

Manual de ayuda para el diseño, simulación, generación y fabricación del PCB de circuitos con ayuda de OrCAD 9.1 (parte I. Capture)

La finalidad de este documento pretende ser una ayuda para la realización de las prácticas de Tecnología Electrónica y no un manual sobre el manejo del paquete de software OrCAD 9.1.

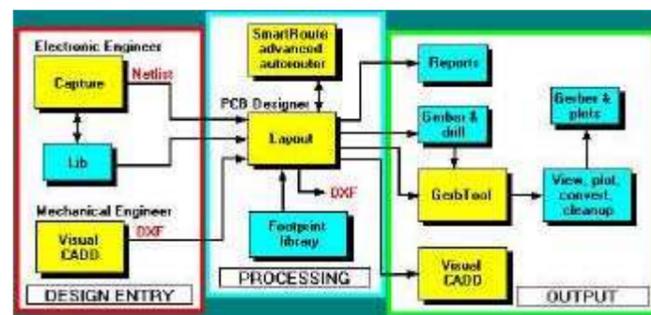
El paquete OrCAD 9.1 incluye, dentro de la carpeta DOCUMENT, un buen número de manuales en formato PDF, con documentación relativa a los distintos programas que contiene. (El fichero rel9pdf. PDF contiene la descripción de cada documento).

Introducción:

El paquete OrCAD 9.1 contiene una serie de programas para el diseño electrónico ayudado por ordenador que incluyen distintas librerías para el diseño del esquema y la simulación del sistema electrónico así como los encapsulados para la realización del PCB.

En este manual utilizaremos los programas CAPTURE para dibujar los esquemas, PSpice para la simulación y LAYOUT para el diseño del PCB:

El proceso de diseño y fabricación asistido por ordenador con ayuda del paquete OrCAD los podemos dividir en las siguientes partes:

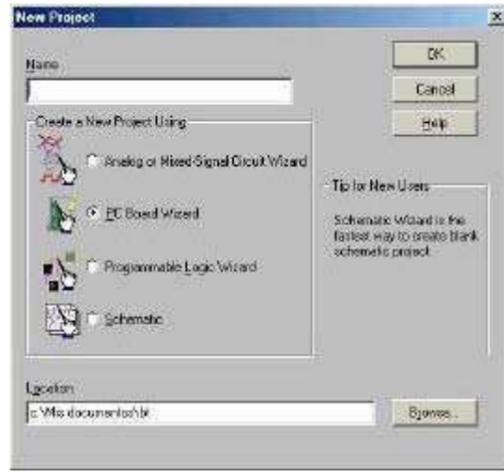


1. Diseño del esquema del circuito con ayuda de CAPTURE.
2. Simulación del funcionamiento del circuito utilizando PSpice.
3. Diseño de la placa del circuito impreso mediante LAYOUT.
4. Realización de la placa.
5. Montaje del circuito y verificación de su funcionamiento.

El programa OrCAD nos permite realizar los tres primeros apartados mediante la realización de un proyecto.

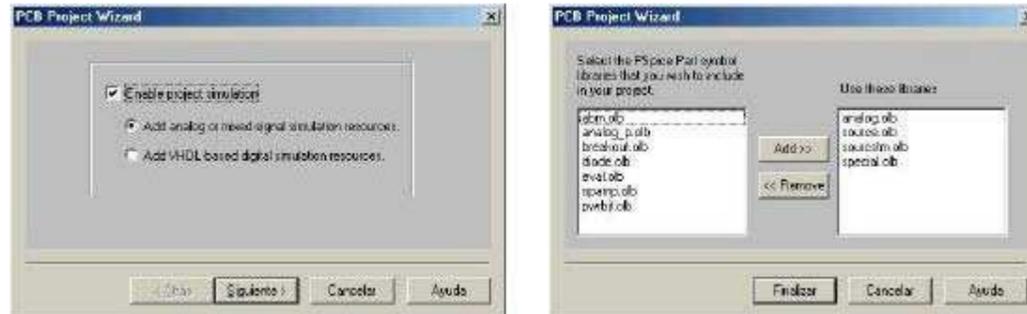
Iniciar el proyecto de diseño de un sistema electrónico.

Arrancar el programa *Capture* y comenzar un proyecto nuevo con la opción **File->New->Project**, aparecerá la siguiente ventana de diálogo para indicar el directorio (Location), el nombre del proyecto (Name) y el tipo de proyecto.



Como se trata de diseñar una placa PCB hemos de señalar la opción **PC Board Wizard**, pues ello simplifica mucho las operaciones posteriores.

Al pulsar el botón OK nos muestra la ventana de diálogo que podemos ver a continuación a la izquierda.



Como queremos simular el circuito con ayuda de PSpice, debemos marcar las casillas que se muestran en la ventana izquierda. En la ventana de la derecha incluiremos las librerías con los modelos de los componentes de nuestro esquema a simular.

1. Diseño del esquema del circuito con Capture.

CAPTURE: El programa Capture es un editor de esquemas que se comunica directamente con otros programas de *OrCAD*. Desde este entorno se puede, dibujar circuitos, simular circuitos con ayuda de PSpice, analizar los resultados de la simulación con Probe, generar los ficheros necesarios para diseñar la placa del circuito impreso con Layout, etc.

Para más información, ver el manual [*OrCAD Capture User's Guide*](#).

El programa dispone de un tutorial (en inglés) que nos permite conocer el manejo básico del programa de diseño de esquemas electrónicos Capture.

Para iniciar el tutorial, una vez arrancado el programa Capture, ejecute el comando Help->Learning Capture y posteriormente el botón Lesson Menu. Aparecerá una ventana con varios botones que representan cada uno de ellos una lección. Cada lección consta de varias páginas que podemos ver paso a paso pulsando el botón Continue, cuando llegue al final encontrará un resumen de lo tratado en la lección. Después, pulse Continue y verá un cuadro de diálogo que le invitará a hacer los ejercicios relacionados con la lección. Al empezar el ejercicio tendrá dos ventanas en la pantalla: la del programa Capture y otra que le va mostrando el texto que le indica lo que tiene que hacer en cada caso.

Las lecciones que contiene el tutorial son las siguientes:

1. About Capture. Introducción a Capture.
2. Designs and schematics. Trata sobre la forma de organizar la información en los proyectos de Capture.
3. Navigating designs. Navegación en los diseños de Capture.
4. Editing a schematic page. Edición de esquemas.
5. Making Connections. Realización y edición de las conexiones.
6. Adding text and graphics. Enseña cómo añadir texto y gráficos en los esquemas..
7. Editing properties. Configuración y edición de los valores de los componentes.
8. Managing parts and libraries. Manejo de componentes y librerías.
9. Making parts. Creación de librerías de componentes.
10. Processing your design. Creación de netlist para simulación, para diseño de la placa de circuito impreso, lista de materiales, etc.
11. Printing and plotting. Procedimiento para obtener una copia impresa del diseño.
12. Compatibility with SDT. El programa Capture puede cargar directamente los diseños realizados con la vieja versión de OrCAD, llamada SDT.
13. CIS (Component Information System). CIS es una base de datos de los componentes utilizados en el diseño que dispone de conexión a Internet.

Elaboración del esquema.

a. Colocar los componente: Con la opción Place->Part nos muestra la ventana para seleccionar el componente, una vez seleccionado, pulsando OK lo deposita en el



Pulsando R podemos rotar el componente. Cuando tengamos varios componente iguales que rotar, es aconsejable rotar uno y luego con la opción copiar y pegar colocar los restantes.

Colocar la masa y alimentación del circuito. Desde Place->Power seleccionar 0/SOURCE para GND y VCC/CAPSYM para Vcc.

Realizar las conexiones. Utilizar la opción Place->Wire.

