

DISEÑO Y EVALUACIÓN DE UN EQUIPO DIDÁCTICO INTERACTIVO: HARD Y SOFT ORIENTADO A LA IMPLEMENTACIÓN DE SISTEMAS DIGITALES COMPLEJOS

Rosado, Luis* - Villasevil, F. Javier** - López Antonio M**

*Dpto. de Inteligencia Artificial U.N.E.D.
C/ Senda del Rey - Ciudad Universitaria 28040 - MADRID
Tel.- (91) 3987158; Fax.- (91) 3986697, Email: rosado@dia.uned.es
Pág. Web: <http://www.dia.uned.es/~rosado>

** Departamento de Ingeniería Electrónica U.P.C.
Av./Victor Balaguer, s/n 08800 - Vilanova i la Geltrú
Tel.- (93) 8967728, Fax.- 8967700
Email:villasevil@eel.upc.es - Email:lopezm@eel.upc.es

RESUMEN

En la enseñanza de Sistemas Digitales complejos, las herramientas que tradicionalmente se utilizan no permiten la interacción con el alumno. La herramienta de simulación creada se ha pensado de modo que permita abordar el estudio de una forma más interactiva. Se ha escogido una estructura típica de máquina algorítmica, que permite la materialización de diferentes algoritmos con tan sólo cambiar el programa de control. El alumno, mediante la utilización de este equipo, puede simular un sistema digital complejo (ej. microprocesador sencillo) para luego implementarlo prácticamente. Se ha evaluado la eficiencia del equipo con resultados satisfactorios frente a otros procedimientos tradicionales.

1. INTRODUCCIÓN

Las recientes teorías de aprendizaje propugnan que el conocimiento es algo que cada individuo reconstruye; por tanto, el conocimiento no se adquiere por mera transmisión. En consecuencia, se sostiene que las estrategias de aprendizaje más efectivas son las que explotan el principio de "aprender haciendo"; en este principio se basa nuestro equipo.

En el aprendizaje de Electrónica Digital se empieza por dar al alumno las bases matemáticas que rigen la lógica binaria. Luego se enseñan diseños más o menos sencillos de sistemas

combinaciones. Y más tarde se estudian sistemas secuenciales de pequeña complejidad con métodos clásicos, que son poco útiles cuando se trata de abarcar sistemas más complejos; y en consecuencia, nace la necesidad de encontrar métodos alternativos. En esta línea, nuestro equipo didáctico, basado en sistemas multimedia interactivos, supone un apoyo importante.

2. OBJETIVOS

El sistema multimedia interactivo que hemos diseñado y construido, es capaz de:

- Proporcionar nuevos elementos de motivación para el aprendizaje.
- Conferir un rol activo al alumno.
- Mejorar los resultados educativos mediante la aplicación de nuevo material que provoque nuevas propuestas metodológicas.
- Establecer una relación profesor - alumno que libere al profesor de funciones de simple informador.
- Garantizar la práctica de una educación individualizada y ayudar al alumno a construir sus propios conocimientos.

3. DISEÑO DEL EQUIPO

El sistema de Simulación *Software* del funcionamiento de una Máquina Algorítmica es un programa que aporta al alumno, de una forma sencilla, la posibilidad de implementar un algoritmo concreto (planteado por el mismo alumno) y ejecutarlo bajo un marco gráfico secuencial. En una estructura fija de máquina algorítmica se visionan de forma consecutiva los diferentes procesos que se van dando en el sistema, a la vez que se tiene constancia del contenido de sus elementos básicos de almacenamiento de información, así como de los resultados. Esta estructura se visualiza en la figura 1.

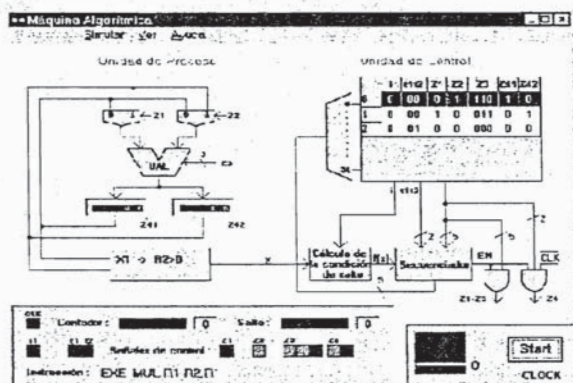


Fig. 1. Detalle ventana del programa.

