODEWARRIOR v5.1

Computadora personal

Desarrollo:

- 1.- El programa a elaborar, debe realizar la configuración de los registros de control y estado del módulo CA/D. El programador erigirá un canal de los disponibles y lo habilitará en modo de conversión continua y con resolución de 8bits.
- 2.- El programa debe mostrar constantemente el valor presente de la conversión, a través de la conexión de los 8 leds colocados en las terminales de puerto B.
- 3.- Configure el generador de funciones para que entregue una señal de 1Hz, con 0.5v de amplitud y con un nivel de voltaje de offset equivalente a la mitad del alcance dinámico del CA/D.
- 4.- Ejecutar el programa y conectar la señal del generador a la terminal correspondiente al canal configurado y observe los leds.
- 5.- Realizar variaciones en cuanto a frecuencia y amplitud con las debidas precauciones y comente.

## **Practica 7: Interrupciones y temporizador.**

Objetivo: Elaborar un programa en lenguaje C del microcontrolador MC9S08QG8, que configure y pruebe el funcionamiento del módulo temporizador haciendo uso de la interrupción por desbordamiento de este módulo.

Material y Equipo:

Protoboard

Conector hembra de 32 terminales

8 Resistencias de 330 ohms.

8 Leds.

Tarjeta de evaluación DEMO9S08

Programa CODEWARRIOR v5.1

Computadora personal

Desarrollo:

- 1.- El programa a elaborar, debe realizar la configuración de los registros de control y estado del módulo temporizador, así mismo se debe configurar la interrupción por desbordamiento de éste.
- 2.- El programa principal puede ser un ciclo o un conjunto de instrucciones que se estén realizando de manera permanente para que dé tiempo a que se presente la interrupción del temporizador.
- 3.- La subrutina de servicio de interrupción, será una un segmento de código breve que cambie uno o varios parámetros del programa principal, a fin de ver el efecto de la interrupción.
- 4.- Auxíliese del arreglo de leds para interactuar con el exterior cuando se este ejecutando el programa o cuando se esté atendiendo la interrupción.

## **Practica 8: Comunicación SCI.**

Objetivo: Elaborar un programa en lenguaje C del microcontrolador MC9S08QG8, que configure y pruebe el funcionamiento del módulo de comunicación SCI, transmitiendo datos (una cadena de caracteres o una secuencia de números) a una PC.

Material y Equipo:

Protoboard

Tarjeta de evaluación DEMO9S08

Programa CODEWARRIOR v5.1

Computadora personal

Desarrollo:

- 1.- El programa a elaborar, debe realizar la configuración de los registros de control y estado del módulo de comunicación SCI.
- 2.- El programa principal debe realizar un ciclo de transferencia de datos (caracteres o números) ubicados en un determinado segmento de memoria definido por el programador
- 3.- Al finalizar la transferencia el programa debe quedarse realizando un ciclo infinito

## **Practica 9: Medición de amplitud, frecuencia y fase de una señal analógica.**

Objetivo: Elaborar un programa en lenguaje C del microcontrolador MC9S08QG8, y un arreglo de hardware que permita medir los tres parámetros que caracterizan a una oscilación armónica (amplitud, frecuencia y fase), de manera que estos puedan ser visualizados en displays o una pantalla de LCD por elección del usuario.

Material y Equipo:

Protoboard.

Componentes varios.

Generador de funciones.

Osciloscopio.

Fuente de alimentación.

Tarjeta de evaluación DEMO9S08.

Programa CODEWARRIOR v5.1.

Computadora personal

Desarrollo:

- 1.- El programa a realizar debe presentar segmentos de configuración tanto del CA/D como del temporizador o de alguna terminal en específico que vaya a utilizar para medir la frecuencia.
- 2.- Configure el generador de funciones para que entregue una señal cuadrada de 1Hz, con 0.5v de amplitud y con un nivel de voltaje de offset equivalente a la mitad del alcance dinámico del CA/D.
- 3.- A través de una red de adelanto o de atraso de fase, provocaremos un cambio en la fase de la señal proveniente del generador.
- 4.- Los resultados de la medición de parámetros serán ubicados en un determinado segmento de memoria definido por el programador
- 5.- La medición debe hacerse de manera constante para así tener siempre se esten actualizado de los parámetros.

6.- A través de un interruptor se debe poder elegir entre que parámetro se va a desplegar en los display o LCD.