

Un Constructor Virtual y Simulador Lógico para el Descubrimiento y Diseño de Circuitos Digitales con Enfoque en Proyectos de Ingeniería

Arturo Miguel-de-Priego
Chincha, Perú
arturo@tourdigital.net

Resumen—Se describe un programa de computadora para construir y simular circuitos digitales empleando modelos lógicos de circuitos integrados estándares, de aplicación específica, circuitos programables, diagramas esquemáticos y diagramas de flujo. Los circuitos pueden dibujarse y simularse con escenarios virtuales como si se estuviera en el mundo real. Además de brindar más oportunidades de diseño, ahorro de tiempo y dinero, y reutilización de los circuitos para enseñanza y aprendizaje, el programa ayuda a aprender conceptos y métodos de diseño desde diferentes perspectivas tecnológicas y facilita el aprendizaje basado en indagación científica y proyectos de ingeniería. Por ejemplo, es posible tanto experimentar virtualmente con circuitos integrados para descubrir patrones y principios lógicos como desarrollar aplicaciones prácticas para optimizar flujo de vehículos, ahorrar energía en el suministro de agua, descubrir leyes de cinemática, construir juegos didácticos, supervisar ambientes, etc. Versiones previas de este programa han sido compartidas a través de la Internet y son utilizadas en varias escuelas, institutos y universidades alrededor del mundo.

Palabras claves— aprendizaje basado en proyectos; diseño digital; entornos virtuales; aprendizaje basado en indagación.

I. INTRODUCCIÓN

Dada la presencia de las tecnologías digitales en casi todos los dispositivos modernos es importante entender cómo funcionan los sistemas digitales tanto en su constitución como en su operación [1]. En el nivel universitario, los cursos de introducción al diseño digital muestran los principios, técnicas y aplicaciones prácticas de estas tecnologías y típicamente abarcan los circuitos estándares y de lógica programable junto con la descripción de hardware. En cursos siguientes se abordan el diseño y programación de microprocesadores y microcontroladores y la estructura y arquitectura de computadoras. Los cursos más avanzados tocan los temas de codiseño de hardware y software, sistemas embebidos y procesamiento digital de señales.

La mayoría de centros de enseñanza de países en vías de desarrollo enfrenta dificultades en el aula y en el laboratorio: fallas mecánicas y eléctricas en la construcción de los circuitos, limitada disponibilidad de componentes e instrumentos de medida, así como errores conceptuales, confusiones acerca de las tecnologías y escasas aplicaciones prácticas [2, 3].

Con respecto a la enseñanza, las técnicas de instrucción frecuentemente están en conflicto con los estilos individuales

de aprendizaje y en muchos casos la instrucción todavía sigue siendo centrada en el docente y manejada por contenidos. Una instrucción más efectiva puede planificarse con actividades de aprendizaje que constructivamente promuevan niveles altos de pensamiento tales como el enfoque en proyectos o el basado en indagación [4, 5, 6].

En cuanto a la tecnología, las herramientas de diseño provistas por fabricantes y vendedores tienen limitaciones pedagógicas y pueden parecer complejos en un primer curso de circuitos digitales debido a las múltiples opciones de diseño.

Gracias principalmente a la Internet, a las múltiples corrientes de oportunidades de aprendizaje (cursos masivos MOOC, repositorios OCW, centros de enseñanza y aprendizaje CTL), al relativo bajo precio y disponibilidad de tecnología (tarjetas de desarrollo, sensores) resulta viable y práctico resolver muchos de los problemas que enfrentan un gran número de sociedades, varios de ellos requiriendo la creación de circuitos, sistemas y aplicaciones digitales.

El propósito del programa descrito en este documento es ayudar a los profesores y estudiantes en los cursos de introducción al diseño de circuitos digitales proporcionando un medio de construcción virtual y de validación por simulación del funcionamiento de los circuitos lógicos interactuando con escenarios o ambientes virtuales. Esta aproximación permite reducir el tiempo de diseño y pruebas, aislar las fallas lógicas y eléctricas, incrementar las posibilidades de diseño, y reducir el tiempo de interferencia en los ambientes.

