

LIBRERIAS DE CIRCUITOS PARA APRENDER ELECTRONICA MEDIANTE EL SIMULADOR PSPICE

S. Aguinaco, J. Flores, M. Esquisabel y J.J. Zamora
E.T.S.I.I. y I.T- Bilbao; Universidad del País Vasco
Departamento de Electrónica y Telecomunicaciones
Alda. Urquijo s/n - 48013 Bilbao (Vizcaya)
Tel: 4278055, Ext. 2387; Fax.: (94)- 441404
e-mail: jtpzabej@bi.ehu.es

RESUMEN.- En este trabajo se presenta un programa que pone al alcance del alumno las ventajas que supone simular circuitos con una potente herramienta electrónica como es el PSPICE, sin la necesidad de aprender a utilizarlo previamente. El estudiante de electrónica dispone de una *librería de circuitos* que le permite estudiar y analizar la evolución de las señales de forma sencilla.

1.- INTRODUCCION

Los profesionales de la electrónica utilizan los simuladores para verificar el funcionamiento de los circuitos y para optimizar el valor de los componentes, esto hace interesante incluir un simulador en la enseñanza de la electrónica. En este sentido, el programa PSPICE es una herramienta software que tiene cada vez más interés pues permite el análisis de circuitos electrónicos mediante ordenador.

Sin embargo, como ya se apuntó en la ponencia "PSPICE como complemento a una formación básica en electrónica" del congreso TAEE-94 [1], junto a todas las ventajas que lleva consigo la utilización de PSPICE también existen algunos inconvenientes como por ejemplo que, este paquete de software es para uso profesional, y por ello a un alumno que comienza a estudiar electrónica, le puede resultar de difícil manejo. Esto provocará un desaprovechamiento de las posibilidades que ofrece este simulador si no se guía al alumno de la manera adecuada. En esta ponencia, se propuso que el alumno utilizara una documentación realizada específicamente con fines didácticos en vez de los manuales profesionales. En vez de un manual, en nuestro caso se ha desarrollado un programa que se adapta para ejecutarlo sobre el PSPICE.

2.- FUNCIONAMIENTO DEL PROGRAMA

A continuación se realiza una breve descripción de la herramienta desarrollada tal y como la va a ver el usuario.

2.1.- Modos de trabajo del alumno

Cuando el alumno arranca la aplicación se muestra la pantalla de presentación del programa y a continuación, se presenta el menú principal donde el usuario puede interactuar con el

programa a través de la barra de menús. Los principales modos de trabajo del usuario son los siguientes:

- Diseño de un circuito
- Simulación de un circuito
- Recuperación y archivo de un circuito
- Impresión de un diseño

2.2.- Librería de circuitos

En la librería de circuitos del programa, los circuitos están clasificados por etapas o grupos. A través de la barra de menús, el alumno tiene la oportunidad de elegir entre varios circuitos optativos para cada una de las etapas. La diferencia entre estos circuitos está tanto en el formato como en el tipo de componentes utilizados. Dentro de cada etapa, al menos uno de los circuitos está completamente diseñado de manera que sirva para orientar al alumno en su proceso de diseño. Estos circuitos se denominan *circuitos modelo* [2].

Algunas de las etapas escogidas como grupos de estudio son las siguientes:

- Amplificadores
- Filtros
- Osciladores
- Rectificadores ...

El alumno puede elegir entre *Ver* un circuito modelo completamente diseñado o *Diseñar* su propio circuito. En ambos casos primero selecciona la etapa que desea estudiar y a continuación elige entre los circuitos que pertenecen a esa etapa. La Figura 1 representa la pantalla de elección de tipo de oscilador.

