

USO DIDÁCTICO DEL SOFTWARE DE AYUDA AL DISEÑO ELECTRÓNICO “PROTEUS”

Mariano Barrón Ruiz

*Departamento de Ingeniería de Sistemas y Automática
Universidad del País Vasco. ispbarum@sb.ehu.es*

RESUMEN

Este trabajo recoge algunas experiencias de uso de un novedoso software de CAD electrónico, obtenidas en las clases prácticas de dos asignaturas de Ingeniería Técnica. *Proteus* es un potente software comercial de captura de esquemas, simulación y auto-ruteado, capaz de simular en tiempo real sistemas completos basados en microcontrolador. El programa tiene un gran interés didáctico debido a su elevado grado de interactividad, ya que permite interactuar con los circuitos mientras se está ejecutando la simulación.

1. INTRODUCCIÓN

El uso de programas de CAD está muy extendido en todas las ingenierías. Para cualquier materia existe la posibilidad de optar por varios paquetes de software de características similares. Por ejemplo, en el campo de la electrónica, excluyendo el software especializado y limitándonos a un tipo de software más genérico, disponemos entre otros de: OrCAD, Protel, Electronics Workbench Multisim, Tango, Circuit Maker, Pads, Microsim, Proteus, etc.

Razones tales como: efectividad, precio, experiencia previa, facilidad de uso, etc., hacen que los profesionales se decanten por una u otra opción. En esta comunicación se exponen los motivos por los cuales utilizamos Proteus, así como sus características más importantes desde el punto de vista didáctico. También se aportan algunos datos sobre las experiencias docentes con este software, obtenidas durante los dos últimos años, en dos asignaturas de Ingeniería Técnica.

2. ¿POR QUÉ PROTEUS?

Proteus es un sistema completo de diseño electrónico que combina un avanzado programa de captura de esquemas, un sistema de simulación mixto (analógico y digital) basado en Spice, y un programa para disposición de componentes en placas de circuito impreso y auto-ruteado. Se trata de un software comercial fabricado por Labcenter Electronics [1], caracterizado por su potencia y facilidad de uso.

La razón principal que nos condujo a experimentar con Proteus fue su capacidad para simular circuitos basados en microcontroladores, con toda la electrónica que rodea a estos diseños (chips, LEDs, teclados, displays LCD, terminales RS-232, etc.), y hacerlo en tiempo real, de forma interactiva, y a partir del fichero ejecutable creado por cualquier ensamblador o compilador. Dicho lo anterior, ya no resulta sorprendente que Proteus también pueda simular diseños que incorporen PLDs y/o memorias ROM o RAM, y que para ello, utilice como entrada los ficheros JEDEC, BIN o HEX generados por cualquier herramienta software.

Desde el punto de vista didáctico, resulta especialmente importante la interactividad que ofrece Proteus. El alumno se interesa más por los circuitos que le permiten mover un potenciómetro, o accionar un teclado o pulsador y observar una respuesta casi real: un LED que se enciende, un display que cambia, un altavoz que emite sonidos, un motor que gira, etc.

