

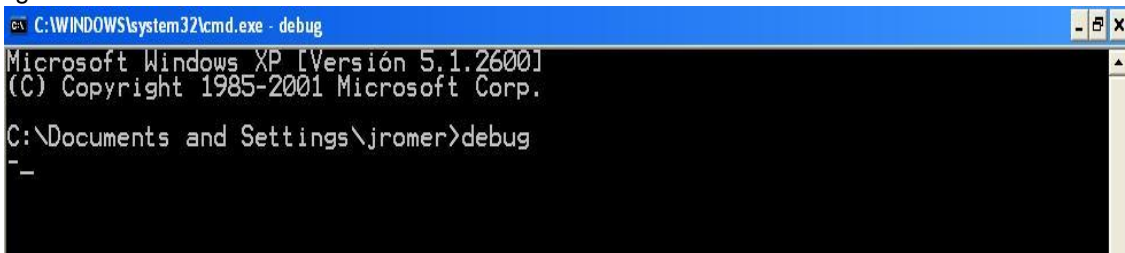
Ejercicio Correspondiente a la segunda clase de Sistemas Operativos del día 17 de Marzo de 2014

Siguiendo con el conocimiento del procesador, sus registros, las interrupciones, la memoria, el lenguaje de máquina y la ejecución de un programa, presentamos un ejercicio básico para comenzar a entender estos temas.

El objetivo del ejercicio es llevar a la memoria del computador una secuencia de bytes que se ingresan por teclado.

Comencemos presentando al Debug, que es la herramienta de software que utilizaremos en este caso para resolver la ejercitación.

Simplemente tipeamos Debug en la línea de comando del DOS, como muestra la siguiente figura.



```
C:\WINDOWS\system32\cmd.exe - debug
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\Documents and Settings\jromer>debug
_
```

Vemos que aparece un guión “_” que es el prompt del debug, donde podremos ingresar los comandos del debug.

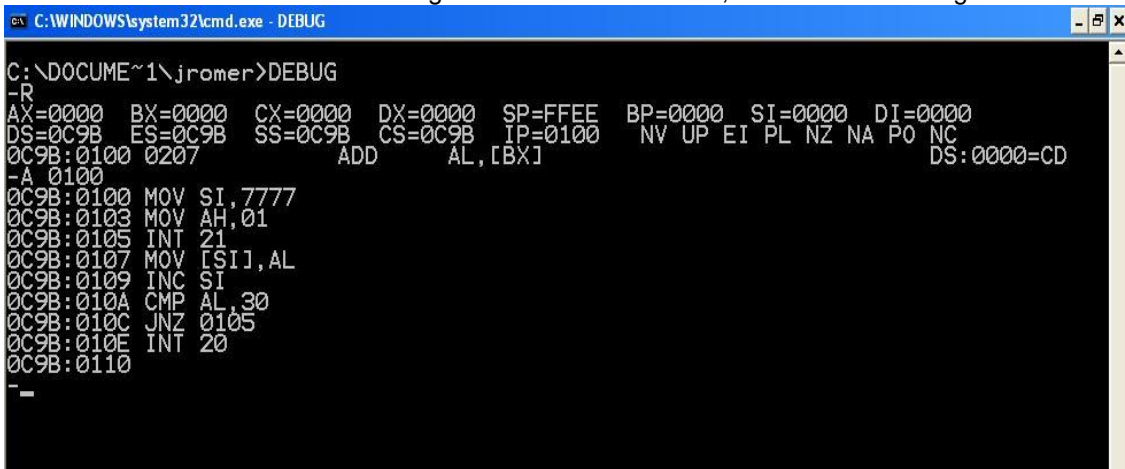
Tipeamos el comando “R” que nos permite visualizar los registros del procesador en su versión 80386, con registros de 16 bits, como muestra la siguiente figura.



```
C:\WINDOWS\system32\cmd.exe - DEBUG
C:\DOCUME~1\jromer>DEBUG
-R
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0C9B ES=0C9B SS=0C9B CS=0C9B IP=0100 NV UP EI PL NZ NA PO NC
0C9B:0100 0207 ADD AL,[BX] DS:0000=CD
_
```

Vamos a ingresar el programa directamente en la memoria, lo tipearemos en assembler, que será más sencillo que tipearlo directamente en lenguaje de máquina.

Utilizaremos el comando “A” de la siguiente manera –A 0100, como muestra la figura.



```
C:\WINDOWS\system32\cmd.exe - DEBUG
C:\DOCUME~1\jromer>DEBUG
-R
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0C9B ES=0C9B SS=0C9B CS=0C9B IP=0100 NV UP EI PL NZ NA PO NC
0C9B:0100 0207 ADD AL,[BX] DS:0000=CD
-A 0100
0C9B:0100 MOV SI,7777
0C9B:0103 MOV AH,01
0C9B:0105 INT 21
0C9B:0107 MOV [SI],AL
0C9B:0109 INC SI
0C9B:010A CMP AL,30
0C9B:010C JNZ 0105
0C9B:010E INT 20
0C9B:0110
_
```

0C9B:0100 es la dirección de memoria en la cual quedo almacenada la instrucción MOV SI,7777 en su lenguaje de máquina.