Actualizar los nombre y los valores de los componentes. Haciendo doble clic sobre la etiqueta correspondiente cambiar los valores

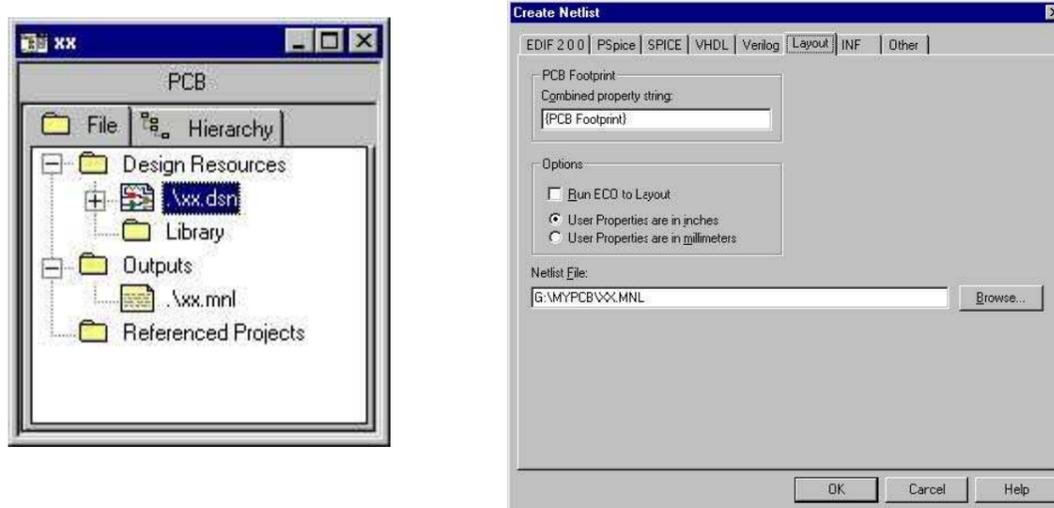
Modificar las fuentes de excitación para el PCB. Los modelos de las fuentes de excitación que hay en la librería Source (VSRC, VDC, etc) identifican el terminal positivo como 0, el programa layout este nombre de terminal no lo reconoce, por lo que vamos a renombrar en las fuentes que utilizamos e el diseño los terminales 0 como terminales 2. Seleccionar la fuente a modificar, pulsar Edit->Part, hacer doble clic sobre el terminal positivo, cambiar en el campo Number el 0 por 2 pulsar OK y File->close, Udate Current para actualizarlo solo en nuestro esquema.

Colocar sondas en los puntos del circuito que queramos visualizar las señales, para que nos las muestre directamente PROBE.

Definir los Footprint para el PCB. Aunque se puede realizar desde la ventana de propiedades, vamos a realizarlo por comodidad desde la ventana del proyecto (en el Terminal Server da error al cerrar la ventana de propiedades). Activar la ventana del proyecto desde Window->nombreproyecto.opj. Seleccionar el icono PAGE1, Edit->Browse->Parts, OK, seleccionar todos los componentes, Edit->Properties, actualizar en la columna Footprints los valores correspondientes a los footprints a utilizar. En la columna PspiceOnly borrar el valor TRUE de las fuentes de excitación para que las incluya en el netlist del PCB.

Elaboración del netlist para PCB Layout.

Desde la ventana del proyecto, seleccionar el icono del proyecto, pulsar Tools



>Create Netlist, activar la pestaña Layout y marcar la opción RUN ECO to Layout y User Properties are in inches para trabajar en pulgadas, indicar el directorio de trabajo donde guardar el netlist (.MNL), OK (Aceptar).

Elaboración del netlist para PSpice.

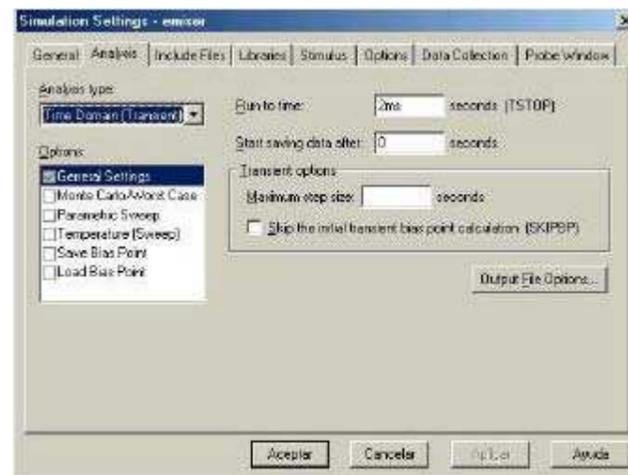
Desde la ventana del proyecto, seleccionar el icono PAGE1, Pspice->Create Netlist. En el caso de que exista algún error en el diseño no mostrara una

ventana (Session.log) con la información de los errores, corregir los posibles errores y volver a generar el netlist.

Definir las opciones de simulación. Pulsar Pspice->New Simulation Profile, nos muestra una ventana de dialogo



definir los análisis a realizar



Arrancar la simulación y visualizar los resultados.

Pulsar Pspice->RUN, si no detecta ningún error (en caso contrario ver el fichero de salida .OUT), nos muestra en una ventana el posprocesador gráfico PROBE donde nos visualiza las señales y donde podemos realizar las mediadas deseadas.

Parte II PSpice

2. Simulación del funcionamiento del circuito utilizando PSpice.

Introducción a PSpice:

SPICE, desarrollado en la universidad de California, en Berkeley, es el standard mundial para la simulación de circuitos analógicos. PSpice, desarrollado por MicroSim Corporation, es una de las derivaciones comerciales de SPICE. Procede de SPICE2, desarrollado hacia la mitad de la década de los 70 a partir del original SPICE. Puesto que se desarrolló con fondos públicos, el código fuente de SPICE2 es de dominio público, lo que ha originado que se realicen numerosas versiones. Incluso algunas compañías tienen una versión propia para adaptar SPICE2 a sus requerimientos. PSpice utiliza los mismos algoritmos que SPICE2, tiene una sintaxis de entrada compatible y comparte su énfasis en la tecnología de circuitos integrados; sin embargo, los conceptos eléctricos son generales y pueden ser útiles para todo tipo de circuitos. P.e., puede emplearse también en la simulación de redes de distribución eléctrica. Para SPICE, no hay diferencia entre microvoltios y megavoltios: ambos son números. PSpice es la primera versión disponible para IBM PC, razón por la cual se ha hecho muy popular. La implementación comercial presenta algunas mejoras respecto del original SPICE2; la más importante es el postprocesador gráfico PROBE que sirve para representar gráficamente los resultados de las simulaciones realizadas por PSpice de forma interactiva y en alta resolución. Otra mejora significativa de PSpice es la posibilidad de simular circuitos digitales y circuitos mixtos; prestación esta que es difícil de encontrar en otros simuladores, ya que la mayoría son exclusivamente analógicos o exclusivamente digitales. PSpice se utiliza normalmente integrado en un sistema de CAE (Computer Aided Engineering) y sus principales funciones son:

- Probar un circuito antes de construirlo.
- Inspeccionar el funcionamiento de un circuito con componentes ideales, sin las limitaciones de los componentes reales.
- Hacer medidas difíciles (debido al ruido eléctrico, la carga de los circuitos, instrumentos de medida no disponibles o destrucción del equipo).
- Simular un circuito muchas veces para comprobar qué porcentaje pasaría un determinado test. El "lenguaje" de PSpice es sencillo de aprender. Las sentencias de control se agrupan en un fichero que es leído por el simulador. Normalmente, las sentencias apenas tienen interacción entre sí y conservan su significado independientemente del contexto. Esta particularidad hace que el lenguaje sea fácil de aprender ya que se puede centrar el estudio en una sentencia, aprenderla y después cambiar a otra. Además, no se necesita conocer muchas sentencias para empezar a trabajar con PSpice. La versión actual de PSpice ha sido desarrollada por OrCAD sobre la última versión de MicroSim.

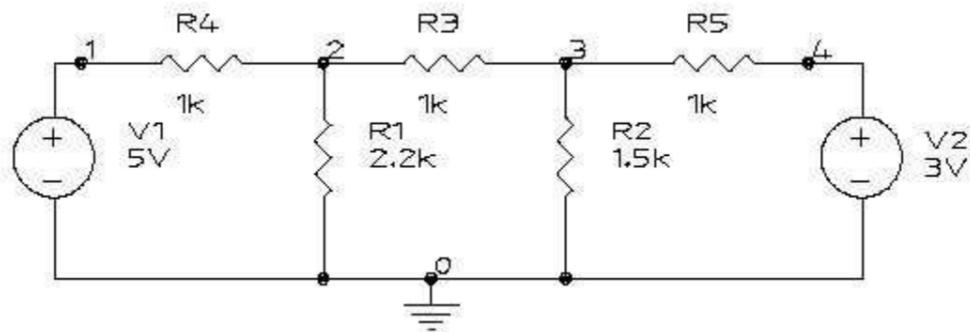
Concepto de Netlist:

El netlist del circuito para la simulación en SPICE es un fichero en texto plano en el que se describe el circuito desde un punto de vista estrictamente eléctrico, excluyendo todo lo referente al encapsulado, circuito impreso y demás detalles físicos.