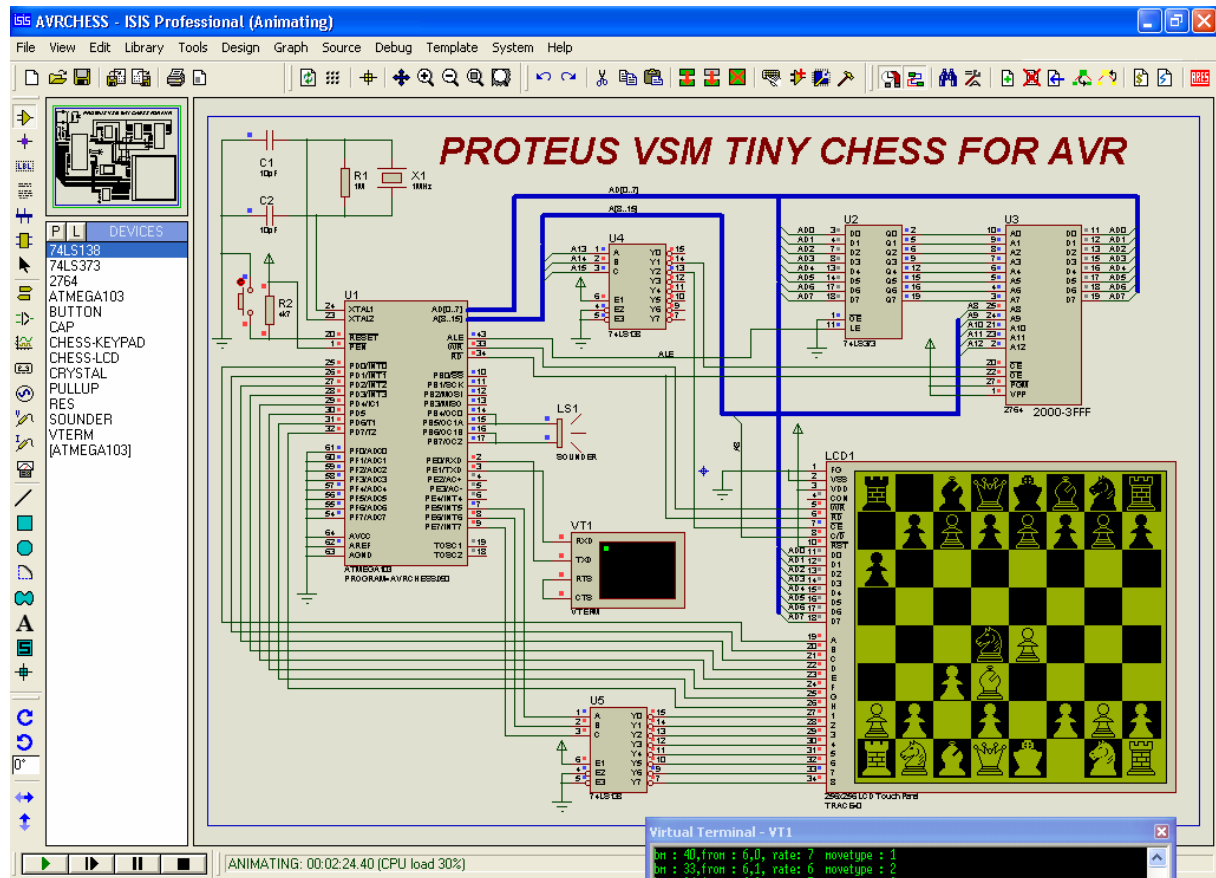


Figura 1: Proteus simulando un complejo sistema basado en microcontrolador que incluye chips MSI, memoria EPROM, LCD gráfico, teclado matricial, terminal RS-232 y altavoz.

3. ALGUNAS CARACTERÍSTICAS DE PROTEUS

Para el usuario, Proteus está formado por dos programas: ISIS (Intelligent Schematic Input System) y ARES (Advanced Routing & Editing Software). La simulación de los circuitos se realiza dentro del propio módulo de captura de esquemas ISIS.

Como programa de captura de esquemas ISIS, tiene todas las características propias de este tipo de software, y otras específicas que le diferencian del resto. Permite realizar diseños multi-hoja, diseños jerárquicos, generación de listados de componentes, generación de netlist en diferentes formatos, etc. Destaca por su: facilidad de uso, calidad de sus esquemas, control total de la apariencia de los dibujos, uso de plantillas para crear esquemas personalizados con un estilo propio, fácil creación de nuevos componentes dentro de ISIS, herramienta visual para asignación de encapsulados a los nuevos componentes, auto-ruteado de los cables de conexión, componentes con buses para reducir el número de hilos, etc.

Dispone de una potente herramienta, el PAT (Property Assignment Tool) que puede programarse fácilmente para realizar tareas repetitivas: asignar, quitar, renombrar, ocultar, mostrar, o cambiar el tamaño a determinadas propiedades de los objetos seleccionados, o al hacer clic sobre un objeto. Algunos usos típicos de esta herramienta son: asignar etiquetas

consecutivas A0, A1, A2,... a determinados hilos de un bus, anotar secuencialmente un conjunto de circuitos integrados, asignar el mismo valor a un conjunto de componentes (P ej. 10 k a un conjunto de resistencias), cambiar el tipo o tamaño de letra de algunos textos, etc.