Tabla resumen del programa

Dirección de memoria de almacenamiento de la instrucción	Instrucción assembler	Descripción de la Instrucción
0C9B:0100	MOV SI,7777	Pone el valor 7777 hexadecimal en el registro SI
0C9B:0103	MOV AH,01	Pone el valor 01 hexadecimal en la parte alta del registro AX, además de indicar la función captura de una tecla presionada por teclado.
0C9B:0105	INT 21	Interrupción que permite la captura de una tecla presionada por teclado
0C9B:0107	MOV [SI],AL	El valor ASCII de la tecla tipeada queda almacenado en la parte baja del registro AX, para no perderla la transferimos a la dirección de memoria 7777 hexadecimal.
0C9B:0109	INC SI	Incrementa SI en 1, ahora SI contiene 7778 hexadecimal
0C9B:010A	CMP AL,30	Compara el contenido de la parte baja del registro AL con el valor 30 hexadecimal
0C9B:010C	JNZ 0105	Cuando el contenido de AL es distinto a 30 hexadecimal, el registro IP toma el valor 0105 y se ejecuta la instrucción INT 21 nuevamente, en caso contrario el registro IP toma el valor 010E y se ejecuta la instrucción INT 20
0C9B:010E	INT 20	Indica finalización de la ejecución del programa

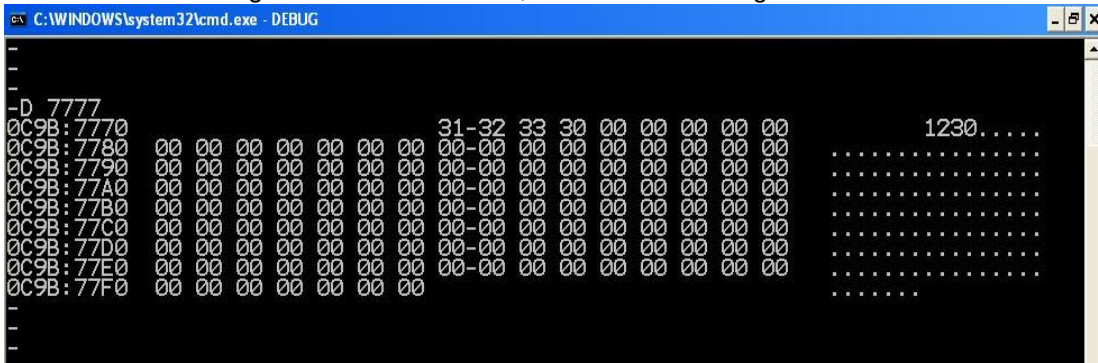
Ahora ejecutamos el programa con el comando "G" del Debug, no aparecerá el prompt del Debug y quedará el cursor esperando que presionemos teclas, en el ejemplo se presionaron las teclas correspondientes a los números 1, 2, 3 y 0 y el programa termina, porque el valor ASCII del 0 es 30 hexadecimal, como muestra la figura.

```

C:\WINDOWS\system32\cmd.exe - DEBUG
C:\>DEBUG
-IP
AX=0000 BX=0000 CX=0000 DX=0000 SP=FFEE BP=0000 SI=0000 DI=0000
DS=0C9B ES=0C9B SS=0C9B CS=0C9B IP=0100 NV UP EI PL NZ NA PO NC
0C9B:0100 0207 ADD AL,[BX] DS:0000=CD
-A 0100
0C9B:0100 MOV SI,7777
0C9B:0103 MOV AH,01
0C9B:0105 INT 21
0C9B:0107 MOV [SI],AL
0C9B:0109 INC SI
0C9B:010A CMP AL,30
0C9B:010C JNZ 0105
0C9B:010E INT 20
0C9B:0110
-g
1230
El programa ha terminado de forma normal
-
-
-

```

El siguiente paso es verificar si en las direcciones de memoria 7777, 7778, 7779 y 777A hexadecimales quedaron los valores 31, 32, 33, 30 hexadecimales, utilizaremos para ello el comando "D" de la siguiente forma -D 7777, como muestra la figura.



Esperamos que el alumno pueda reproducir esta experiencia en su máquina para que se familiarice con los elementos básicos del procesador y se comience a hacer preguntas, desde nuestro punto de vista, la mayoría de las personas tiene una gran avidez por encontrar respuestas muy fácilmente, pero nos olvidamos que si buscamos respuestas es porque hay preguntas, entonces no deberíamos encontrar preguntas para provocar una necesidad de la búsqueda de la respuesta, en este contexto se me ocurre que se le pueden ocurrir algunas preguntas, por ejemplo:

¿Qué significa a nivel de la Unidad de Control que el registro IP toma el valor 0105 y se ejecuta la instrucción INT 21, en caso contrario el registro IP toma el valor 010E y se ejecuta la instrucción INT 20?

¿Pero si se ingreso el número 1 y quedo almacenado el valor 31 y se me ocurre hacer una suma entre el primer número y el segundo, qué debería hacer?

¿Si existe una interrupción para capturar una tecla presionada por teclado, debería haber una interrupción para mostrar datos por pantalla, en el caso que quisiera mostrar la suma de dos números por pantalla o quisiera realizar el eco de lo que ingreso, cómo haría entonces?

¿El assembler no es ejecutable, entonces cuál es el código del programa que realmente se ejecuta?

¿Cuál será la dirección de memoria donde comienza el código de máquina que resuelve la interrupción?

¿Si la primer instrucción MOV SI,7777 esta almacenada en la dirección de memoria 0C9B:0100 y la siguiente en 0C9B:0103, entonces cuál es la longitud bytes de la instrucción?

Terminando, estas son algunas preguntas que se me ocurren que se le ocurrirán a ustedes, espero se le ocurran muchas mas preguntas, por ahora deberán contestar estas preguntas y algo más, deberán programar con la metodología empleada en este pequeño apunte, la suma de dos números ingresados por teclado y mostrar por pantalla el resultado.

Muchísimas Gracias