
	<p style="text-align: center;">UNIVERSIDAD DE COSTA RICA ESCUELA DE INGENIERÍA ELÉCTRICA</p>	
<p style="text-align: center;">IE-623 MICROPROCESADORES Prof. Geovanny Delgado</p>		

TAREA #7 RELOJ DE TIEMPO REAL DS1307

Se debe realizar un programa para implementar un reloj que indique horas y minutos en la pantalla de 7 segmentos. Los valores a desplegar serán leídos del RTC DS1307 implementando las rutinas de acceso por IIC.

El DS1307 se deberá ajustar a los valores colocados en T_Write_RTC cada vez que se presione el botón PH0 y será leído una vez por segundo con base en la interrupción RTI, almacenando sus registros en T_Read_RTC.



Adicionalmente el reloj manejará una Alarma de Despertador. Para ello se debe crear una constante tipo Word llamada ALARMA que tendrá la hora donde debe activarse una alarma. El formato de este Word será MM:HH, donde MM son los minutos y HH son las horas, a las que debe activarse la alarma. Estos valores deben ser almacenados en BCD. El programa principal deberá validar constantemente si la hora actual es igual a ALARMA, de ser así deberá activar la interrupción OC5. Esta interrupción se deshabilitará cuando se presione el botón PH1 de la Dragon 12.

Esta tarea hará uso de las subrutinas de multiplexación de pantallas de la Tarea de Pantallas BCD_7SEG y OC4_ISR, con la salvedad de que no se llamará a la subrutina BIN_BCD pues los valores a desplegar ya están en BCD. La subrutina OC4 solo llama a la Subrutina BCD_7SEG pasando los parámetros BCD2 y BCD1. En la subrutina BCD_7SEG el LSB de los segundos de la T_Read_RTC debe copiarse al MSB de DISP2 y DISP3, con el fin de que los leds del punto decimal de estos dos dígitos parpadeen dando la indicación de los segundos.

De igual manera se deben incluir las subrutinas para el manejo del LCD (Cargar_LCD, Delay, SEND_COMMAND, SEND_DATA) para que al inicio del programa principal se envíe el siguiente mensaje a la pantalla LCD:



SUBROUTINA RTI_ISR (INCIO DE LECTURA): La interrupción RTI se debe programar con un tiempo de interrupción de 50 mS. Cuando se cumpla 1 segundo se deberá activar hacer RTC_RW=1 e iniciar la lectura del RTC en el DS1307, enviando un comando de START al IIC y el calling address (WRITE). Adicionalmente en esta

	<p style="text-align: center;">UNIVERSIDAD DE COSTA RICA ESCUELA DE INGENIERÍA ELÉCTRICA</p>	
<p style="text-align: center;">IE-623 MICROPROCESADORES Prof. Geovanny Delgado</p>		

subrutina cada segundo se deben copiar, del arreglo T_Read_RTC, los minutos a BCD1 y las horas a BCD2.

SUBROUTINA PTH_ISR: Esta subrutina será la encargada de atender tres tareas en función de la fuente de interrupción, como se describe a continuación. Todas las interrupciones deben atenderse por el flanco decreciente de las respectivas entradas.

- a. **Interrupción PTH0 (INICIO DE ESCRITURA):** En este segmento de código se debe poner el bit RW_RTC = 0 e iniciar la escritura del RTC en el DS1307, enviando un comando de START al IIC y el calling address (WRITE).
- b. **Interrupción PH1:** En esta interrupción se debe deshabilitar la interrupción OC5 para suprimir la alarma del despertador.
- c. **Interrupción PTH3/PH2:** En este modulo se debe incrementar (PH3)/Decrementar (PTH2) un contador de brillo, almacenado en una variable denominada BRILLO cuyo valor estará entre 0 y 100. Por cada activación de uno de estos botones se debe incrementar/decrementar en 5 la variable brillo. A partir de la variable brillo se deberá obtener el valor de K en la subrutina OC4_ISR.