La simulación se realiza sobre el mismo esquema y dentro de ISIS. El simulador está basado en el SPICE 3F5 desarrollado en la universidad de Berkeley, con extensiones para simulación mixta (analógica y digital) y para el uso de elementos animados e interactivos.

La simulación puede ser: basada en gráficos o interactiva. Para realizar una simulación basada en gráficos, se conectan generadores en las entradas de los circuitos, se editan sus propiedades (amplitud, frecuencia, etc.), se añaden sondas a los puntos que se desea monitorizar, se coloca un gráfico (analógico, digital, de frecuencia, de ruido, etc.) en un área libre del dibujo, se edita el tiempo de inicio y de parada de simulación, se ejecuta la simulación y se obtienen los resultados en gráficos que pueden inspeccionarse con la ayuda de cursores. La simulación interactiva se realiza utilizando en el esquema componentes de dos bibliotecas especiales. Una de ellas, la biblioteca ACTIVE.LIB contiene elementos interactivos (pulsadores, teclados, zumbador, potenciómetro, conmutadores, etc.) y elementos animados (lámparas, semáforo, motores). La biblioteca DISPLAY.LIB dispone de elementos animados de visualización (LEDs, displays de 7 segmentos, displays matriciales, displays numéricos, alfanuméricos y displays gráficos). En simulación interactiva se pueden utilizar los **instrumentos virtuales** disponibles: voltímetro, amperímetro, osciloscopio de doble haz, generador de señal, analizador lógico, generador de patrones, frecuencímetro & contador, y terminal RS-232.

Las bibliotecas, con unos 8000 componentes, no son tan completas como las de otros programas (por ejemplo OrCAD), sin embargo es muy sencillo crear nuevos componentes. La creación de modelos de simulación para los nuevos componentes, es un proceso algo más complicado que puede realizarse a tres niveles: aprovechando los modelos SPICE que proporcionan los fabricantes de dispositivos analógicos; usando primitivas de las bibliotecas DSIMMDLS.LIB o ASIMMDLS.LIB para dispositivos digitales o analógicos; o bien utilizando DLLs para Windows creadas con un lenguaje de programación, en el caso de elementos más complejos como, por ejemplo, un monitor de bus I²C.

La biblioteca DSIMMDLS.LIB contiene algunas primitivas que representan mapas de fusibles de PLDs, y que pueden utilizarse para construir modelos de cualquier PLD. No obstante, la biblioteca PLD.LIB proporciona modelos para algunos de los PLDs más populares (PAL16V8, PAL20V8, PAL22V10, etc.). La información sobre cómo debe programarse el PLD se obtiene de un fichero JEDEC generado por cualquier compilador (PALASM, ABEL, CUPL, OrCAD-PLD, etc.), de esta manera Proteus no obliga a utilizar un determinado software para PLDs, aunque tampoco aporta ninguna solución en este sentido.

4. USO DE PROTEUS EN LAS PRÁCTICAS DE TÉCNICAS DIGITALES

“Técnicas Digitales” es una asignatura obligatoria que se cursa durante el primer cuatrimestre del segundo curso de la titulación de Ingeniería Técnica, en la EUITI de Eibar. Su carga lectiva es de 2 horas de teoría y 1 hora de prácticas semanales. Las clases prácticas se redistribuyen a razón de 2 horas cada dos semanas, para cada grupo de prácticas, lo que equivale a unas 7 clases de 2 horas en total, en todo el cuatrimestre. En un tiempo tan escaso se intenta que los alumnos practiquen con los circuitos secuenciales MSI: biestables, flip-flops, registros de desplazamiento, contadores, etc., PLDs, memorias RAM y ROM, astables

y monostables contruidos con puertas lógicas; que analicen y diseñen máquinas de estados; que utilicen programas de captura de esquemas; que realicen simulación digital y en menor medida analógica; que utilicen el compilador de lógica programable y se familiaricen con la simbología lógica normalizada. Para conseguir tales objetivos, las prácticas deben realizarse casi en su totalidad con la ayuda de un buen simulador, y debe lograrse una muy buena sincronización entre las clases teóricas y prácticas.

