Ordenes del simulador VHDL VSYSTEM

BD archivo linea

Borra el punto de ruptura (breakpoint) colocado previamente en la *línea* y *archivo* especificados.

Ejemplo:

```
BD  rom.vhd  123
```

Borra el punto de ruptura de la línea *123* en el archivo *rom.vhd*

BP [archivo linea [orden { ; orden}]]

Coloca un punto de ruptura (*breakpoint*) en la *línea* y *archivo* especificados. Si se omiten el archivo y la linea, lista todos los puntos de ruptura actuales. Se puede incluir más de una *orden* separadas por *punto y coma* (;) para que se ejecuten cuando se alcance el punto de ruptura.

Ejemplos:

```
BP
```

Lista todos los puntos de ruptura, sus archivos fuente y las posibles órdenes asociadas.

```
BP  rom.vhd  123
```

Coloca un punto de ruptura en la línea *123* del archivo *rom.vhd*.

```
BP  ram.vhd  43  EX  var1; EX var2
```

Coloca un punto de ruptura en la línea *43* del archivo *ram.vhd*, y examina (EX) los valores de las variables *var1* y *var2*.

CHANGE variable valor

Cambia una *variable* a un nuevo *valor*. El simulador debe encontrarse en un punto de ruptura o parado como consecuencia de la orden STEP. La variable debe ser de tipo enumerado, entero, físico, punto flotante o *array* de caracteres. Los *arrays* en general y los registros (*record*) se pueden cambiar elemento a elemento.

Ejemplo:

```
CHANGE  palabra  16#FFFF
```

Cambia el valor de la variable *palabra* a hexadecimal FFFF

CONT

Continua la ejecución de una simulación después de un punto de ruptura o la orden STEP.

DESCRIBE nombre

Describe en la ventana *Transcript* el tipo de la variable o señal desinada por *nombre* . Para describir una variable el simulador deberá estar parado en un punto de ruptura o como consecuencia de la orden STEP. Una señal se puede describir en cualquier momento

DO archivo { parámetros }

Ejecuta las ordenes del Simulador VHDL contenidas en un *archivo* macro. Opcionalmente se pueden pasar hasta 9 valores al archivo macro utilizando *parámetros* actuales colocados a la derecha del nombre del archivo y separados por blancos. Estos parámetros se corresponderán en orden con los parámetros formales $1 hasta $9 del archivo macro.

Ejemplos:

```
DO prueba rom.vhd 123
```

En este caso se supone que existe un archivo macro de nombre *prueba* que contiene la orden *BP $1 $2*:

prueba

```
.
bp $1 $2
.
```

y el efecto será colocar un punto de ruptura en la línea *123* del archivo fuente VHDL *rom.vhd*.

DRIVER señal

Muestra en la ventana Transcript el *driver* de la *señal*, es decir, el valor actual de la señal y los futuros valores planificados. Si la señal es de tipo *array* o *record*, sólo se muestra el *driver* de la primera componente.

ENV [nuevo-entorno]

Si no se especifica *nuevo_entorno* visualiza el camino del entorno actual de señales, en caso contrario *nuevo_entorno* pasa a ser el entorno actual. Los entornos se corresponden con los niveles estructurales del programa.

EXAMINE nombre

Visualiza sobre la ventana Transcript el valor de la variable o señal *nombre*. Para visualizar el valor de una variable, el simulador debe estar parado en un punto de ruptura o como consecuencia de la ejecución de la orden STEP. Las señales se pueden visualizar en cualquier momento.

FORCE señal valor [tiempo] {, valor tiempo} [-repeat periodo]

Se utiliza para aplicar estímulos interactivamente a una unidad de diseño bajo simulación. Planifica en el *driver* de la *señal* el *valor* especificado para un *tiempo* -opcionalmente especificado- a partir del tiempo actual de simulación. Si no se especifica ningún tiempo se toma por defecto *0*.

Se puede definir con esta orden una forma de onda compleja planificando sobre el *driver* de la señal más de una pareja (*valor tiempo*) separadas por comas (,). Opcionalmente, se puede repetir la orden cada *periodo* de tiempo utilizando el parámetro *-repeat*. El tipo de la señal puede ser enumerado, entero, flotante, o un *array* de caracteres. Los *arrays* en general y los registros (*records*) se pueden forzar elemento a elemento.