Reglas sintácticas del netlist.

Las reglas sintácticas para elaborar un netlist son muy sencillas. La primera línea es siempre ignorada, por lo que puede ser utilizada para escribir el título o cualquier otra información de interés. Si cualquier otra línea comienza por el carácter '*' o ';' será interpretada como un comentario. Las demás líneas contendrán información acerca de los componentes del circuito y sus conexiones o bien serán directivas para el simulador que indicarán los tipos de análisis a realizar o le suministrarán información relativa a dichos análisis. En cada línea sólo puede haber un componente o una directiva de análisis.

1º Ejemplo, se analizará el siguiente circuito de continua:



El fichero de entrada es:

```
Ejemplo 2.3
R2 3 0 1.5K
R3 2 3 1K
R4 1 2 1K
R5 3 4 1K
R1 2 0 2.2K
V2 4 0 3V
V1 1 0 5V
.OP .END
```

Comentarios al netlist. La primera línea es siempre ignorada por el simulador y se utiliza para dar alguna información relativa al circuito. La penúltima línea es la directiva .OP que indica al simulador que realice un cálculo del punto de trabajo (Operational Point). Esta directiva puede ser colocada en cualquier línea del fichero de entrada, no necesariamente en la penúltima. Las demás líneas constituyen una descripción del circuito en forma de listas de

componentes en las que se especifica su valor y los nudos a los que están conectados. Este circuito tiene solamente dos tipos de dispositivos: resistencias y fuentes independientes de tensión. La sintaxis para las resistencias es: R<ref> <nudo 1> <nudo 2> <valor> El primer carácter es 'R' que indica el tipo de dispositivo (resistencia); la referencia <ref> puede ser cualquier cadena alfanumérica, que se escribe después del carácter 'R', sin dejar espacios. Los tres argumentos siguientes indican los nudos de conexión y el valor en ohmios. Los separadores para los argumentos pueden ser espacios, tabuladores o comas; pero siempre deben estar en la misma línea. Para las fuentes de tensión independientes, la sintaxis es: V<ref> <nudo positivo> <nudo negativo> <valor> El orden en que se especifiquen los componentes es indiferente. La última línea es la directiva .END que indica el final del análisis. Esta directiva tiene que escribirse en la última línea. Una vez que se haya terminado de escribir este fichero mediante un editor de texto plano, se grabará en el disco dándole un nombre cualquiera con la extensión .CIR, p.e.: EJ_3_1.CIR

Una vez analizado el circuito se obtiene como resultado el fichero EJ_3_1.OUT que contiene los resultados del análisis en formato de texto.

```

**** 09/18/97 19:04:27 ***** Evaluation PSpice (July 1993) *****
Ejemplo 2.3 **** CIRCUIT DESCRIPTION
***** R2 3 0
1.5K R3 2 3 1K R5 3 4 1K R1 2 0 2.2K V2 4 0 3V V1 1 0 5V .OP .END ****
09/18/97 19:04:27 ***** Evaluation PSpice (July 1993) *****
Ejemplo 2.3 **** SMALL SIGNAL BIAS SOLUTION TEMPERATURE =
27.000 DEG C
***** NODE
VOLTAGE NODE VOLTAGE NODE VOLTAGE NODE VOLTAGE

```

```

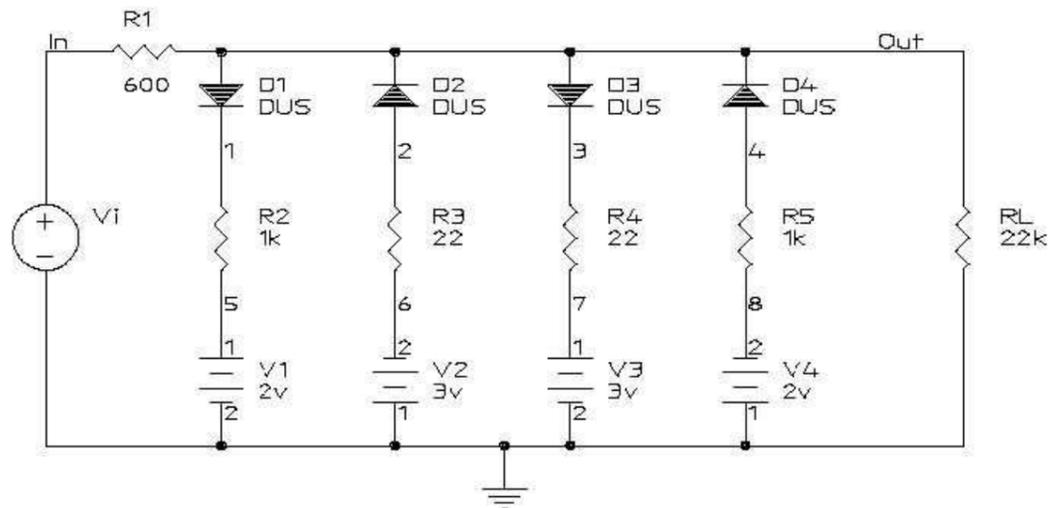
( 1) 5.0000 ( 2) 2.9454 ( 3) 2.2295 ( 4) 3.0000 VOLTAGE SOURCE
CURRENTS NAME CURRENT V2 -7.705E-04 V1 -2.055E-03 TOTAL
POWER DISSIPATION 1.26E-02 WATTS **** 09/18/97 19:04:27 *****
Evaluation PSpice (July 1993) ***** Ejemplo 2.3 **** OPERATING
POINT INFORMATION TEMPERATURE = 27.000 DEG C
*****

```

JOB CONCLUDED
TOTAL JOB TIME .06

Como se puede ver, el fichero de salida presenta al principio una copia del fichero de entrada; después el análisis del punto de trabajo (Small Signal Bias Solution), en el que se especifica la tensión en cada nudo y la corriente en cada una de las fuentes independientes (el signo indica el sentido de la corriente a través de la fuente: positivo significa que la corriente entra por el terminal positivo). También se indica la potencia disipada en el circuito en el punto de trabajo. Al final del fichero, se especifica el tiempo total que se ha utilizado la CPU para el análisis.

2º Ejemplo. Circuito limitador con diodos.



LIMITADOR CON DIODOS EJ_3_2.CIR Línea de título

* NetList del circuito

* Fuente de tensión de entrada, definida por una sucesión de puntos

* unidos por segmentos rectilíneos Vi In 0

PWL (0 0 20m 5 40m 0 60m -5 80m 0)

* Resistencias

R1 In Out 600

R2 1 5 1k R3 2

6 22 R4 3 7 22

R5 4 8 1k RL

Out 0 22k

* Fuentes de tensión continua

V1 5 0 2v V2 0 6 3v V3 7 0 3v

V4 0 8 2v *

* Diodos D1 Out 1 DUS ; DUS es el modelo de diodo

D2 2 Out DUS
D3 Out 3 DUS
D4 4 Out DUS

*

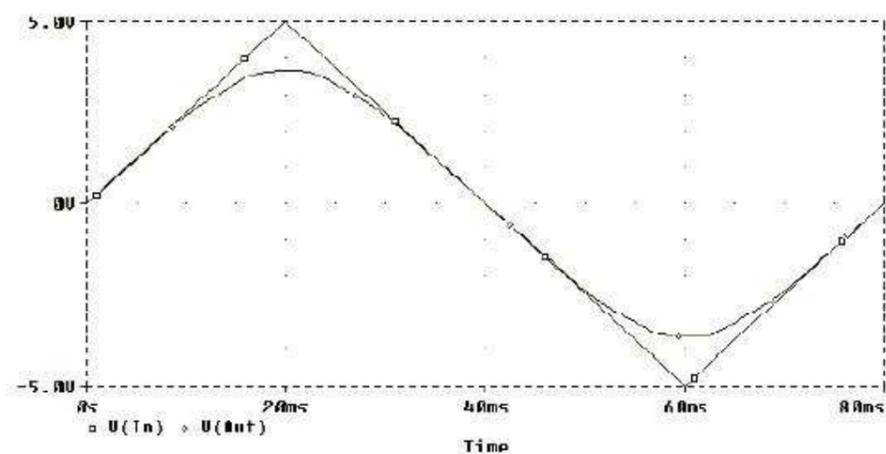
* Modelo de diodo DUS .MODEL DUS D ;
parámetros por defecto *