Subrutina IIC_ISR: Esta subrutina llama a dos subrutinas:

- a. Subrutina Write_RTC: Se llamará si el bit RW_RTC=0. Esta subrutina deberá escribir los valores almacenados en la Tabla T_Write_RTC en los correspondientes registros del DS1307.
- b. Subrutina Read_RTC: Se llamará si RW_RTC=1. Esta subrutina leerá los valores del DS1307 y los colocará en el arreglo T_Read_RTC.

La interfaz IIC debe configurarse para una bit rate de su elección y en formato 24 horas. El Indice de Interrupción se va a almacenar en una variable denominada INDEX_RTC.

Subrutina OC5: Esta es la subrutina de atención a interrupciones de OC5 que se habilitará cuando se cumpla la hora de la ALARMA del despertador. El objetivo es que se genere un tono audible en el buzzer de la Dragon 12. Para ello debe elegir una frecuencia que desee usar. Note que deberá elegir un ciclo de trabajo apropiado para que el buzzer se escuche con suficiente potencia.

Entregue un informe con todos los detalles de diseño, incluyendo los cálculos realizados. Incluya los diagramas de flujo de todas las subrutinas. Estructure adecuadamente el programa, el código del programa debe ubicarse a partir de la posición \$2000. En la Tabla #1 se muestran las posiciones para las estructuras de

	<p style="text-align: center;">UNIVERSIDAD DE COSTA RICA ESCUELA DE INGENIERÍA ELÉCTRICA</p>	 <small>ESCUELA DE INGENIERIA ELECTRICA UNIVERSIDAD DE COSTA RICA</small>
<p style="text-align: center;">IE-623 MICROPROCESADORES Prof. Geovanny Delgado</p>		

datos a utilizarse, **únicamente** se deben utilizar estas estructuras de datos, todas las estructuras de datos empiezan en la dirección \$1000 **en el orden que aparecen** en la Tabla #1. El registro BANDERAS es para las variables tipo bit que se requieren. Defina cada una de estas banderas y asígnele un nombre y un bit en este registro.

Nota sobre Prueba: Para efectos de comprobar la operación de su tarea cree una T_Write_RTC con la hora y fecha de la entrega de la tarea y cree la constante ALARMA un minuto posterior a la fecha programada en T_Write_RTC.

Debe enviar el código del programa con el formato SuNombre#7.asm a la dirección del curso a más tardar a las 8:00 a.m. del día en que debe entregar su tarea.



IE-623 MICROPROCESADORES
Prof. Geovanny Delgado

TABLA #1
Estructuras de datos

ESTRUCTURA DE DATOS	Tipo	DIRECCION
CONT_RTI	Variable byte	\$1000
BANDERAS	Variable byte	\$ 1001
BRILLO	Variable byte	\$1002
CONT_DIG	Variable byte	\$1003
CONT_TICKS	Variable byte	\$1004
DT	Variable byte	\$1005
BCD1	Variable byte	\$1006
BCD2	Variable byte	\$1007
DIG1	Variable byte	\$1008
DIG2	Variable byte	\$1009
DIG3	Variable byte	\$100A
DIG4	Variable byte	\$100B
LEDS	Variable byte	\$100C
SEGMENT	Tabla	\$100C - \$1015
CONT_7SEG	Variable word	\$1016 - \$1017
Cont_Delay	Variable byte	\$1018
D2mS	Constante byte	\$1019
D240uS	Constante byte	\$101A
D60uS	Constante byte	\$101b
Clear_LCD	Constante byte	\$101C
ADD_L1	Constante byte	\$101D
ADD_L2	Constante byte	\$101E
IniDisp	Arreglo	\$101F-\$1023
Index_RTC	Variable byte	\$1024
Constante Dir_WR	Constante byte	\$1025
Constante DIR_RD	Constante byte	\$1026
Constante Dir_Seg	Constante byte	\$1027
Constante ALARMA	Constante Word	\$1028-\$1029
T_Write_RTC	Tabla	\$1030 - \$1036
T_Read_RTC	Arreglo	\$1040 - \$1046
Inicio Mensajes	Tablas	\$1050