Versiones previas de este programa han sido compartidas a través de la Internet [7] y son utilizadas en escuelas, institutos y universidades de América Latina, Israel, Egipto y de varios países de Europa. En Perú también ha sido utilizado en varios talleres escolares de ingeniería en el colegio nacional José Pardo y Barreda de Chincha y en cursos de universidades particulares y estatales.

En la sección 2 se describen las características básicas del programa y se mencionan sus aplicaciones para la enseñanza y aprendizaje en relación con varias tecnologías modernas. En la sección 3 se ilustra cómo puede utilizarse esta herramienta en un curso de diseño digital con énfasis en el aprendizaje basado en indagación y con un enfoque en proyectos de ingeniería. En las conclusiones de la sección 4 se resaltan las bondades del programa y se indican algunas ampliaciones para mejorar su funcionalidad.

II. CARACTERÍSTICAS DEL SOFTWARE

Este programa permite dibujar en el computador circuitos digitales utilizando modelos de circuitos integrados (estándares y ASIC), protoboards, tableros de entrada (interruptores, pulsadores) y de salida (indicadores luminosos, visualizadores de siete segmentos), y simularlos con escenarios virtuales y analizadores lógicos. Una fuente de voltaje provee VCC de 5V y GND. Un temporizador genera señales de 1, 2, 5 y 10 Hz. En la Fig. 1 pueden observarse varios de estos elementos. Las posiciones de los componentes y las conexiones entre ellos se establecen utilizando el ratón. El programa evita cortocircuitos y posiciones inapropiadas de los chips. Todos los chips están modelados sin retardos de propagación de señales.

Los modelos de circuitos integrados incluyen modelos lógicos de los circuitos estándares TTL y CMOS (puertas lógicas, sumadores, multiplexores, decodificadores, flip-flops, unidades lógico-aritméticas, registros), microprocesadores (MIPS básico, GNOME), microcontroladores y circuitos ASIC (controladores de luces de tráfico, parqueo, brazo robot, elevador; cronómetro, radio-alarma, frecuencímetro, juegos). Este conjunto de componentes permite a los profesores y estudiantes crear programas y circuitos para aplicaciones típicas que requieran operaciones lógico aritméticas y escritura y lectura de datos.

Los escenarios virtuales modelan el ambiente de operación de los circuitos. Pueden trabajar en dos modos: interactivo con el usuario e interactivo con los circuitos. Ejemplos de escenarios son: cruce de avenidas, tanques de agua, elevadores, robots, etc. Este método reduce los riesgos operativos, el tiempo de diseño y pruebas y el costo total del diseño. La Fig. 2 muestra un controlador de tráfico con un escenario de cruce de avenidas que cuenta con sensores que pueden ser utilizados en un nuevo circuito para optimizar el flujo de vehículos.

Los analizadores lógicos proveen nueve canales para capturar y visualizar señales lógicas. Operan en dos modos: asíncrono para capturar señales cuando cambia cualquiera de ellas y síncrono para capturar señales cuando cambia la señal del noveno canal.

Este programa incluye símbolos lógicos, diagramas de flujo y dispositivos lógicos programables (PLD). Estos elementos poseen retardos de propagación de pin a pin de 100 milisegundos para poder analizar retardos en puertas y circuitos realimentados. Los símbolos lógicos incluyen puertas lógicas, flip-flops, multiplexor, decodificador; funciones programables; y sumadores, contadores, comparadores y registros de 4 bits.

Los diagramas de flujo permiten describir funciones de dos y tres variables mediante expresiones lógicas seleccionadas en las casillas de decisión y de asignación. Sirven para determinar funciones equivalentes de puertas básicas, latches y flip-flops. Conectando varios diagramas entre sí es posible definir funciones lógicas más complicadas. Esta característica ayuda a introducir la descripción algorítmica de hardware.

Los modelos de circuitos PLD están representados por diagramas esquemáticos donde se realizan las interconexiones directamente. Se incluye un arreglo lógico programable (PLA) y un dispositivo lógico programable simple (SPLD). Ambos circuitos son útiles para ilustrar el funcionamiento de los

circuitos y para diseñar rápidamente circuitos lógicos que luego pueden ser llevados a circuitos de lógica estándar. La Fig. 3 ilustra la configuración de un sumador completo y de un contador síncrono de 4 bits. Esta aproximación ayuda a introducir la tecnología de lógica programable.

