

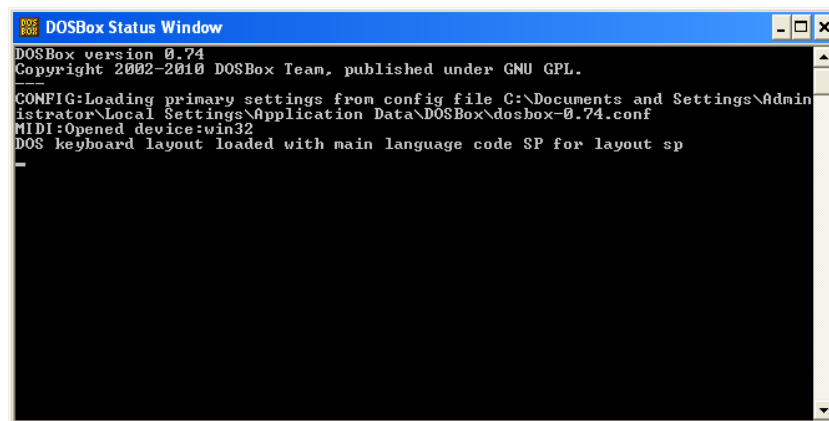
Práctico Nº 10 - MacSim

1) Para poder efectuar la práctica con el simulador de MAC (MacSim), cree una carpeta con el nombre homónimo, por ejemplo, C:\MacSim, baje de la plataforma el simulador_macsim a la carpeta recientemente creada y descomprímalo ahí mismo.

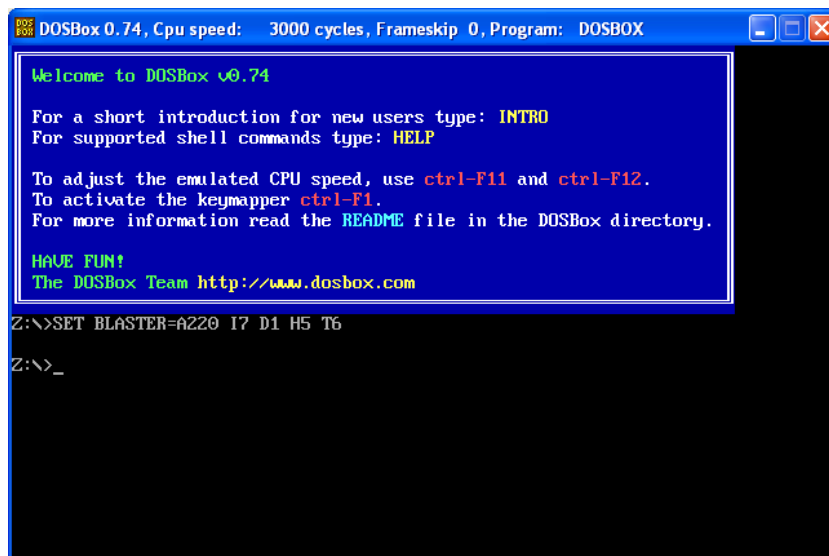
El paso número dos y siguientes, se debe realizar si no funciona el simulador, es decir, bajo entorno WinXP o superior, de 64 bits, no funcionara, ya que el MacSim es una aplicación de 16 bits, es decir, bajo DOS. Para saberlo, ejecútelo desde Window, haga doble click en el archivo macsim, desde el explorador o desde la consola (C:\macsim\macsim). Si funciona tendrá que ver la pantalla de inicio, que se ve más adelante. Siga la práctica a partir de la figura de dicha pantalla.

2) Si el resultado anterior no es satisfactorio, baje de la red un emulador de DOS, como por ejemplo el DosBox (GNU GPL <http://dw3.uptodown.com/ic/dosbox-0-74-en-win-setup.exe> o <https://sourceforge.net/projects/dosbox/files/dosbox/0.74/DOSBox0.74-win32-installer.exe/download>) e instálelo. También puede correr el MacSim desde una máquina virtual, con DOS como sistema operativo o WinXP de 32 bits.

