

Laboratorio de Sistemas Basados en Microprocesadores

Práctica 4: Diseño de programas residentes.

Diseñar un programa residente que se ejecute mediante la INT 60h y que proporcione un primer servicio (con AH=12h) de conversión de un entero representado como cadena de caracteres de dígitos decimales (ej.: "65535") al entero correspondiente representado como cadena de dígitos hexadecimales (ej.: "FFFF"), y un segundo servicio (con AH=13h) que haga la conversión inversa: de hexadecimal a decimal.

Las direcciones de las cadenas se pasarán a la RSI en DS:BX y volverán convertidas en DS:CX. Ambas cadenas terminarán en '\$'. Se supone que los números hexadecimales podrán tener 4 dígitos como máximo (ej.: "FFFF") y que los enteros son **SIN** signo.

```
; =====  
;                               Int 60h - Conversor DEC a HEX  
; Input:  
;     Ah = 12h  
;     DS:BX = Cadena ASCII+$ (Decimal)  
; Output:  
;     DS:CX = Cadena ASCII+$ (hexadecimal)  
; =====  
;                               Int 60h - Conversor HEX a DEC  
; Input:  
;     Ah = 13h  
;     DS:BX = Cadena ASCII+$ (Hexa)  
; Output:  
;     DS:CX = Cadena ASCII+$ (Decimal)
```

Programa 1: p4a.asm (3 puntos)

Escribir en ensamblador un programa (.COM) que:

- Implemente una rutina de servicio a la **interrupción 60h**, que ofrezca los servicios especificados anteriormente
- Cuando se ejecute sin parámetros, muestre **el estado** de instalación del driver, **grupo y nombres de la pareja**, y las instrucciones de uso
- Cuando se ejecute con el parámetro **/I**, instale el driver caso que no lo esté ya
- Cuando se ejecute con el parámetro **/D**, desinstale el driver caso de estar instalado

Programa 2: p4b.asm (4 puntos)

Realizar un programa **que pruebe el funcionamiento** del programa residente anterior. Debe imprimir en pantalla la información a convertir y el texto convertido, **usando los dos servicios** de la INT 60h. La información a convertir puede estar almacenada en memoria o puede solicitarse desde el teclado.

La interrupción periódica (1Ch)

El PC, con cierta electrónica de temporización, realiza una petición de interrupción al microprocesador a un ritmo aproximado **de 18,2 veces** por segundo. Esto permite al programador utilizar dicha interrupción cuando necesite realizar una tarea periódica. Esta interrupción es la interrupción hardware 08 cuya rutina de atención, además de ciertas tareas de control y mantenimiento, realiza una llamada a la interrupción software 1Ch. El único contenido de la rutina de atención a esta interrupción es la instrucción IRET. Reinstalando los vectores de esta interrupción el programador puede realizar tareas periódicas sin más modificaciones.

Programa 3: p4c.asm (3 puntos)

Realizar un programa que, dado un entero **introducido desde el teclado**, escriba en pantalla cada uno de sus dígitos convertidos a decimal o hexadecimal **utilizando la interrupción 60h** definida en el apartado anterior, a un ritmo de **un carácter por segundo** aproximadamente, utilizando la interrupción 1Ch para la temporización.

Se recomienda **actualizar el programa p4a.asm** incluyendo la RSI de la INT 1Ch para ser utilizado por el programa p4c.asm

Adicionalmente, **se solicita habilitar/deshabilitar las interrupciones** para instalar el driver, pero en lugar de usar CLI/STI, aplicar una inhibición/ desinhibición selectiva del **TIMER** modificando el bit 0 del registro de máscara (IMR) del controlador de interrupciones maestro. Para ello se recomienda usar **IN** para leer el registro con la ayuda de una máscara binaria para cambiar el bit de menos peso y un **OUT** para escribir el resultado.

Al inicio o cuando se reciba la instrucción **“hex”**, el programa deberá convertir a hexadecimal las cadenas siguientes, salvo que se reciba dec o quit.

Cuando se reciba la instrucción **“dec”**, el programa deberá convertir a decimal las cadenas siguientes, salvo hex o quit.

Cuando se reciba la instrucción **“quit”** el programa deberá finalizar.

Fichero Makefile recomendado:

all: p4a.com p4b.exe p4c.exe

p4a.com: p4a.obj
tlink /t p4a
p4a.obj: p4a.asm
tasm /zi p4a.asm

p4b.exe: p4b.obj
tlink /v p4b
p4b.obj: p4b.asm
tasm /zi p4b.asm

p4c.exe: p4c.obj
tlink /v p4c
p4c.obj: p4c.asm
tasm /zi p4c.asm

clean:
del *.obj
del *.com
del *.MAP
del *.exe

ENTREGA DE LA PRÁCTICA: Fecha y contenido

La práctica se desarrollará en parejas.

Se deberá subir a Moodle un fichero zip que contenga los ficheros fuentes de los programas y el fichero makefile. Sólo podrá ser subido por uno de los miembros de la pareja.

Los ficheros a entregar deberán contener en la cabecera los nombres de los autores y el identificador de la pareja. Asimismo, el código de los ficheros entregados deberá estar correctamente tabulado y comentado. La falta de comentarios o la baja calidad de éstos, será calificada negativamente.

El límite de fecha de subida de los ficheros, para cada grupo es el siguiente:

Grupos del Miércoles: 5 de Mayo de 2020 a las 23:55h

Grupos del Jueves: 6 de Mayo de 2020 a las 23:55h

Grupos del Viernes: 7 de Mayo de 2020 a las 23:55h