Además, el programa cuenta con tutoriales específicos para oportunidades de diseño típicas tales como experimentos de cinemática, estática, dinámica, robótica, electricidad y sensores de humedad, presión y temperatura. El programa está diseñado con orientación a objetos para poder agregar fácilmente más componentes, tutoriales y escenarios.

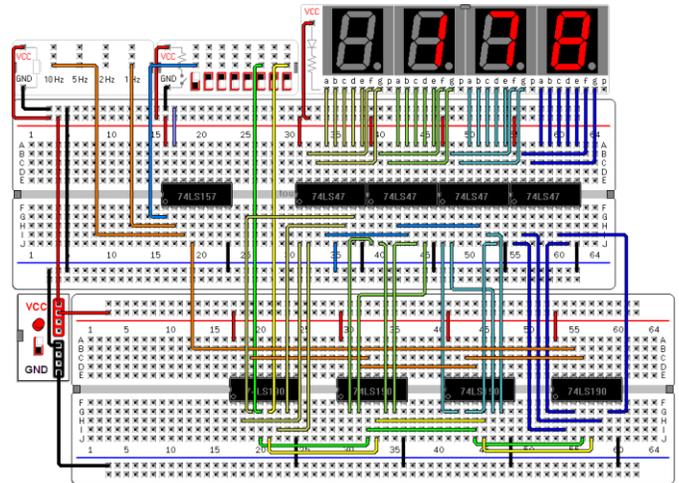


Fig. 1. Un cronómetro para experimentos de cinemática.

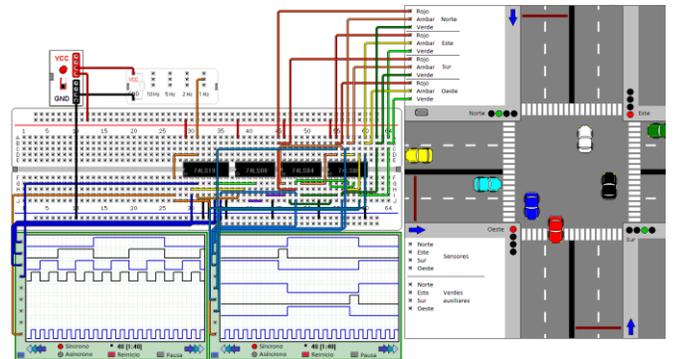


Fig. 2. Un circuito de control de tráfico en un cruce de avenidas.

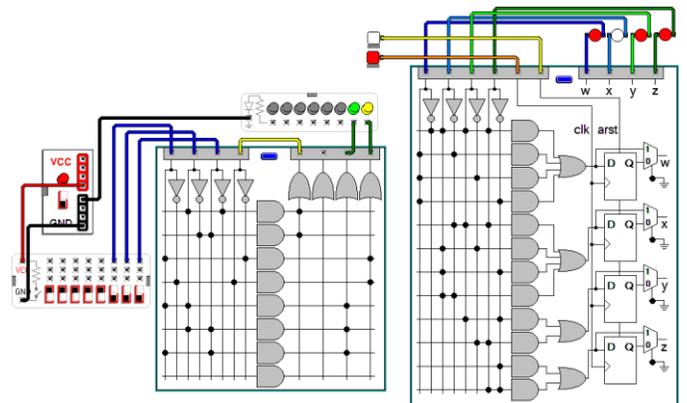


Fig. 3. Un sumador completo en PLA y un contador de 4 bits en PLD.

III. APLICACIONES DEL PROGRAMA

Este programa puede utilizarse tanto en el nivel escolar superior como en el nivel universitario y tecnológico. La Fig. 4 muestra un esquema para un curso de introducción al diseño de circuitos digitales en tres fases: análisis, diseño y proyecto. Las barras celestes en las casillas indican una medida relativa del uso de esta herramienta para aprender los conceptos y entender las aplicaciones en cada tema. Este esquema permite emigrar a herramientas de diseño industrial de una manera más clara, sólida y sencilla.

La primera fase de instrucción inicia con el análisis de puertas básicas, latches y flip-flops para entender los conceptos de retardos y funciones universales con lógica estándar y brindar una visión general de la lógica programable y la descripción de hardware. Se continúa con los fundamentos del álgebra de conmutación y de síntesis lógica y luego se analizan circuitos contadores, sumadores, comparadores, multiplexores, descodificadores y codificadores.