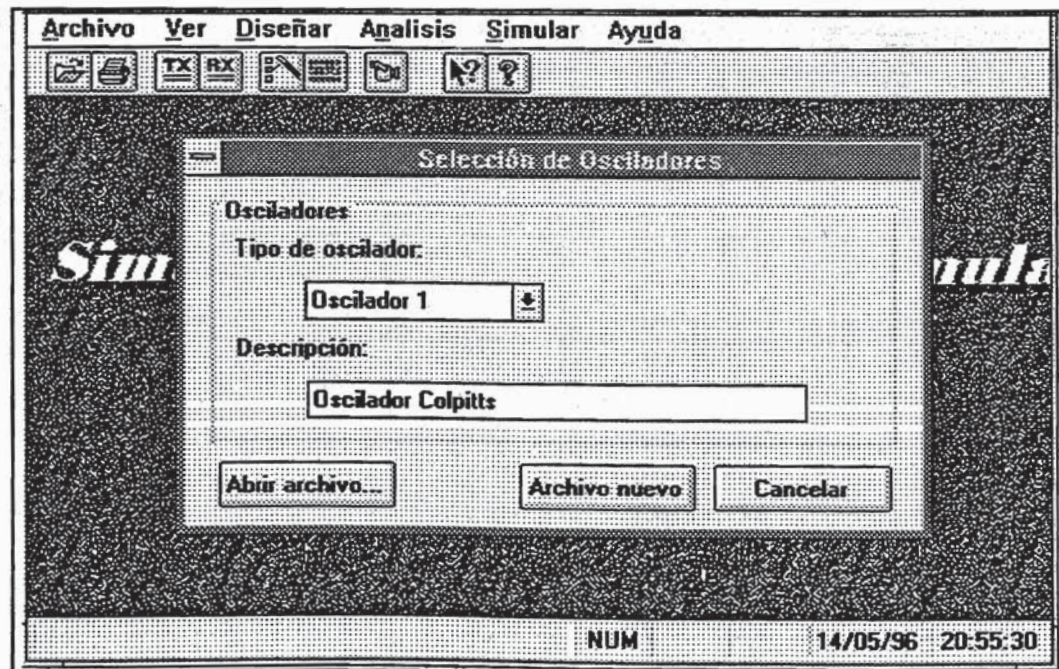


Figura 1.- Pantalla de elección de un oscilador

Una vez seleccionado, el programa presenta en pantalla el esquema del circuito y solicita el valor de los componentes y los análisis que el alumno desea realizar. En la Figura 2, se presenta la pantalla de diseño de un circuito oscilador. Los valores de los componentes se

introducen mediante el teclado. El cursor indica a que componente se le está dando valor y para localizarlo en el esquema, aparece simultáneamente un punto de color junto al componente, que en el caso de la figura está junto a RL. Si algún componente no tiene asignado un valor, el programa no permite la simulación.

Una vez escogida la opción de simular, se solicitan los tipos de análisis que se desean realizar y sus parámetros y a continuación se ejecuta el PSPICE. Cuando la simulación finaliza, los resultados se pueden comprobar mediante el fichero de salida del PSPICE y/o mediante el interpretador gráfico Probe [3]. Durante la ejecución del programa el alumno ve la pantalla de simulación del PSPICE pero no se le permite interactuar con ella, aunque si tiene acceso al programa Probe.