* Análisis .TRAN 1ms 80ms ; Análisis temporal en pasos de 1ms,
desde 0 hasta 80ms *

* Generar el fichero de datos para Probe
.PROBE *

* Final
.END

Al realizar la simulación igual que en el ejemplo anterior, se generan dos ficheros uno de los cuales tiene la extensión .OUT y contiene los resultados de la simulación que se pueden representar en formato de texto; el otro fichero tiene la extensión .DAT y contiene los resultados gráficos de la simulación en un formato binario que puede ser interpretado por el programa de representación gráfica *Probe*. Este programa arranca automáticamente al terminar la simulación, carga los datos del fichero .DAT y presenta una pantalla gráfica vacía y un menú con varias opciones. La opción *Add_Trace* sirve para trazar cualquier magnitud del circuito que se haya guardado en el fichero .DAT. La traza *V(in)* mostrará la señal de entrada y la traza *V(out)* mostrará la señal de salida.



El fichero Netlist se puede generar:

De forma automática, dibujando el esquema con Capture.
De forma manual mediante un editor de textos.

a. Simulación de circuitos generando el netlist con Capture.

Este es el procedimiento que utilizaremos en las prácticas.

- 1 Para iniciar un nuevo diseño en Capture utilice el comando File->New->Project y, en el cuadro de diálogo que aparece, elija Enable project simulation y Analog or Mixed-Signal Circuit Wizard. Después aparecerá otra caja de diálogo donde podemos seleccionar las librerías que se han de utilizar en el diseño.
- 2 Dibujar el esquema del circuito a simular con Capture.
- 3 Desde Capture generar el netlist para Pspice. Desde la ventana del proyecto, seleccionar el icono del proyecto y la opción Tools->Create Netlist.
- 4 Podrá comprobar que el menú PSpice->Run está deshabilitado. No podrá realizar la simulación hasta que haya definido los tipos de análisis y sus correspondientes parámetros. Esto se hace mediante el menú PSpice->New Simulation Profile.
- 5 Active dicho menú y le aparecerá un cuadro de diálogo donde tendrá que cumplimentar el nombre y la ubicación del fichero de análisis. (Puede darle el mismo nombre que al diseño).
- 6 Después de cerrar este cuadro aparecerá otro donde se indicará las directivas de simulación. Marque las opciones correspondientes a los análisis a realizar. Menú Analysis->Analysis type. Cierre el cuadro.
- 7 Realice el análisis mediante el comando PSpice->Run. Al activar este

comando, se inicia el siguiente proceso:

- Creación del netlist del circuito.
- Arranque del simulador.
- El simulador carga el netlist y realiza la simulación.
- Se genera el fichero de salida .OUT
- Se genera el fichero de datos .DAT si hubiera algún análisis cuya salida se representase en forma gráfica.
- Se arranca el programa *Probe* (Sólo si se ha generado el fichero .DAT)
- Probe carga el fichero .DAT (Si hubiera arrancado).
- Si se produjera algún error, proceda a su corrección.

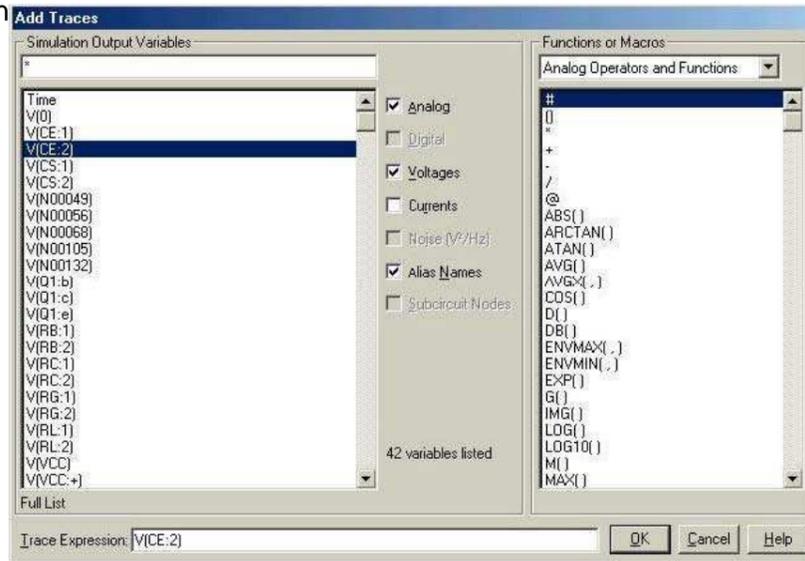
b. Simulación de circuitos generando el netlist con el editor de texto.

1. Arranque el programa de simulación Pspice. Abra un fichero de texto nuevo File->New->Text File. Se abrirá una ventana donde podrá escribir el netlist del circuito y las directivas de análisis (ver tablas de resumen en parte II anexo de la ayuda).
3. Cuando termine, grábelo en su directorio de trabajo. Para simular el circuito tendrá que volver a abrir el fichero como fichero de simulación, no como fichero de texto. Para ello utilice el comando File->Open Simulation.
5. Una vez abierto el fichero de simulación podrá realizarla mediante el comando Simulation->Run (o bien pulsando el botón correspondiente de la barra de herramientas).

6. Para ver el fichero de salida de la simulación utilice el comando View > Output File. Si el simulador avisase de algún error o los resultados no coincidieran con los esperados, realice las oportunas correcciones en el netlist.
7. Analice detenidamente el fichero de salida, comprobando qué clase de información suministra. Si no entiende alguna sección de los resultados, consulte el manual de referencia de PSpice.
8. Para los resultados de forma gráfica, la pantalla del intérprete gráfico Probe se arranca de forma automática. Para visualizar las trazas, utilice el comando Trace->Add Trace y elija la que le interese en el cuadro de diálogo.
9. Si lo desea, puede realizar medidas numéricas mediante los cursores Trace->Cursor-> Display.

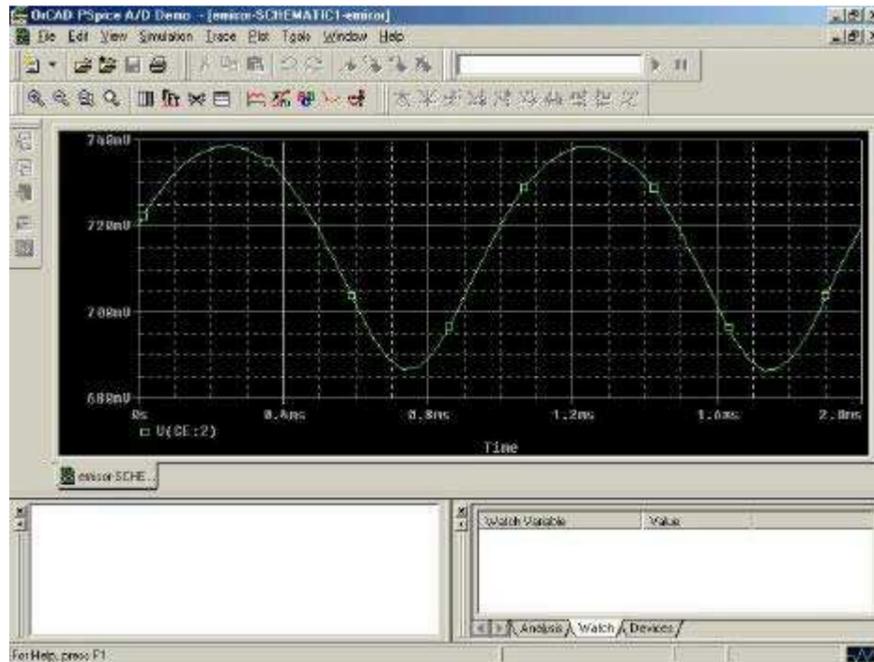
Nota: en el anexo podemos ver la sintaxis a utilizar para los componentes, las fuentes de excitación y los tipos de análisis.