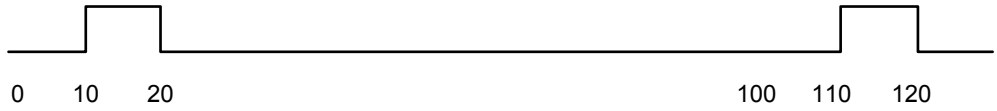
Ejemplo:

```
force input1 0  100
```

Fuerza el valor de la señal *input1* a *0 100* unidades de tiempo después del tiempo actual de simulación.

```
force  input1  1  10,  0  20  -rep  100
```

Fuerza la señal *input1* a *1* cuando transcurran 10 unidades de tiempo, y a *0* cuando transcurran *20* unidades de tiempo. Esto se repite cada *100* unidades de tiempo, por lo que se genera la siguiente forma de onda:



HELP

Visualiza sobre la ventana Transcript la lista de las Ordenes del Simulador VHDL junto a un comentario resumido de su función.

LIST { señal [= nombre] }

Lista sobre la ventana List las *señales* especificadas junto a sus valores. Los tipos de las señales pueden ser enumerado, entero, físico, punto flotante, o un *array* de caracteres. Los *arrays* en general y los registros (*record*) se pueden visualizar componente a componente.

Opcionalmente se puede visualizar una *señal* con otro *nombre*. Esto es útil cuando los nombres originales de las señales son demasiado largos y no caben simultáneamente en la ventana.

Ejemplo:

```
list  a    /nivel/señal = s   b
```

Visualiza sobre la ventana List las columnas de valores de las señales especificadas encabezadas por los nombres a s b

NOFORCE señal

Anula la repetición de una orden FORCE sobre una *señal*.

NOLIST señal

Deja de visualizar en la ventana List la señal especificada.

PROCESS

Visualiza sobre la ventana Transcript el estado de todos los procesos. Los posibles estados son:

- Done

(no espera nada)

- Inactive wait for signal

(espera un evento en una señal)

- Inactive wait for timeout

(espera que transcurra un valor de temporización)

- Ready

(listo para ser ejecutado en el ciclo actual de simulación)

- Atctive Breakpoint

(parado en un punto de ruptura)

- Active Single-step

(parado en ejecución paaso-apaso)

QUIT

Abandona el simulador.

RUN [tiempo] [-step]

Ejecuta la unidad de diseño seleccionada previamente. Si no se especifica ningún tiempo, el simulador avanza un valor por defecto (200 ns).

Si se especifica el parámetro **-step**, la simulación se detiene en la primera línea de ejecución del del código VHDL.

Ejemplo:

```
run 2000
```

El simulador avanza 2000 unidades de tiempo.

SHOW [entorno] [-all]

Visualiza sobre la ventana Transcript todas las señales y regiones accesibles desde un entorno.

Si no se especifica ningún entorno, por defecto se toma el entorno actual.

Si se especifica el parámetro **-all**, se visualizan todas las señales y regiones por debajo del entorno actual.

SOURCE archivo

Visualiza el archivo en la ventana Source. Una vez visualizado el archivo se pueden poner o borrar puntos de ruptura con el ratón, llevando el cursor a la línea correspondiente.

STEP [-over]

Avanza a la siguiente sentencia del programa VHDL. Si se utiliza el parámetro **-over**, se omiten los subprogramas (*procedures* y *functions*)

VIEW [archivo]

Visualiza el archivo ASCII especificado. Para cerrar la ventana se introduce VIEW sin nombre de archivo.

Tabla de Ordenes del Simulador VHDL

BD archivo linea

BP [archivo linea [orden { ; orden}]]

CHANGE(ch) variable valor

CONT(co)

DESCRIBE(de) nombre

DO archivo { parametros }

DRIVER(dr) señal

ENV [nuevo-entorno]

EXAMINE(ex) nombre

FORCE señal valor [tiempo] {, valor tiempo} [**-REPEAT** periodo]

HELP(h)

LIST(l) { señal [= nombre] }

NOFORCE(nof) señal

NOLIST(nol) señal

PROCESS(p)

QUIT(q)

RUNr) [tiempo] [-paso]

SHOW(sho) [entorno] [**-all**]

SOURCE(so) archivo

STEP(st) [**-over**]

VIEW(v) [archivo]