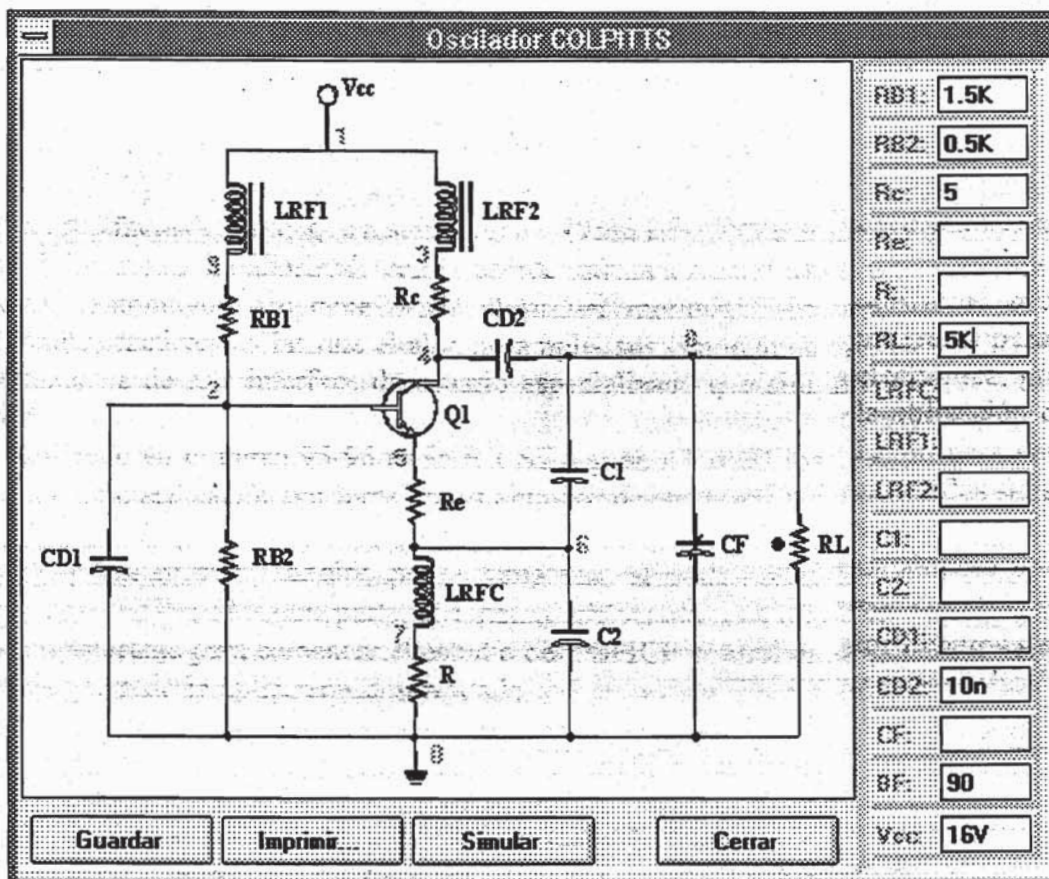


Figura 2.- Pantalla diseño de un circuito oscilador

Si el usuario desea abandonar el programa se pide, antes de finalizar su ejecución, su conformidad ante una eventual pérdida de datos. Si la respuesta es afirmativa cierra todas las ventanas que se encuentren abiertas, elimina los ficheros temporales que se hayan creado y termina la ejecución.

2.3.- Ayuda en ejecución

El objetivo principal de este programa es que sea didáctico y por ello, una parte muy importante y en la que se ha puesto un especial cuidado es el control de mensajes de error o de información y sobre todo la ayuda en ejecución.

La ayuda en ejecución se presenta en tres modos. Estos son:

- Las etiquetas de ayuda sobre los botones de la barra de botones.
- La información que aparece en la barra de estados al pulsar los botones del menú principal de la aplicación.
- La información contenida en el fichero de ayuda del programa, a la cual se accede mediante el índice de la ayuda que se presenta en el menú principal o bien utilizando la ayuda de contexto.

La Figura 3 muestra como se presenta alguno de estos modos de ayuda.

