

Taller 9 – E/S y manejo de Interrupciones

Objetivos

- Implementar manejadores de dispositivos.
- Realizar accesos a entrada / salida.

Introducción

Las interrupciones son cambios en el flujo de ejecución no causados por otros programas, sino por otros eventos, usualmente relacionados con la entrada/salida. Una vez finalizada la interrupción el flujo de ejecución vuelve al punto donde se detuvo el programa interrumpido.

Responder las siguientes preguntas

- Nombre y defina brevemente los esquemas de entada/salida que conoce.
- ¿En qué cambia la resolución de problemas relacionados con interrupciones, el hecho de usar una máquina dedicada?

Interrupciones y Puertos de E/S

La empresa Fit-Flop le ha encomendado la programación de un microcontrolador para su nueva línea de caminadores domésticos (*tipo D*). El caminador se compone de una cinta potenciada por un motor, varios botones para controlar el funcionamiento y un display donde se despliega información, por ejemplo el tiempo de caminata o la velocidad alcanzada.

El caminador dispone de 3 botones para controlar la velocidad. Al presionar cada botón, la velocidad cambia a 11, 5 o 0 km/h, respectivamente. Los 3 botones son accesibles en los bits 2 al 0 del puerto de entrada/salida de solo lectura de un byte, BOTONES. El contenido del puerto es eliminado una vez leído. Al presionar un botón se activa la rutina de interrupción `hay_boton()`. Se asume que como máximo un único bit de BOTONES puede estar encendido en cada lectura.

El motor se enciende al escribir un 1 en el bit menos significativo del byte de entrada/salida de solo escritura MOTOR y se apaga colocando un 0. En los 7 bits más significativos de dicho puerto se debe indicar la velocidad objetivo, en décimas de kilómetros por hora (una velocidad de 11 km/h, se indica como 110).

Para medir la velocidad de funcionamiento, la cinta dispone de conectores cada 10 cm, los cuales al pasar frente a un sensor (incorporado en la cara interna de la cinta), disparan la interrupción `cinta()`. Si no se alcanza la velocidad deseada pasados 30 segundos desde la presión de uno de los botones, se considera que la cinta precisa lubricación. Esto se debe informar encendiendo un bit de alarma.

El caminador posee un display que debe mostrar el tiempo transcurrido desde que el motor se puso en funcionamiento y la velocidad de movimiento de la cinta. El display se debe actualizar una vez por segundo, escribiendo en dos puertos de entrada/salida de solo escritura de 16 bits, DISPLAY_HI y DISPLAY_LO, con el siguiente formato:

- El bit 15 de DISPLAY_HI controla el led de alarma
- El bit 14 de DISPLAY_HI indica si el motor está en funcionamiento
- En los bits 13...7 de DISPLAY_HI se indica la velocidad de la cinta durante el último segundo, en décimas de km/h (por ejemplo, una velocidad de 9,5km/h se indica como 95)
- Los bits 6...0 de DISPLAY_HI no se utilizan
- En DISPLAY_LO se indica el tiempo transcurrido en segundos

Se dispone de un timer que interrumpe con frecuencia de 1hz, invocando a la rutina `tiempo()`.

Se pide: implementar en un lenguaje de alto nivel (preferentemente C), todas las rutinas necesarias para el correcto funcionamiento del sistema, considerando que la máquina está dedicada a la tarea.

Nota: Se puede utilizar la siguiente aproximación: $1\text{km/h} \approx 30\text{cm/s}$

Nota 2: El usuario puede querer usar el caminador sin encender el motor, por lo que se debe calcular e imprimir la velocidad en todo momento. El tiempo de uso solamente se debe imprimir cuando el motor pasa de apagado a prendido.

Nota 3: Es importante terminar de resolver este taller ya que se utilizará como insumo en el próximo taller.