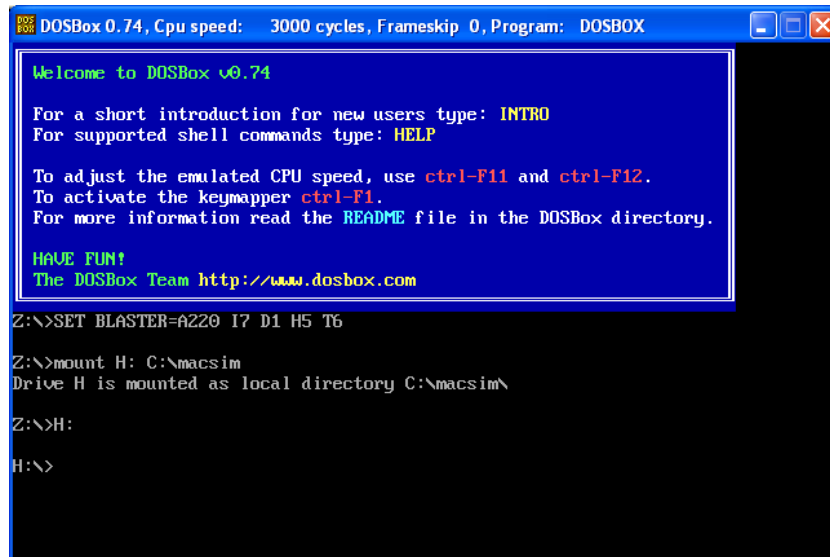
3) Ejecutando el DosBox, le deberá aparecer dos ventana, la primera muestra la configuración por defecto del DosBox:



La siguiente es la de trabajo:



Por último, para poder trabajar con el MacSim, debemos montarlo, con el comando pertinente, es decir Z:\>mount H C:\macsim, como se aprecia en la imagen siguiente:



```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

Welcome to DOSBox v0.74

For a short introduction for new users type: INTRO
For supported shell commands type: HELP

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

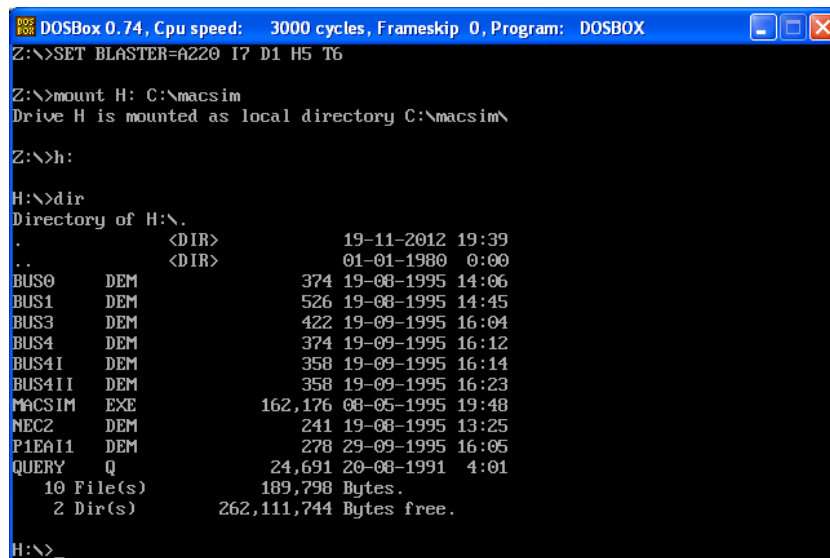
HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>mount H: C:\mcsim
Drive H is mounted as local directory C:\mcsim\

Z:\>H:
H:\>
```