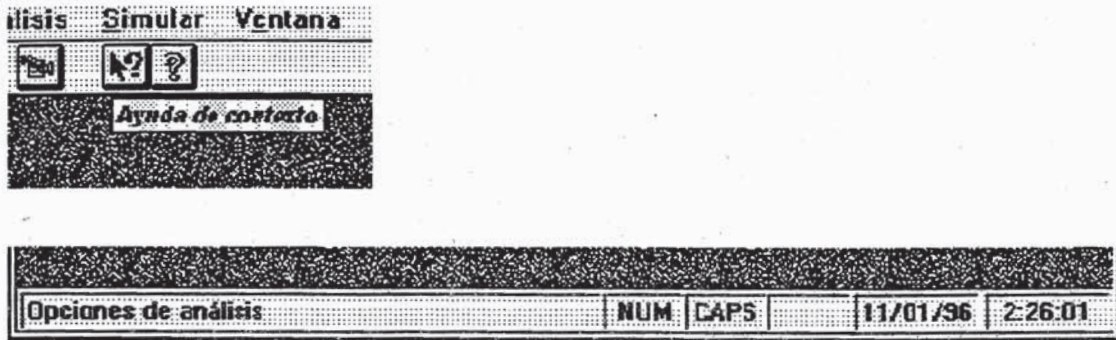


Figura 3.- Modos de ayuda en ejecución

3.- ESTRUCTURA DEL PROGRAMA

3.1.- Requisitos de ejecución

En general, este programa proporciona un interface de usuario en entorno Windows que consta de módulos que se centran en los distintos aspectos de la programación como pueden ser el tratamiento de ventanas, pantallas gráficas, cálculos, etc, en definitiva, todo lo necesario para la ejecución del programa, desde la toma de datos introducidos por el usuario hasta la presentación de los resultados en la pantalla [4].

```
* Netlist *
R_RB1      2 9
R_RB2      8 2
R_Rc       3 4
R_Re       6 5
R_R        8 7
R_RL       8 8
L_RFC      6 7
L_L1       1 9
L_L2       3 1
C_C1       6 8
C_C2       8 6
C_CD1      8 2
C_CD2      4 8
C_CF       8 8
Q_Q1       4 2 5 TRAN
U_Ucc      1 0 dc
.MODEL TRAN NPN(BF= )
```

Además, se ha creado una colección de plantillas que constituyen la *Librería de circuitos* del programa y contiene todos aquellos circuitos que van a ser presentados al usuario. Cada una de estas plantillas es un fichero con extensión *.net* que contiene el netlist del circuito. El netlist de un circuito constituye básicamente los datos que se deben introducir en el simulador, como son la lista de componentes, sus conexiones, sus valores, los análisis a realizar y los parámetros para la simulación. Las plantillas presentan la particularidad de que no contienen los valores de los componentes, sino variables.

Figura 4.- Plantilla de un circuito

3.2.- Ciclo de ejecución del programa

El programa vuelca los datos introducidos por el usuario a la *plantilla* asociada al tipo de circuito seleccionado, genera los ficheros de entrada al PSPICE (.cir) y los simula. El PSPICE genera dos ficheros de salida, uno en ASCII (.out) y otro binario (.dat). Este último es el fichero de entrada al procesador gráfico Probe, que presenta gráficamente los resultados de la simulación. El ciclo completo está representado en la Figura 5.

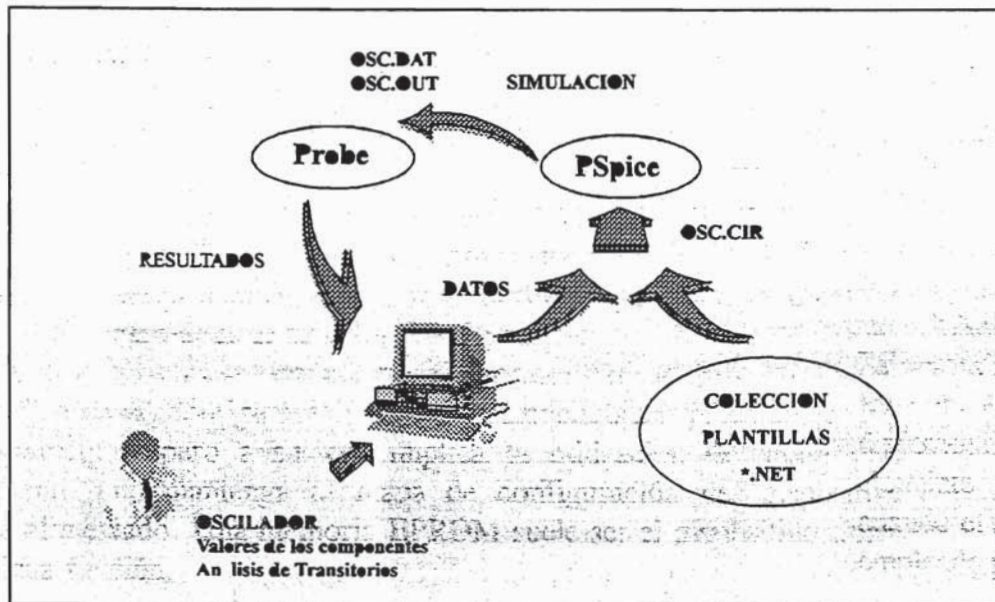


Figura 5.- Ciclo de ejecución del programa

3.3.- Desarrollo del software del programa

El desarrollo del software ha consistido en realizar un interface de usuario que presenta un entorno sencillo y en dotar al programa del soporte necesario para permitir su interacción con la componente electrónica de la aplicación. Esta interacción se traduce en un intercambio de información entre el programa diseñado y las herramientas del simulador PSPICE que se utilizan. La gestión de las distintas funciones que componen esta aplicación la realiza el programa principal. Estas funciones son principalmente las siguientes:

- Lectura y escritura de los ficheros del disco duro
- Control del proceso de diseño de los circuitos
- Simulación de los circuitos diseñados
- Conversión de los ficheros simulados a formato gráfico
- Interface con el usuario: Algoritmos destinados a pedir y ofrecer datos al usuario

Además, el programa principal presenta las opciones de funcionamiento al usuario e invalida las que no pueda escoger en cada momento.

El programa accederá a ficheros distintos, como el fichero de inicialización, los netlist de los ficheros plantilla y los netlist de los modelos. Además guardará los datos introducidos por el usuario y los resultados de la simulación. Desde el menú principal, se presentan una serie de opciones para la ejecución del Probe como pueden ser la elección de las variables de salida,

o si se desea ejecutar el procesador Probe inmediatamente después de la simulación o no. Estos datos se tienen en cuenta al generar el fichero de entrada al procesador gráfico. Por otro lado, esta aplicación se ha desarrollado de forma que la ampliación de la librería de circuitos disponibles en el programa puede realizarse sin apenas variar el código del programa, pudiendo actualizarlo fácilmente.

4.- REQUISITOS

Este programa es una interface de usuario en entorno Windows que esta diseñado para ser ejecutado en ordenadores personales que cumplan los siguientes requisitos: en primer lugar que posean el sistema operativo MS-DOS 5.0 o superior, Windows 3.1 o superior y al menos 20M en el disco duro. Por otro lado para su ejecución será necesaria una librería de enlace dinámico, los controles (.vbx) propios del Visual Basic que es la herramienta de programación utilizada, el fichero de inicialización de la aplicación y el fichero de ayuda.

5. UTILIZACION DEL PROGRAMA

El interfaz de usuario ha sido diseñado de forma que resulta sencillo de manejar, consiguiendo así que cualquier alumno mínimamente familiarizado con el uso de ordenadores sea capaz de utilizar el producto con un rendimiento aceptable desde el primer momento. Así, este programa permite realizar unas prácticas, en las que el alumno amplía sus conocimientos de electrónica, gracias a la existencia de este interface de usuario especializado que simplifica la utilización del simulador.

La primera versión se ha utilizado en el curso 95-96 en la E.T.S.I.I. y de I.T de Bilbao con resultados satisfactorios y una gran aceptación por parte de los alumnos de electrónica, si bien aún continúa ampliándose y depurándose.

El alumno con preparación media se siente atraído por el programa al poder estudiar la evolución de la señal de forma sencilla. Por otra parte, el alumno adelantado puede utilizar los ficheros generados por el programa para comenzar el estudio del PSPICE en profundidad, aunque deberá hacerlo desde fuera de la aplicación desarrollada.

6.- CONCLUSIONES

Como conclusión, el estudiante de electrónica dispone de una librería de circuitos y de un interface de usuario sencillo para ejecutarlos sobre el PSPICE. Con estos dos elementos consigue un ahorro de tiempo pues no necesita capturar los esquemas electrónicos ni aprender el manejo del simulador; solamente tiene que seleccionar el circuito de la librería, asignar valor a los componentes y comprobar los resultados de la simulación.

7.- REFERENCIAS

- [1] Carpeño, A. , López, S.y Arriaga, J. "PSPICE como complemento a una formación básica en electrónica". *TAAE-94*. Universidad politécnica de Madrid, 1994.
- [2] Wesley, A. "Communication circuits: analysis and design". 1978.
- [3] García, E., Ibañez, J. y Gil, L. "PSPICE". Ed. Paraninfo, Madrid 1995.
- [4] Nelson, R. "El libro del Visual Basic para Windows". Ed. Anaya Multimedia, 1994.