El programa se ha estructurado de manera que contribuye a ilustrar el proceso de funcionamiento de una máquina algorítmica. Siguiendo esta filosofía, se han diferenciado claramente las diferentes partes que la componen: Unidad de Proceso, Unidad de Control, Memoria, etc.. Además se ha considerado oportuno visualizar el contenido de los registros y de las direcciones de memoria en cada uno de los períodos de reloj de la máquina; así como resaltar la evolución de cada uno de estos ciclos de reloj. Todo ello contribuye a la comprensión del funcionamiento de la máquina, puesto que se van visualizando los cambios, así como las transferencias de información en cada uno de los ciclos de reloj.

El sistema *hard* asociado al programa, parte de la misma filosofía de separación clara de cada una de las partes que definen este tipo de máquinas; así como de su función en el conjunto. Así se ha diferenciado muy claramente, igual que en el programa, la unidad de proceso de la unidad de control.

La unidad de proceso es diseñada y montada por el alumno; mientras que el *hard* diseñado en el equipo didáctico implementa la unidad de control de una máquina algorítmica capaz de controlar cualquier unidad de proceso. Para ello se le ha dotado de un programa específico en función de la unidad de proceso a controlar. Este programa puede ser modificado y gravado en memoria por el propio alumno; de manera que no es un equipo estático, sino que el alumno trabaja conceptos importantes en él.

Un detalle de la placa de *hard* es la observada en la Fig 2.

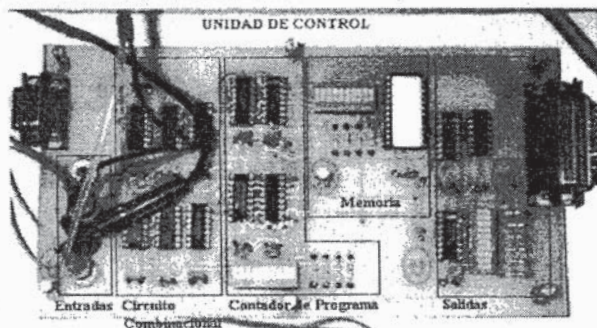


Fig 2. Placa "hardware".

La unidad de control está constituida con una estructura PLA; es decir, esta formada por una matriz OR y una matriz AND.

La matriz AND recibe dos tipos de entrada para decodificar las direcciones:

- Los estados de la máquina secuencial.
- Las variables de condición, generadas por la unidad de proceso.

La matriz OR está formada por la memoria propiamente dicha, y en ella se almacenan dos tipos de información.

- La dirección de la siguiente instrucción.
- Los valores de las variables de control que habilitan la ejecución de las operaciones incluidas en la instrucción en curso.

El inconveniente de este tipo de estructura, es que la memoria necesaria para implementar la matriz OR ha de ser de un tamaño considerable, ya que cada fila ha de contener una dirección y un vector Z de variables de control.

Existen dos métodos principales que permiten reducir el tamaño de la memoria de la unidad de control :

1. Mediante decodificación de órdenes.
2. Con secuenciador.

El primero de los dos métodos se basa en la probabilidad de que el número de vectores Z diferentes sea mucho menor que 2^m . Si esto es así, podrían almacenarse los distintos vectores Z de manera codificada con un número de bits inferior. Para obtener el vector Z completo a partir del contenido de la memoria se utiliza un circuito decodificador, que genera las variables de control Z a partir de las órdenes en forma codificada.

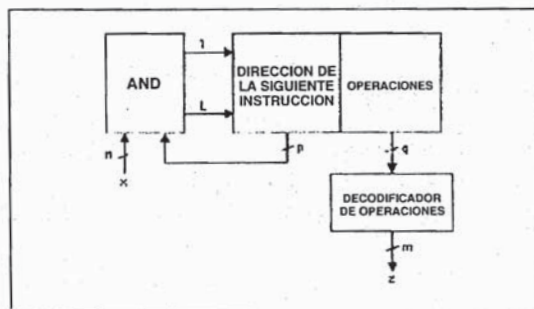


Fig. 3 Estructura unidad de control

El segundo método, con secuenciador, se basa en la reducción de la zona de la memoria que contiene la dirección de la próxima instrucción. Este será el método utilizado para el diseño de nuestra unidad de control y se describe a continuación.

En la matriz OR, el primer campo contiene de forma explícita, la dirección de la siguiente instrucción. Mediante la utilización de un secuenciador será posible eliminar este campo.