PROBE: Es un posprocesador gráfico que nos muestra los resultados de la simulación



Para representar las distintas gráficas de las señales se debe elegir la opción Trace->Add Trace,

Nos aparecerá una ventana con la lista de todos los puntos del circuito que podremos visualizar, tanto de tensión como de corriente. Además se puede



representar gráficamente cualquier operación matemática como la diferencia, la suma, el producto, etc... de dos o más variables, como por ejemplo el producto

de una corriente de rama por la caída de tensión en una resistencia para calcular la potencia disipada.

Para insertar la representación gráfica de una traza formada por más de una variables, en primer lugar se selecciona una variable, seguidamente la función que se desea y por último la segunda variable. Para simplificar la operación de selección de señales se desactivarán las opciones que no interesen en la ventana Add Traces

Una vez que tengamos representadas las gráficas se puede realizar sobre ellas diferentes operaciones, como pueden ser: poner etiquetas, marcas, conocer las coordenadas de cada punto de la gráfica, etc... A continuación se pasa a describir alguna de estas operaciones.

Opción Plot->Label.

1. (*Plot/Labe/Text*): Escribir texto o poner etiquetas en cualquier parte del gráfico para identificar puntos importantes del gráfico de forma rápida.
2. (*Plot/Labe/Line* y *Poly-line*): Dibujar líneas: líneas simples para completar trazados.
3. (*Plot/Labe/Arrow*): Señalizar mediante una flecha para indicar el punto exacto que se intenta resaltar.
4. (*Plot/Labe/Box*): Dibujar un cuadrilátero .
5. (*Plot/Labe/Circle*): Dibujar un círculo.
6. (*Plot/Labe/Ellipse*): Dibujar una elipse
7. (*Plot/Labe/Mark*). Marcar un punto determinado mediante el cursor. Como resultado se obtiene una etiqueta con las coordenadas de la posición del cursor cuando se pulsó esta opción.

Opción Trace-> Cursor.

Se activa la opción *Display*, para que aparezcan los cursores, Una vez activados, se pueden mantener en la pantalla sin estar utilizándolos activando la opción *Freeze*.

Los cursores se desplazan de una gráfica a otra con la simple acción de seleccionar la gráfica con la que vamos a trabajar. Esto se lleva a cabo pinchando con los dos botones del ratón (uno para cada cursor; botón izquierdo, cursor B1 y botón derecho, cursor B2) sobre el icono que aparece al lado del nombre de cada señal en la parte inferior de la pantalla.

Las operaciones más importantes que se pueden realizar mediante el uso de los cursores, y siempre para obtener información de interés sobre determinados puntos de las trazas, son:

1. Mover el cursor a un máximo local o pico de la gráfica (*Cursor Peak*).
2. Mover el cursor a un mínimo local de la gráfica (*Cursor Trough*).
3. Mover el cursor al punto de máxima pendiente (*Cursor Slope*)
4. Mover el cursor al máximo global de la gráfica (*Cursor Max*).
5. Mover el cursor al mínimo global de la gráfica (*Cursor Min*).

6. Mover el cursor punto a punto por la gráfica. Se debe haber pulsado el icono (*Cursar Point*).

Otras posibilidades interesantes de esta aplicación son las siguientes:

1. Trace->Fourier: Transformada de Fourier de la señal visualizada.
2. Trace->Eval Goal Functions: Permite calcular algunos parámetros de las señales presentes en la simulación, como pueden ser ancho de banda, periodo, etc..
3. Plot->Add Plot to Windows: Permite visualizar dos graficas al mismo tiempo. Por ejemplo Ganancia vs Fase en diagramas de Bode.
4. Plot->Add Y Axis. Permite visualizar varias trazas con diferentes ejes Y.
5. Plot->Axis Settings: Permite modificar la configuración de los ejes. Límites de representación, lineales o logarítmicos, etc..
6. Windows->Copy to ClipBoard: Permite capturar los gráficos de simulación para ser importados a otras aplicaciones Windows a través del portapapeles.

Muchas de estas opciones se pueden acceder directamente, haciendo clic sobre el icono correspondiente.



Parte III Layout

3. Diseño de la placa del circuito impreso mediante LAYOUT.

Introducción a Layout:

La finalidad del programa *Layout* es el diseño de las placas de circuito impreso. Está integrado con el resto de los programas de diseño y análisis electrónico de OrCAD (*Capture* y *PSpice*) y también puede leer y generar ficheros en diversos formatos standard, lo que le permite la comunicación con herramientas de diseño de otros fabricantes como Autocad, Gerber o HP.

El proceso que realiza es el siguiente:

- Lectura de netlist con la información de las conexiones de los componentes.
- Especificación de los parámetros tecnológicos de la placa: distancias mínimas entre componentes, pistas, grosor de las pistas, número de capas, etc.
- Emplazamiento y comprobación de los componentes sobre la placa.
- Trazado automático de las pistas.
- Generación de ficheros de salida para impresora o plotter.

El programa dispone de un tutorial (en inglés) que nos enseña el proceso de realización de una placa ([Help->Learning Layout->Lesson Menu](#)).

Ficheros que intervienen en la creación de una placa. **.MNL** (Netlist). Describe las conexiones de un esquema realizado en Capture, utilizando los nombres de los nudos (nets), componentes y pines. El netlist contiene los footprints, nombre de los componentes, nombres de los nudos, asignaciones de los pines a los nudos, información adicional de los nudos, pines y componentes. **.TCH** (Technology template). Especifica las características de la placa. Incluye la estructura de capas, ajuste de rejilla, normas de espaciado y otros criterios afines. **.TPL** (Board template). Combina los límites geométricos de la placa, con los agujeros de montaje, conectores y otros objetos. **.MAX** (Board file). Contiene toda la información geométrica y eléctrica de la placa.

Información previa.

Antes de comenzar es necesario conocer los términos más importantes utilizados en el diseño de placas de circuito impreso.

Autoplacement. Emplazamiento automático de los componentes. El trazado de las conexiones entre componentes puede simplificarse o complicarse en función de emplazamiento que tengan sobre la placa. Los programas para

diseño de **PCB** incluyen generalmente algoritmos para emplazamiento automático.

Autorouting. Trazado automático de las conexiones.

Drill. Taladro. Los componentes convencionales (no **SMD**) se sueldan a la placa insertando sus terminales en los taladros realizados sobre los pads. **Fan-out.** Los terminales de los componentes **SMD** no penetran en todas las capas sino que se sueldan a la **top-layer** solamente. Para conectar estos terminales a las capas internas es necesario indicarlo mediante el fan-out.

Footprint. Literalmente, "huella del pie". Cada componente situado sobre la placa ocupa un espacio que no puede ser invadido por otro componente. Sus patillas son soldadas a la placa sobre unos **pads**. **Footprint** es el conjunto formado por los pads y la línea que delimita físicamente el componente.

Layer. La placa está formada por varias capas de cobre separadas por material dieléctrico. Las capas externas son la **top layer**, donde están situados los componentes, la **bottom layer**, que es la capa opuesta. Las capas internas son las **inner layers**.

Mounting hole. Agujero para los tornillos de sujeción de la placa.

Net. Nudo. Colección de pistas (**tracks**) eléctricamente unidas.

Pad. Isleta de cobre donde se sueldan los terminales de los componentes.

Pin. Terminal de conexión de un componente.

Plane. Como la mayoría de los circuitos integrados de una placa han de ser alimentados, se reservan dos capas internas para distribuir la tensión de alimentación y la masa a toda la placa. Se denominan **Power plane** y **Ground plane**.

Routing. Trazado de las conexiones. Puede ser manual o automático.

SMD (Surface Mounting Device). Componentes cuyos terminales se sueldan a la superficie de la placa (**top layer**), sin taladro. Si se quiere conectar algún terminal a las capas internas, es preciso recurrir al **fan-out**.