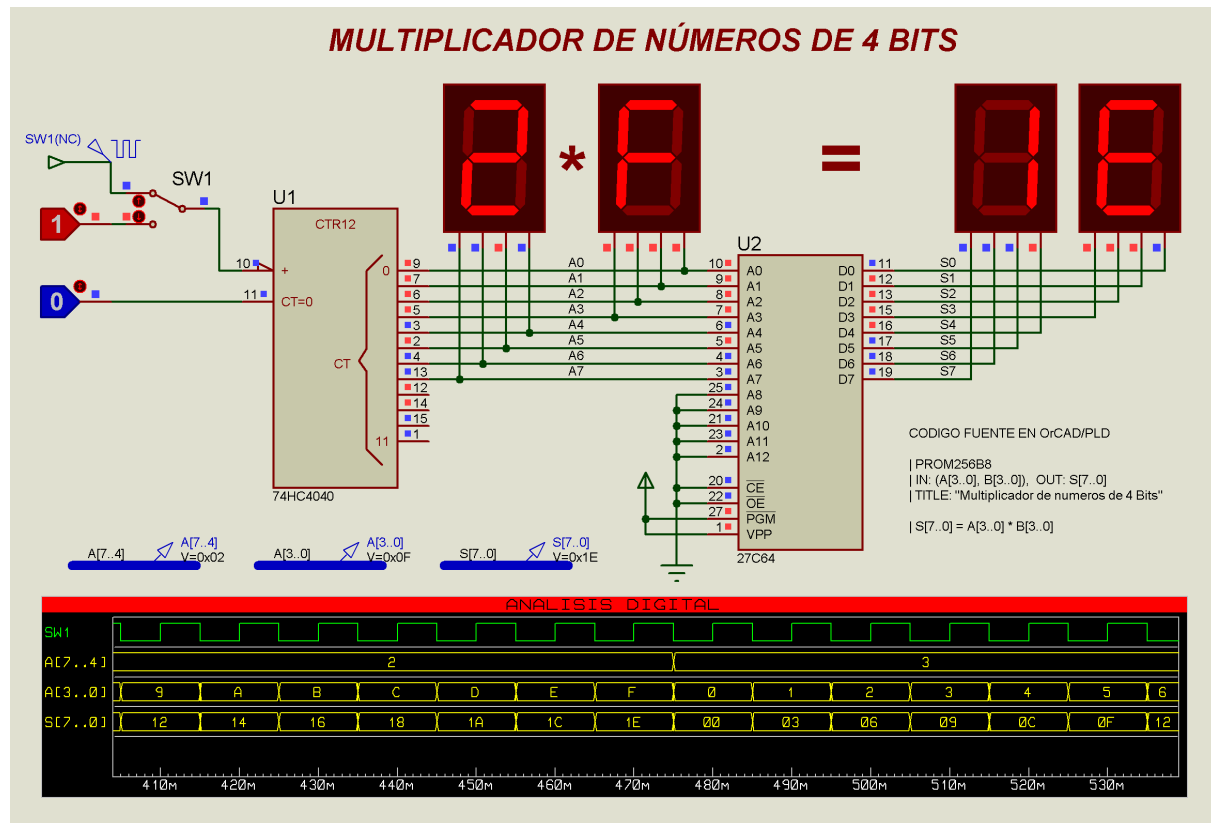


Figura 2: Práctica de simulación con una EPROM programada como multiplicador de 4x4 bits.