Si aplico el comando dir o DIR, ya que en DOS es indistinto mayúsculas y/o minúsculas, obtendría los archivos descomprimidos del simulador, como muestra la siguiente figura:



```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

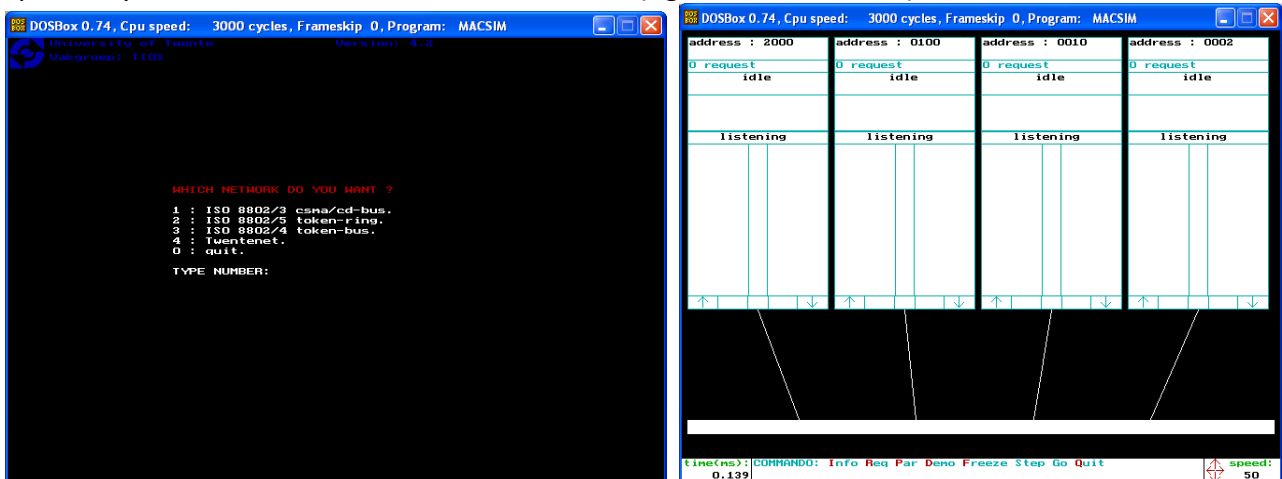
Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>mount H: C:\mcsim
Drive H is mounted as local directory C:\mcsim\

Z:\>h:
H:\>dir
Directory of H:\.
.                <DIR>                19-11-2012 19:39
..               <DIR>                01-01-1980 0:00
BUS0             DEM                  374 19-08-1995 14:06
BUS1             DEM                  526 19-08-1995 14:45
BUS3             DEM                  422 19-09-1995 16:04
BUS4             DEM                  374 19-09-1995 16:12
BUS41            DEM                  358 19-09-1995 16:14
BUS41I           DEM                  358 19-09-1995 16:23
MACSIM           EXE                  162,176 08-05-1995 19:48
NEC2             DEM                  241 19-08-1995 13:25
P1EA11          DEM                  278 29-09-1995 16:05
QUERY           Q                    24,691 20-08-1991 4:01
10 File(s)      189,798 Bytes.
2 Dir(s)        262,111,744 Bytes free.

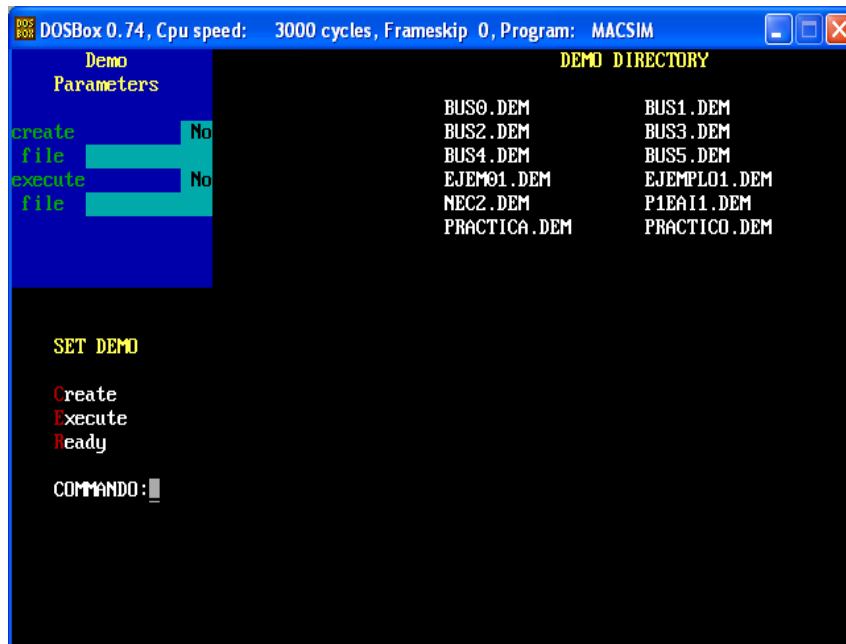
H:\>_
```

Escriba mcsim y tendrá que ver la ventana de la izquierda (pantalla de inicio). Luego, opte por la opción 1 y entrará al simulador de LAN Ethernet (figura de la derecha).

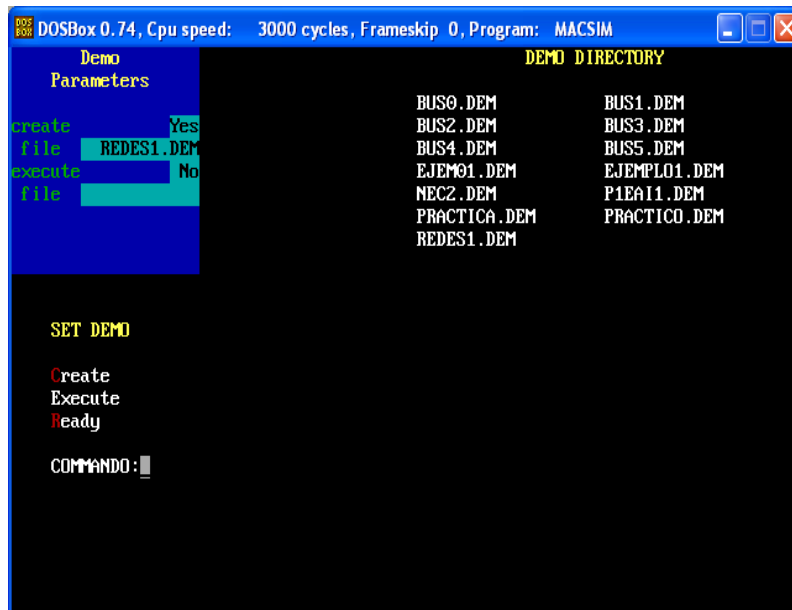


Creación de una simulación:

1) Para ir a los distintos menús se debe presionar la letra en color **ROJO** (mayúscula y/o minúscula es indistinto). Desde la pantalla de la derecha congelamos la futura simulación, presionando la tecla F (**F**reeze) y disminuimos la velocidad de simulación (**s**peed) con las teclas de abajo del movimiento del cursor, por ejemplo a 10.



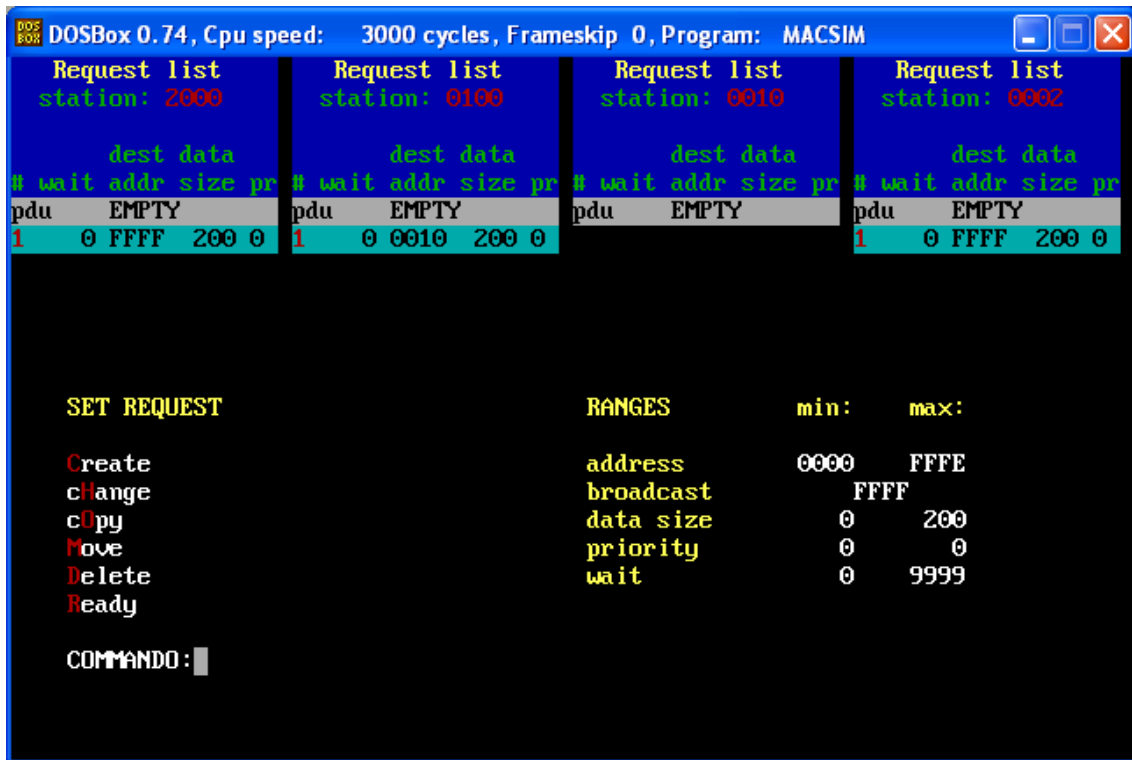
2) Presionamos la letra D (**D**emo), vamos a ver la pantalla de la derecha. Para crearla tecleamos la letra C (**C**reate). Nos pide un nombre de 8 letras y nos propone una extensión por defecto. Escriba redes1, luego acepte y tendrá que ver la ventana de abajo. Luego presionando la tecla, R (**R**eady) o Esc, se vuelve a la pantalla anterior, en este caso la figura donde están las 4 estaciones.



3) Para crear los Requerimientos, por ejemplo a, b y c, para la simulación, presionamos R (**R**eq) y configuramos lo siguiente:

- Desde la estación cuya MAC es 2000, se trasmite 200 B a la MAC de Broadcast (FFFF).
- Desde la estación cuya MAC es 0002, se trasmite 200 B a la MAC de Broadcast (FFFF).
- Desde la estación cuya MAC es 0100, se transmiten 200 B a la MAC 0010.