Procedimiento

Iniciar el diseño de una placa

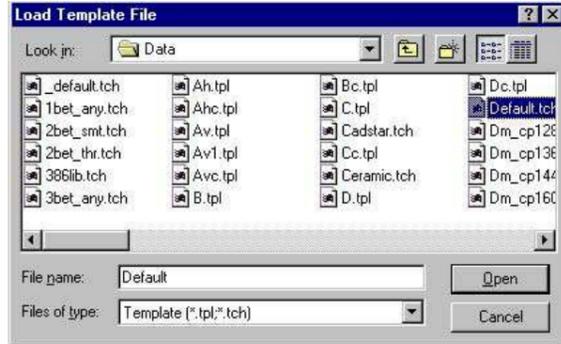
El diseño de la placa se puede iniciar directamente desde el programa de diseño de PCB (*Layout*).

En este caso, los footprints de los componentes han de ser importados desde las librerías donde se encuentren y las conexiones han de ser especificadas una a una.

Lo más frecuente es iniciar el diseño en *Capture*, donde se dibuja el esquema del circuito. Este es el procedimiento que seguiremos, una vez diseñado el esquema, simulado el funcionamiento del circuito con Pspice y generado el netlist para PCB Layout, ejecutamos PCB Layout.

Arrancar PCB Layout

1. Arrancar el programa Layout y pulsar New para iniciar un nuevo proyecto.
2. Una caja de diálogo nos pide el fichero donde se especifican los parámetros tecnológicos.

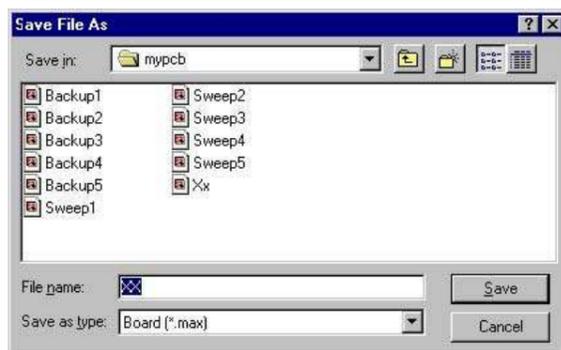


Los parámetros tecnológicos son todos aquellos datos de la placa que no están incluidos en el netlist.

Elija el fichero Default.tch



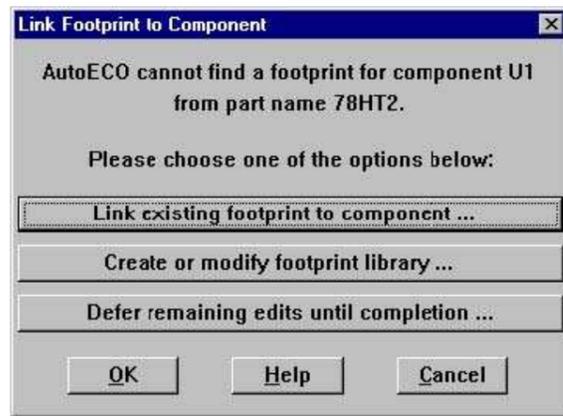
Una nueva caja de diálogo nos pide el nombre y el path del netlist (.MNL).



Por último, el programa solicita el nombre para guardar el proyecto de la placa (.MAX)

3. El proceso AutoECO comienza:

Al cargar el netlist, el programa trata de localizar los footprints de los dispositivos en las librerías; pero cuando alguno de ellos no está asignado o no lo encuentra, nos muestra la siguiente caja de diálogo:



La primera opción nos invita a explorar las librerías y asignar manualmente un footprint al dispositivo en cuestión. Esta opción es adecuada si sabemos de antemano cuál es el footprint que debemos asignar.

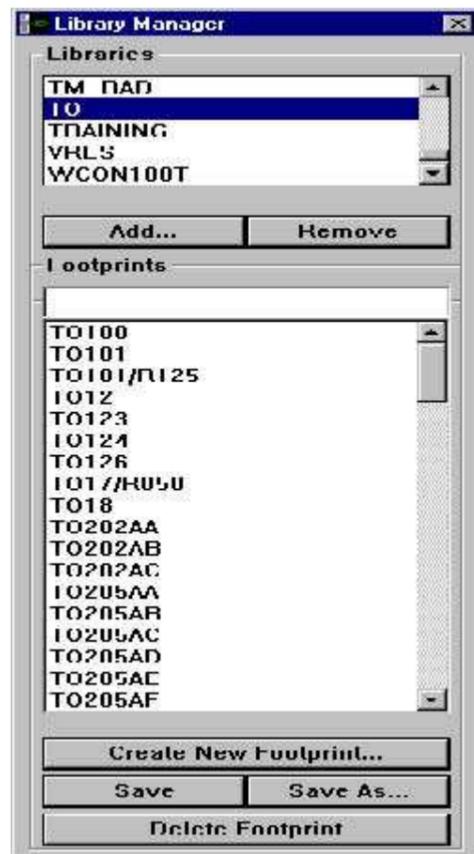
La segunda opción nos invita a explorar la librería completa y elegir el que más se

parezca para modificarlo.

La tercera opción permite postergar esta decisión.

El editor de footprints

Si elegimos la segunda opción se arrancará el programa editor de footprints.



Los footprints están organizados en librerías. En el ejemplo hemos elegido la librería **TO** y podemos ver todos los footprints asociados a esa librería.

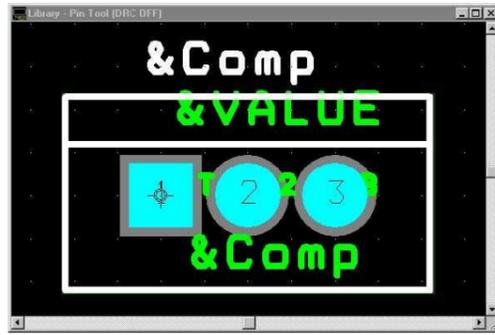
Se pueden crear nuevos footprint, modificar los existentes o crear nuevas librerías.

Al crear un nuevo footprint, utilizar el menú **Options->System Setting**, para fijar la rejilla de trabajo en Mils (m)= 0.01 " y viable 50m (0.05"), de esta forma cada dos puntos visibles corresponden a 0.1" (distancia entre los pines más próximos de un ci).

Para modificar el footprint de un componente de nuestra placa, seleccionar el componente en la placa y con el botón derecho del ratón, seleccionar **Propiedades->footprint->nuevo footprint**.

(Ver lección Working with footprint del tutorial)

Si seleccionamos el footprint TO220AB obtendremos la siguiente figura:



Este es el momento de comprobar si el footprint mostrado corresponde a la cápsula del dispositivo en cuestión. Si no fuera así, habría que buscar otro.

Las dimensiones geométricas de los dispositivos se incluyen habitualmente en los manuales de los fabricantes. Por otra parte, tienen unas dimensiones y nomenclatura normalizadas, de modo que será poco frecuente que tengamos

que diseñar cápsulas nuevas.

En el caso de que no tengamos que diseñar un footprint nuevo, tomaremos nota de este footprint para el componente indicado. Volvamos al cuadro de diálogo y elijamos la primera opción para hacer la asignación

Los principales problemas que detecta el AutoECO referente a la asignación de los footprint son:

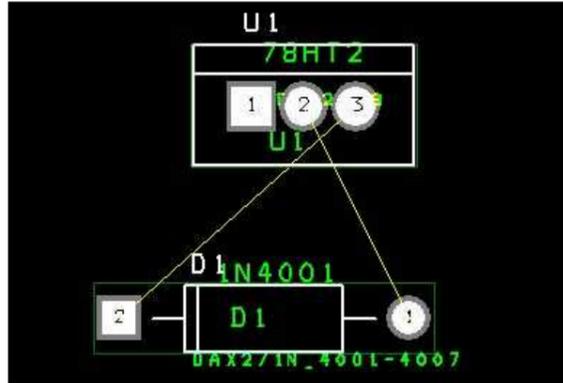
- Que los footprints que asigna el programa Capture por defecto no se corresponden con los que vienen en la librerías de Layout.
- Que los nombre de los footprints de las librerías de Layout no los admite el programa Capture por contener caracteres especiales (/ - por ejemplo para un diodo el DAX2/1N_4001-4007 de la librería TM_DIODE).
- Que no se correspondan las etiquetas de los terminales en el esquema de Capture con los del footprint. Por ejemplo si los terminales del diodo en Capture estuvieran marcados con "A" y "C" en lugar de "1" y "2" como ocurre con otros footprints de esta misma librería, se tendría un error cuando el programa intentase establecer las conexiones.

