**Interrupciones**

Las interrupciones en los microcontroladores pueden ser de dos tipos: de hardware y de software.

Consisten en el cambio en la secuencia de ejecución de un programa cuando se da un determinado evento de hardware.

El microcontrolador ejecuta una rutina asociada con el evento de hardware implicado, conocida como "rutina de servicio".

Una vez que concluye la ejecución de la rutina de servicio, el microcontrolador debe retornar a la secuencia de ejecución que fue interrumpida.

**Ejemplos**

Para un determinado MCU, eventos de hardware podrán ser entre otros: que el puerto serie asíncrono haya recibido un byte y éste esté listo para ser leído, que se de un flanco de bajada en un determinado bit de puerto, etc.

En esta exposición, a los diversos eventos de hardware que pueden hacer que se de una interrupción, se les denomina INSTANCIAS DE INTERRUPCIÓN.

**Requerimiento**

Cuando una instancia de interrupción se presenta, se dice que se tiene un requerimiento de interrupción (RI).

En resumen, las interrupciones de hardware en los microcontroladores permiten manejar eventos de hardware de manera eficiente, ejecutando rutinas de servicio asociadas a cada evento. Al presentarse una instancia de interrupción, se tiene un requerimiento de interrupción que el microcontrolador debe atender. Es importante conocer los eventos de hardware que pueden generar interrupciones en un MCU para programar adecuadamente su manejo.

El MC9S08SH32 cuenta con 33 vectores de interrupción, numerados del 0 al 32.

Cada vector de interrupción corresponde a un evento específico y tiene asociada una rutina de servicio.

**Prioridad**

Las interrupciones en el MC9S08SH32 tienen prioridad predecible.

Las interrupciones de menor número tienen mayor prioridad que las de mayor número.

Las interrupciones de igual número tienen prioridad según el orden en que ocurren.

**Habilitación y deshabilitación**

Las interrupciones se habilitan y deshabilitan mediante el registro CCR (Condition Code Register).

Para habilitar las interrupciones, se debe establecer el bit I del registro CCR en 1.

Para deshabilitar las interrupciones, se debe establecer el bit I del registro CCR en 0.

**Ejemplo**

Supongamos que se desea manejar la interrupción correspondiente al vector 12, asociado al evento de recepción de datos por el puerto serie asíncrono.

Para habilitar esta interrupción, se debe establecer el bit PEIE del registro SCIxC2 en 1 y el bit RIE del registro SCIxCR2 en 1.

Se debe también establecer la dirección de la rutina de servicio correspondiente en la tabla de vectores de interrupción.

Cuando se reciba un byte por el puerto serie asíncrono, se generará una instancia de interrupción y se ejecutará la rutina de servicio asociada al vector 12.

En resumen, el MC9S08SH32 cuenta con 33 vectores de interrupción que permiten manejar eventos específicos de manera eficiente. Las interrupciones tienen prioridad predecible y se habilitan y deshabilitan mediante el registro CCR. Es importante conocer cómo se manejan las interrupciones en este MCU para programar adecuadamente su uso en aplicaciones específicas.

PASO 1: Guarda en el stack cinco bytes en el siguiente orden:

* PCL (byte bajo del PC de retorno)
* PCH (byte alto del PC de retorno)
* Registro X
* Registro A
* Registro CCR

PASO 2: El bit “I” se pone en uno lógico

PASO 3: Lee el valor del vector de interrupción asociado, esto desde el par de direcciones, propias de la instancia de interrupción implicada.

PASO 4: Salta a la dirección denotada por el vector de interrupción capturado en el paso anterior, esto es, inicia la ejecución de la rutina de servicio asociada con la instancia de interrupción para la cual se está respondiendo.

Tiempo de latencia:

11Tb<=Tlat<=22Tb. Donde Tb es el periodo del reloj de bus. Por ejemplo, si éste

es de 20 MHz, Tlat podrá estar entre 0.55 µs y 1.1 µs

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamenteTexto

Descripción generada automáticamenteTexto

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Manejo LCD conectado al MCU**

Esta tarjeta cuenta con un módulo LCD alfanumérico de 2 líneas por 16 caracteres, conectado al puerto C del MCU.

**Configuración**

Para configurar el LCD, se debe inicializar el puerto C del MCU como salida.