Cada requerimiento se crea ejecutando el menú **Create**. En cada uno de ellos, las variables no especificadas se consideran las de por defecto. Por último, la ventana tendría que quedar como se muestra a continuación:



Presionando **R** (Ready), regreso a la pantalla de las 4 estaciones de donde puedo comenzar la simulación. Ésta la puedo comenzar presionando **S** (Step), es decir, paso a paso, o con **Go** se ejecuta continuamente. En el transcurso de la simulación, el flujo de bits se identifica con distintos colores cuando es datos (celeste), colisiones (azul) y ruido (fuxia).

Las estaciones pueden tomar los siguientes estados:

- Con respecto del canal, puede adoptar:
 1. Listening: siempre que el canal está desocupado.
 2. Ignoring: la estación ignora el canal por dos motivos:
 - i. La dirección de destino de la trama que le llega, por el canal, no es la suya.
 - ii. Detectada una colisión y una vez enviado los 48 bits (ruido), se mantiene en este estado hasta que se limpia el canal de los restos de tramas de la colisión.
 3. Transmitting: cuando estamos enviando una trama de un requerimiento (pdu) o los 48 bits para avisar a las demás estaciones que ocurrió una colisión.
 4. Receiving: cuando estamos recibiendo una trama con nuestra dirección de destino o de broadcast.
- Respecto del requerimiento la estación puede estar:
 1. Jamming: cuando una estación detecta que está recibiendo más potencia que la que está enviando, sabe que ha ocurrido una colisión, por lo que aborta su transmisión y genera una ráfaga de ruido de 48 bits para avisar a las demás estaciones.
 2. Deferring: cuando tiene un requerimiento bloqueado porque no lo puede transmitir. Puede ser porque el canal esta tomado por otra estación o porque hubo una colisión.

3. Idle: cuando no tiene ningún requerimiento.
4. Transmit data: cuando está transmitiendo el requerimiento (PDU - **Unidad de Dato de Protocolo**).
5. Collision Backoff Timer: después de un número de colisiones, la estación espera un cierto tiempo fijo para poder trasmitir. Este tiempo se calcula, teniendo en cuenta el modelo binario regresivo como (2^i-1) , las ranuras de tiempo fijas de acuerdo a la siguiente tabla:

El slot es dependiente de la velocidad

10 Base slot - 512 bits - 51.2 μ s

100 Base slot - 512 bits - 5.12 μ s

1000 Base slot - 4096 bits - 4.09 μ s

y del número especificado en el Truncate. Recordar que cada BackOffTimer, es con respecto a cada estación.

- 4) Simular los siguientes requerimientos y analizar el comportamiento del canal, las colisiones que se efectúan, en que estación se disparan los BackOffTimer y que ocurre luego.

Request list station: 4000	Request list station: 3000	Request list station: 2000	Request list station: 1000
dest data	dest data	dest data	dest data
# wait addr size pr	# wait addr size pr	# wait addr size pr	# wait addr size pr
pdu EMPTY	pdu EMPTY	pdu EMPTY	pdu EMPTY
1 0 1000 200 0	1 0 1000 200 0	1 0 1000 200 0	1 500 FFFF 200 0
2 1000 FFFF 200 0			

Request list station: 4000	Request list station: 3000	Request list station: 2000	Request list station: 1000
dest data	dest data	dest data	dest data
# wait addr size pr	# wait addr size pr	# wait addr size pr	# wait addr size pr
pdu EMPTY	pdu EMPTY	pdu EMPTY	pdu EMPTY
1 0 1000 200 0	1 0 2000 200 0	1 0 3000 200 0	1 0 4000 200 0

Request list station: 4000	Request list station: 3000	Request list station: 2000	Request list station: 1000
dest data	dest data	dest data	dest data
# wait addr size pr	# wait addr size pr	# wait addr size pr	# wait addr size pr
pdu EMPTY	pdu EMPTY	pdu EMPTY	pdu EMPTY
1 0 FFFF 200 0	1 0 FFFF 200 0	1 0 3000 200 0	1 0 4000 200 0

Request list station: 4000	Request list station: 3000	Request list station: 2000	Request list station: 1000
dest data	dest data	dest data	dest data
# wait addr size pr	# wait addr size pr	# wait addr size pr	# wait addr size pr
pdu EMPTY	pdu EMPTY	pdu EMPTY	pdu EMPTY
1 0 4000 200 0	1 0 3000 200 0	1 0 2000 200 0	1 0 1000 200 0