Es importante observar que a parte coincidir los nombres, se correspondan con los mismos terminales, por ejemplo en el caso del diodo si la etiqueta "2" corresponde al cátodo en Capture, en el footprint el terminal marcado como el cátodo debe ser el 2.

Realizaremos el mismo proceso para todos los componentes de nuestra placa que nos den problemas con los footprints.

Después de estas operaciones el proceso AutoEco termina y aparecerán sobre el espacio de trabajo de Layout los footprints seleccionados y las conexiones en forma de trazos amarillos rectilíneos. Esta representación de la placa se denomina *ratsnest*.

4. Colocar los Footprint de los componente en los lugares deseados de la placa componente.



El programa coloca los componentes de una determinada forma que generalmente no es la que deseamos, aunque dispone de una opción que permite colocarlos de forma automática (Menu Auto->Place->Board) los colocaremos de forma manual.

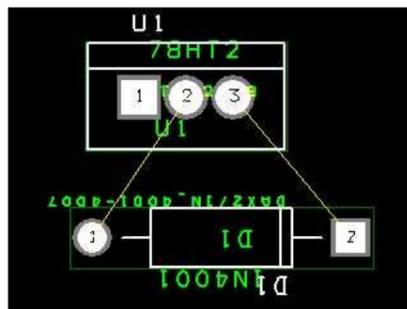
La disposición de los componentes sobre la placa realizada por Layout

no es siempre la mejor.

Los componentes se pueden desplazar por el método habitual de Windows que consiste en marcar y

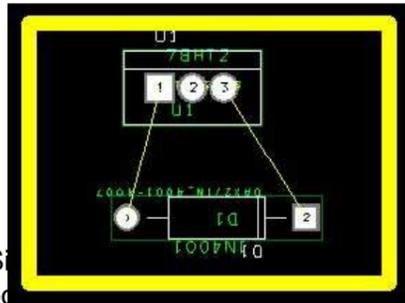
arrastrar con el ratón.

Rotar un componente



El cruce de las líneas del ratsnest en la figura anterior, se puede evitar si se rota el diodo. Para ello basta con marcar el componente y pulsar el botón derecho. En el menú flotante hay que seleccionar "Rotate" o pulsando la tecla R.

5. Establecer los bordes de la placa



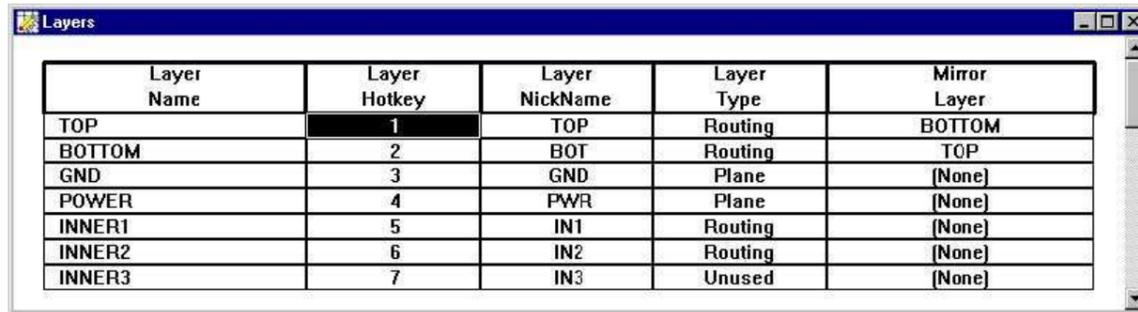
Utilice el comando **Tools->Obstacle->New** y marque con el botón izquierdo los vértices del polígono que delimita la placa. Para terminar, pulse el botón derecho y elija "Finish" en el menú flotante. Observe que el borde tiene el color amarillo correspondiente a la "Global Layer"

Se representa por una línea tan gruesa, estética.

Utilice el comando **Tool->Obstacle->Select Tool** y marque el borde. Después pulse el botón derecho y elija **Properties...** Ahora cambie el parámetro **Width** a su gusto.

6. Habilitación de capas para el trazado de pistas

La habilitación de capas para el trazado de pistas se hace a través de una especie de hoja de cálculo que se puede editar mediante el comando **Tool>Layer->Select from Spreadsheet...**



Layer Name	Layer Hotkey	Layer NickName	Layer Type	Mirror Layer
TOP	1	TOP	Routing	BOTTOM
BOTTOM	2	BOT	Routing	TOP
GND	3	GND	Plane	(None)
POWER	4	PWR	Plane	(None)
INNER1	5	IN1	Routing	(None)
INNER2	6	IN2	Routing	(None)
INNER3	7	IN3	Unused	(None)

En la columna **Layer Type**, con un doble clic de botón izquierdo del ratón, se puede especificar si la capa seleccionada está habilitada para el ruteado (Routing) o no (Unused).

7. Definir el grosor mínimo de las pistas Desde la opción **Tool->Net->Select**

from Spread Sheet.... realice un doble clic con el botón izquierdo del ratón en la

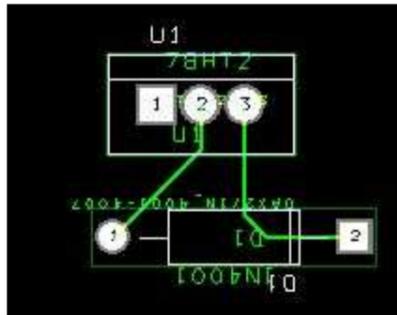
columna **Width** de la hoja de cálculo que aparece, indique el mínimo grosor en la casilla **Min Width**

8. Trazado de las pistas mediante autoroute

Para la realización de trazado de las pistas podemos utilizar las siguientes opciones:

Trazado mediante autoroute

Para trazar las pistas mediante autoroute utilícese el comando **Auto->Autoroute->Board**



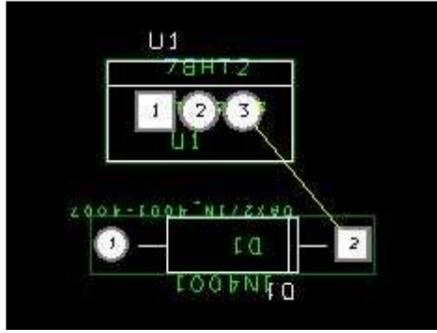
Volver al rat-nest

Para volver al ratsnest utilícese el comando

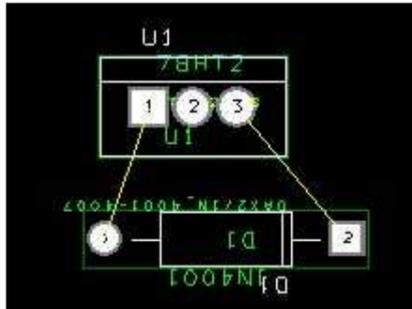
Auto->Unroute->Board

Eliminación de una conexión

Utilice el comando **Tool->Connection->Delete** y marque la conexión del ratsnest que quiera borrar.



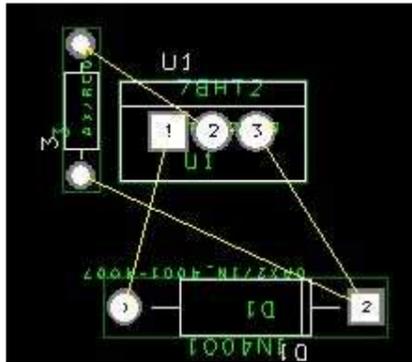
Añadir una conexión



Utilice el comando **Tool->Connection->Add**. Marque con el botón izquierdo el pad 1 del diodo y después el pad 1 del regulador. Para salir del modo de conexión pulse el botón derecho y en el menú flotante elija "End command".

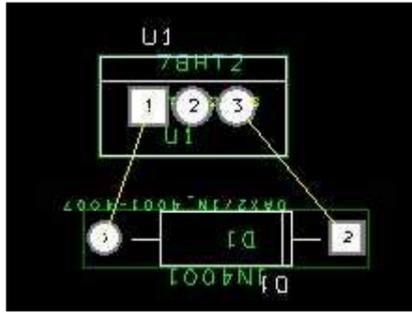
Añadir un nuevo componente

Utilice el comando **Tools->Component->New**. Aparece un cuadro de diálogo donde deberá seleccionar un footprint. Elija, por ejemplo, AX/RC05 de la librería TM_AXIAL.