El desarrollo de las 7 clases prácticas es el siguiente: En las dos primeras clases se explica el manejo básico del software de captura de esquemas ISIS, la creación de nuevos símbolos, se realiza completamente un esquema de complejidad media, y se practica la simulación interactiva con sencillos circuitos digitales. En la tercera clase se editan y compilan diseños con PLDs ya tratados en las clases de teoría, se utiliza el compilador de lógica programable OrCAD-PLD que soporta PLDs sencillas y memorias PROM. Para ganar tiempo, se prescinde del programa VECTORS, que es un simulador funcional para PLDs que acompaña a OrCAD-PLD, y la simulación de los diseños se realiza con Proteus. La cuarta y quinta clase siguen practicando con los nuevos circuitos secuenciales presentados en las clases teóricas, muchos de ellos se rediseñan dentro de una PAL22V10 y se simulan de forma interactiva y utilizando simulación gráfica. En la sexta clase se utiliza el programa BOOLE-DEUSTO [2], presentado en TAAE'2000 por Javier García Zubía, para la síntesis de máquinas de estados y para la generación del código de las mismas en el lenguaje de descripción de hardware OrCAD-PLD. El código así obtenido se compila, el fichero JEDEC resultante se asigna a un PLD y el circuito completo se simula en Proteus. La última clase se destina a la verificación del funcionamiento de los circuitos astables y monostables realizados con puertas lógicas, utilizando simulación mixta (analógica y digital).

Resumiendo, en todas las clases prácticas se utiliza el software Proteus. En todas las clases se realizan capturas de esquemas y simulación de los circuitos. En varias clases se utiliza el compilador de lógica programable OrCAD-PLD, pero la simulación de los diseños se realiza dentro de Proteus. En la síntesis de máquinas de estados se utiliza el programa BOOLE-DEUSTO. Este programa suele presentarse en la segunda clase, para diseño de circuitos combinatoriales, y luego se retoma para las máquinas de estados.

La experiencia de uso del programa Proteus en la asignatura de Técnicas Digitales es óptima. Los alumnos lo utilizan incluso para los trabajos a realizar en casa, que les piden en algunas asignaturas de electrónica analógica, y ello a pesar de que en éstas asignaturas utilizan otros programas de simulación tales como OrCAD y Multisim.

5. USO DE PROTEUS EN LAS PRÁCTICAS DE MICROCONTROLADORES

“Desarrollos con microcontroladores” es una asignatura optativa que se cursa durante el segundo cuatrimestre por alumnos del tercer curso de la titulación de Ingeniería Técnica, en la EUITI de Eibar. Su carga lectiva es de 2 horas de teoría y 2 horas de prácticas semanales, impartándose las 4 horas en un laboratorio provisto de ordenadores, pizarras y proyector LCD de 1024x728 pixels de resolución.

El objetivo de la asignatura es el desarrollo, hardware y software, de sistemas basados en microcontroladores. Su temario trata sobre la arquitectura típica de un microcontrolador y la de sus periféricos, los circuitos que complementan los diseños con microcontroladores y la programación de los mismos, fundamentalmente en lenguaje C.