Se debe también configurar el LCD en modo de 8 bits, estableciendo los bits correspondientes del registro de control.

Luego se deben enviar una serie de comandos al LCD para configurar su funcionamiento, como establecer el número de líneas, el tamaño de los caracteres y la posición del cursor.

**Escritura**

Para escribir en el LCD, se deben enviar primero los datos al puerto C del MCU.

Luego se debe establecer la señal RS del LCD en 1 para indicar que se van a enviar datos y no comandos.

Se debe establecer también la señal E del LCD en 1 para indicar que se va a realizar una escritura.

Finalmente, se debe establecer la señal E del LCD en 0 para completar la escritura.

**Ejemplo**

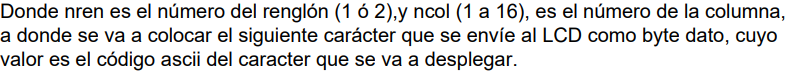
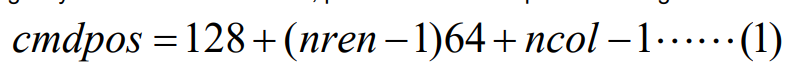
¡Supongamos que se desea escribir la cadena “Hola mundo!" en la primera línea del LCD.

Primero se deben enviar los caracteres correspondientes al puerto C del MCU: 'H', 'o', 'l', 'a', ' ', 'm', 'u', 'n', 'd', 'o', '!', y los bits correspondientes de control (RS=1, E=1, E=0).

Para establecer la posición del cursor, se debe enviar el comando correspondiente al LCD, por ejemplo: 0x80 para la posición 0 de la primera línea.

Luego se debe enviar la cadena de caracteres como se explicó anteriormente.

En resumen, para manejar el LCD conectado al puerto C del MCU presente en la tarjeta FACIL\_08SH se deben configurar el puerto como salida y el LCD en modo de 8 bits, y luego enviar comandos y datos al LCD a través del puerto C del MCU. Es importante conocer la configuración y los comandos necesarios para utilizar adecuadamente el LCD en aplicaciones específicas.



**Interrupciones periódicas**

Las interrupciones periódicas son aquellas que se generan en intervalos regulares de tiempo.

Estas interrupciones son muy útiles en aplicaciones que requieren un control preciso del tiempo, como en sistemas de control de procesos y en aplicaciones de medición de tiempo.

Las interrupciones periódicas se pueden generar a través de una variedad de fuentes, como un temporizador interno del microcontrolador o un cristal de cuarzo externo.

**Configuración**

Para configurar interrupciones periódicas, se debe primero seleccionar la fuente de interrupción apropiada.

Luego, se deben configurar los registros del temporizador para establecer el intervalo de tiempo entre interrupciones.

Es importante tener en cuenta la frecuencia del reloj del microcontrolador y el valor de preescala que se va a utilizar para calcular el tiempo de interrupción.

**Rutina**

Una vez que se ha configurado la interrupción periódica, se debe crear una rutina de interrupción para manejarla.

Esta rutina se ejecutará cada vez que se genere una interrupción y se encargará de realizar las tareas necesarias en ese momento.

Es importante tener en cuenta que la rutina de interrupción debe ser lo más corta y eficiente posible, para no afectar el funcionamiento normal del programa.

**Ejemplo**

Supongamos que se desea medir la temperatura de un horno cada 5 segundos.

Se puede configurar un temporizador interno del microcontrolador para generar una interrupción cada 5 segundos.

Se puede crear una rutina de interrupción que lea la temperatura del horno y la guarde en una variable.

Luego, se puede utilizar esta variable en el programa principal para realizar acciones específicas, como encender un ventilador para enfriar el horno si la temperatura es demasiado alta.

En resumen, las interrupciones periódicas son una herramienta importante en el manejo de microcontroladores para el control preciso del tiempo. La configuración adecuada de las interrupciones y la creación de una rutina de interrupción eficiente son fundamentales para su correcto uso en aplicaciones específicas.

Texto, Carta

Descripción generada automáticamente

Texto, Carta

Descripción generada automáticamenteTexto

Descripción generada automáticamenteTexto, Carta

Descripción generada automáticamenteCaptura de pantalla de un celular con texto

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto

Descripción generada automáticamenteTexto

Descripción generada automáticamente