En un programa de control, lo normal es que un gran número de instrucciones no produzcan una bifurcación o salto durante la ejecución. Es decir, existirán secuencias de instrucciones

que se ejecutarán una detrás de otra, secuencialmente, por lo que en muchos casos la dirección y^+ de la siguiente instrucción a ejecutar será la dirección de la instrucción en curso más 1.

$$y^+ = y + 1 \quad [1]$$

Este razonamiento sugiere la utilización de un contador de programa, constituido por un circuito capaz de calcular la dirección de la siguiente instrucción a ejecutar. De esta manera no tendrá que estar almacenado en memoria.

El circuito realiza la siguiente operación:

$$y = \text{LOAD} \cdot E + \overline{\text{LOAD}} \cdot (y + 1) \quad [2]$$

Para su funcionamiento necesitaremos una señal de carga (LOAD) que indicará al contador del programa si la siguiente dirección ha de calcularla ($y + 1$) o la ha de cargar de la memoria (E).

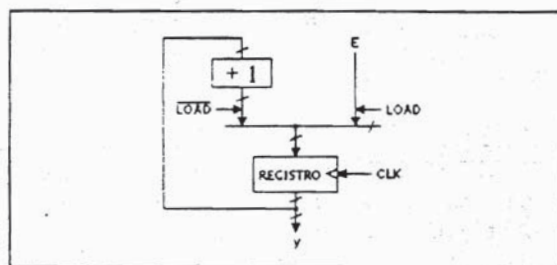


Fig. 4 Contador del programa

Si $\text{LOAD} = 0$ se está ejecutando una instrucción sin bifurcación, por lo que la dirección de la siguiente instrucción será calculada por el contador de programa y $y = y + 1$.

Si por el contrario, $\text{LOAD} = 1$, la dirección de la siguiente instrucción está almacenada en la memoria y el contador de programa simplemente la carga en el registro.

En una instrucción de bifurcación o salto, no se ejecuta ninguna operación en la unidad de proceso, las variables de control son 0 y no necesitan ser almacenadas en la memoria. Por lo tanto podemos utilizar el campo destinado a las variables de control para almacenar la dirección de la siguiente instrucción, reduciendo así el tamaño de la memoria.

4. ORIGINALIDAD

Es un equipo totalmente diseñado y evaluado en el aula por nuestro grupo de trabajo. Permite el seguimiento paso a paso del flujo de las señales del proceso que diseña el alumno, previa simulación. Esto aumenta la comprensión del alumno, no sólo en la utilización de sistemas digitales más complejos (microprocesadores, autómatas, etc.), sino también en su diseño.

5. CONCLUSIONES

Un aspecto fundamental en la labor educativa es conseguir aplicar la metodología adecuada para que el alumno mantenga una actitud activa durante el proceso de aprendizaje. El estudiante que se habitúa a un modelo de enseñanza basado casi exclusivamente en la transmisión verbal / escrita, normalmente adopta posturas pasivas, y su único objetivo es la productividad, entendida ésta como el resultado positivo en los exámenes. Su preocupación se centra en aplicar estrategias de estudio orientadas no a profundizar y solidificar sus conocimientos, sino con el fin de sacar un buen rendimiento en las pruebas escritas.

Para actuar sobre ello, y así corregir la actitud del alumno y no solo mejorar su rendimiento sino también solidificar sus conocimientos y obtener de él una actitud más participativa:

- Se ha realizado un equipo de enseñanza aplicado al campo de sistemas digitales para el diseño de estructuras complejas. Consta de una parte de programa interactivo y una parte circuital para comprobar físicamente que el resultado es el mismo que se consiguió en la simulación con el programa.

Se ha aplicado este equipo didáctico a la realidad del aula; evaluando su utilidad mediante herramientas de medida generadas específicamente para ello y tratando posteriormente los resultados estadísticamente se ha confirmado su utilidad de manera objetiva.

Con ello hemos confirmado que:

- Se ha conseguido realizar una enseñanza más participativa e individualizada.
- Se ha mejorado el rendimiento académico.
- Se ha liberado al profesor de su acción de simple informador.
- El equipo se adecua a las necesidades del aula en la enseñanza de la Electrónica Digital en Ingeniería.
- Hemos conseguido integrar teoría y práctica en el proceso de aprendizaje.

6. BIBLIOGRAFÍA

- [1] D. J. Comer. "Digital Logic and State Machine Desing". 2d. ed. Philadelphia: Saunders, 1990.
- [2] S. Devadas. et al. " State Assignment of Finite-State Machines Targeting Multi-Level Logic Implementations". *IEEE Transactions on Computer-Aided Desing*. Vol.7, pp 1290-1300. Diciembre, 1988.