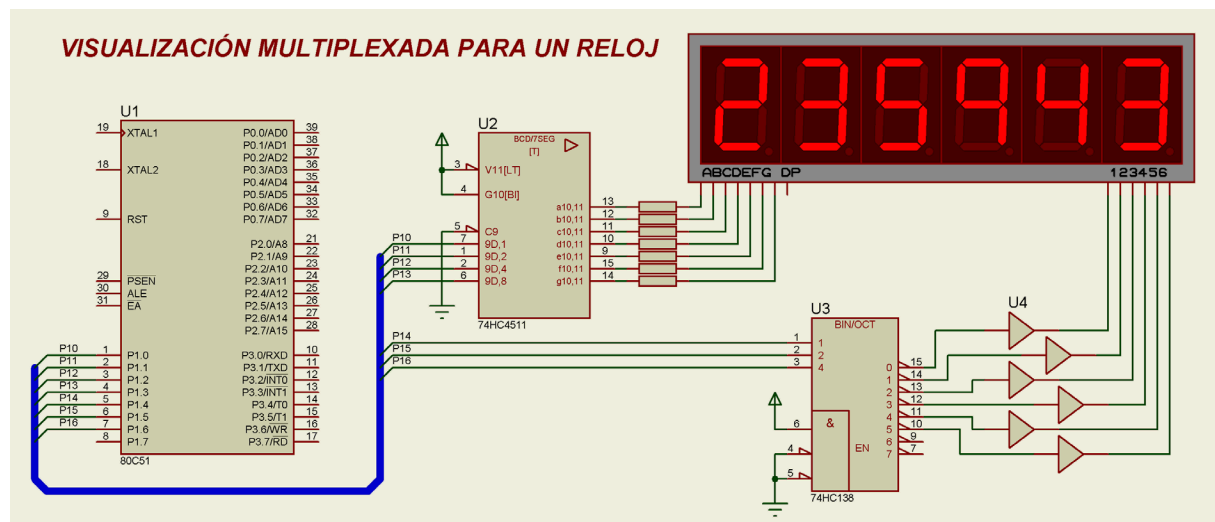


Figura 3: Práctica de visualización multiplexada con un microcontrolador 8051.

Las prácticas contemplan: la ampliación del número de patillas de entrada y salida de los micros, uso de LEDs y pulsadores, visualización dinámica en displays de 7 segmentos, uso de las interrupciones, captura y generación de señales analógicas, manejo de LCDs alfanuméricos, lectura de teclados matriciales, generación de música y sonidos, comunicación serie RS-232 e I²C, y control de velocidad de motores de CC mediante salidas PWM.

Todas las prácticas comienzan con la explicación del hardware necesario, y su plasmación en un esquema realizado con Proteus. La programación del microcontrolador se realiza utilizando el compilador de Keil [3] para la familia Intel 8051, utilizando lenguaje C y ocasionalmente lenguaje ensamblador. Se practica la depuración de tres formas distintas:

utilizando el simulador y el depurador Dscope de Keil; mediante depuración remota a través del puerto serie del PC, tras descargar el código objeto sobre la tarjeta de prácticas a la que se han conectado todos los periféricos; y finalmente usando la capacidad de Proteus para simular el sistema completo, incluyendo toda la periferia, y permitiendo la interacción del usuario.

La integración de Proteus con el software de Keil es absoluta. Ambos programas pueden comunicarse a través de TCP/IP, corriendo en el mismo PC o en ordenadores diferentes. De esta forma Proteus se comporta como un emulador virtual de sistemas completos, permitiendo que el depurador de Keil controle la sesión de emulación, corriendo los programas paso a paso o con puntos de parada, tanto en lenguaje ensamblador como en lenguaje C.

La mayoría de las aplicaciones desarrolladas con microcontroladores pertenecen a la categoría de los sistemas reactivos, o sistemas cuyo comportamiento viene determinado por sucesos externos. Estos sistemas reaccionan ante un evento exterior ejecutando la acción apropiada. En tales sistemas es muy importante disponer de una herramienta de simulación interactiva como Proteus. Lamentablemente el software no cuenta con modelos para todos los microcontroladores del mercado, pero sí dispone de los más utilizados en nuestras escuelas. En la actualidad soporta algunos miembros de las familias: Intel 80C51, Motorola 68HC11, Microchip y Atmel.

En el caso de que se disponga de una tarjeta de prácticas en el laboratorio, se puede llevar a un esquema y entregárselo a los alumnos, para que realicen prácticas fuera del laboratorio. De esta forma podrán comparar los resultados obtenidos en casa, bajo un entorno virtual, con los obtenidos en el laboratorio utilizando un hardware real.

La experiencia del uso de Proteus en las clases prácticas con microcontroladores es excelente. Utilizando esta herramienta, el desarrollo de las clases se aproxima bastante a una experiencia de diseño con microcontroladores. Se parte de un esquema en blanco donde se plasma un hardware, se escribe la aplicación en lenguaje C y se depura en un entorno muy parecido a la realidad teniendo la posibilidad de interactuar con el sistema.

6. REFERENCIAS

[1] Labcenter Electronics (www.labcenter.co.uk)

[2] J. García Zubía. “Adecuación de los entornos computacionales a la clase de Electrónica Digital: Boole-Deusto”. Actas TAAE’2000 pg. 573-576 (Barcelona 2000).

[3] Keil Software Inc (www.keil.com)