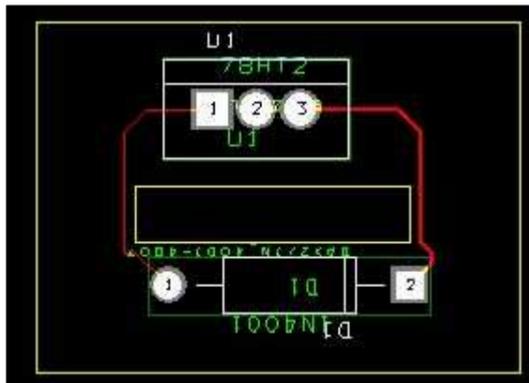
Emplácelo en el espacio de trabajo y trace las conexiones:

Eliminación de un componente



Marque el componente a borrar con el botón izquierdo. Pulse el botón derecho. En el menú flotante elija **Delete**. Elimine ahora el tercer componente.

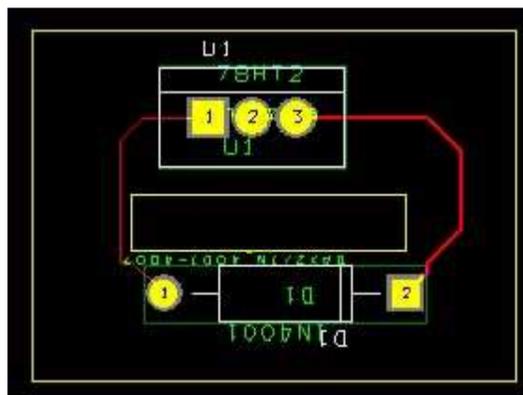
Trazar un obstáculo



Los obstáculos son polígonos que se trazan del mismo modo que los bordes de la placa. Utilice el comando **Tools->Obstacle->New** para trazar un obstáculo similar al que se muestra en la imagen. Después, mediante **Tool->Obstacle>Select Tool** selecciónelo. Pulse el botón derecho y elija **Properties...** en el menú flotante. En **Obstacle type** seleccione "Route keepout". Esto

impedirá que se tracen pistas por el interior del polígono. Compruébelo mediante **Autoroute**.

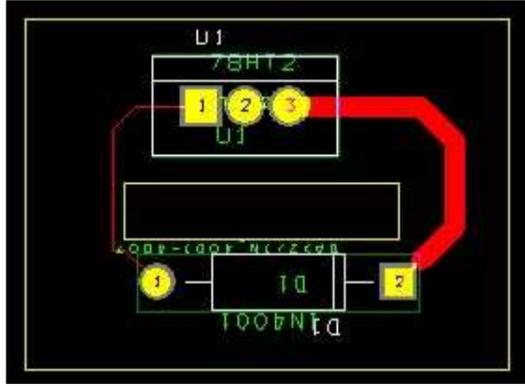
Mover una pista



Para mover una pista utilice el comando **Tool->Track Segment->Select Tool** y marque con el botón izquierdo el segmento que quiera mover. Al mover el ratón, el segmento seleccionado se moverá manteniendo la conexión con los segmentos adyacentes y manteniendo las reglas tecnológicas relativas a ángulos, distancias etc.

Cuando haya desplazado el segmento hasta una posición satisfactoria pulse el botón izquierdo. El segmento quedará fijado y podremos seleccionar otro para moverlo.

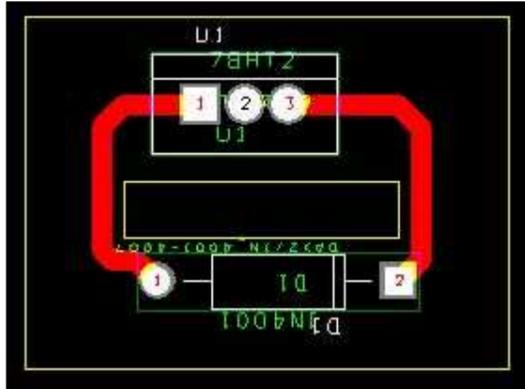
Cambiar el grosor de una pista



Seleccione la pista mediante el comando **Tool->Track->Select**. Después pulse el botón derecho y elija "Change Width" en el menú flotante. Cambie el valor del parámetro **Width**. (El programa le avisa de que no le gusta el cambio. Ignore la impertinencia).

Edite del mismo modo el resto de los segmentos hasta que la pista quede como a usted le guste.

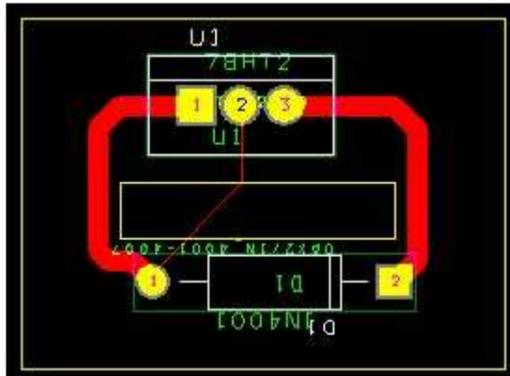
Cambiar el grosor de todas las pistas



Para cambiar el grosor de todas las pistas, primero ejecute **Unroute>Board**. Después **Tool->Net->Select from Spread Sheet...** En hoja de cálculo que aparece pulse dos veces con el ratón en la cabecera de la columna **Width**. Indique el mínimo grosor en la casilla **Min Width**

Vuelva a trazar todas las pistas con **Autoroute->Board**

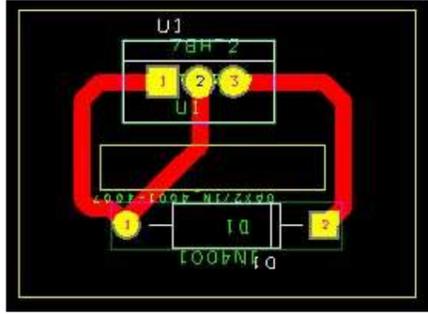
Trazar una pista a mano



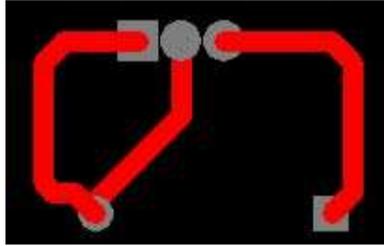
La mejor manera de hacerlo es trazar la conexión y después **Autoroute**; pero también se pueden trazar pistas como obstáculos de tipo **Free Track**.

Utilice el comando **Tool->Obstacle>New** y trace la pista como en la siguiente imagen

Después, edite las propiedades de ese obstáculo y cambie el parámetro **Width**



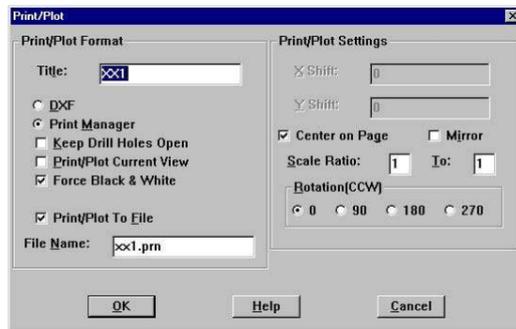
Vista preliminar de la placa en pantalla



El comando **View->Visible<>Invisible** sirve para mostrar u ocultar la capa seleccionada. Así, se puede mostrar en pantalla sólo un subconjunto de las capas.

8. Configuración para imprimir las capas seleccionadas

Para imprimir las capas se utiliza el comando **File->Print/Plot...** . Se imprimirán las capas que se hayan seleccionado.



El formato será DXF (Autocad) o el de la impresora seleccionada por el Windows Manager.

Se puede activar la opción de imprimir a un fichero. P.e. si se quiere tener un fichero PostScript hay que seleccionar una impresora de Windows que tenga un driver PostScript e imprimir a un fichero.

Si seleccionamos la opción **Keep Drill Holes Open**, imprime los pads con la marca para el taladro. Si no se selecciona imprime los pads rellenos.

Para imprimir el acetato final para la realización de la placa fotosensible, utilizar **Options->Post Process Settings...**, seleccionar las caras a imprimir y en Device el dispositivo de salida, (para ver en la pantalla, **Windows->Tile** seleccionar la cara y desde el menú del botón derecho la opción **Preview**) ejecutar la opción **Run Batch** desde el menú del botón derecho