En la fase de diseño se aplican los conceptos de máquinas de estados finitas y algorítmicas para construir circuitos más avanzados, como microprocesadores y microcontroladores, mientras los estudiantes desarrollan niveles de pensamiento más abstractos con lenguajes de descripción de hardware y síntesis sobre dispositivos de lógica programable de tipos CPLD y FPGA. Al finalizar esta fase se presenta el estado del arte del diseño y de las tecnologías digitales.

En la fase de proyecto los estudiantes trabajan en equipo para tratar problemas reales del entorno local con soluciones prácticas. Los estudiantes adquieren más responsabilidades por sus aprendizajes y el docente es tanto un asesor como compañero de diseño: brinda guías para planificar y ejecutar los proyectos y prepara tutoriales, herramientas y escenarios virtuales para ayudar a entender a los estudiantes el problema y las soluciones que están abordando. El diseño en ingeniería entonces se utiliza para realizar circuitos óptimos y prácticos utilizando componentes virtuales primero y tecnología real después.

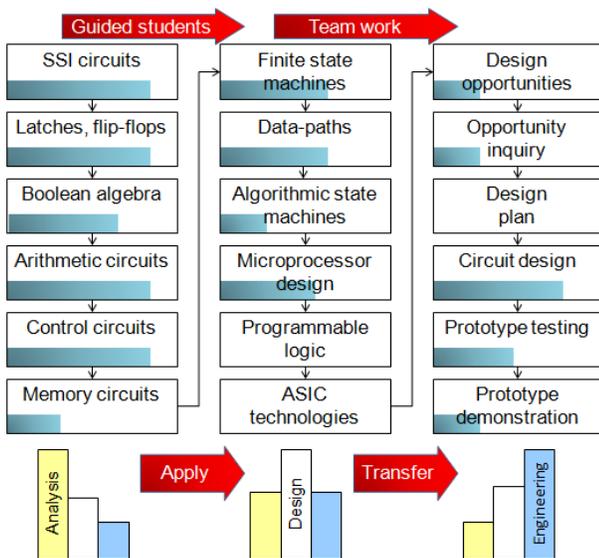


Fig. 4. Una organización de temas para un primer curso de diseño lógico.

A. Aprendizaje basado en indagación

Los métodos de indagación científica pueden ser aplicados en el aula para aprender inductiva y experimentalmente el funcionamiento de los circuitos integrados y de los circuitos digitales. Al ser entrenados con técnicas de análisis y de diseño los estudiantes van adquiriendo autonomía intelectual y responsabilidad por sus propios aprendizajes. Especialmente, el método de instrucción 5E [8] puede utilizarse en las sesiones de clases para aprender a descubrir el funcionamiento de los circuitos lógicos y pensar en aplicaciones. Este método de cinco fases consiste en: **emocionar** a los estudiantes para despertarles la curiosidad e interés en un tema específico y sacar a la luz sus entendimientos actuales; **explorar** mediante cuestionamientos y experimentos para conocer más sobre el tema; **explicar** con sus propias palabras y diagramas lo que se va comprendiendo; **elaborar** un plan de acción para aplicar lo aprendido en nuevos contextos; y **evaluar** el entendimiento conseguido y las habilidades desarrolladas con el nuevo concepto, fenómeno o dispositivo.

La Fig. 5 muestra un caso de estudio de la función NAND: 1) se presenta un circuito de alarma activado por láser para captar el interés en el tema; 2) los estudiantes analizan un circuito digital básico para obtener la tabla de verdad de la función NAND; 3) explican sus hallazgos con pseudocódigos y diagramas de flujo; 4) elaboran circuitos basados en NAND para obtener otras funciones lógicas y se les reta para mejorar el diseño del circuito de la alarma original; y 5) junto con el docente, evalúan y comparan sus aprendizajes compartiendo ideas y resultados a la vez que planifican nuevos diseños.

Acorde con la clasificación de Bloom, al concluir esta sesión se espera que los estudiantes **conozcan** la operación de puertas lógicas y latches, **entiendan** cómo los circuitos integrados digitales pueden realizar funciones lógicas; **apliquen** sus conceptos para descubrir por sí mismos más circuitos lógicos; **analicen** circuitos esquemáticos sencillos para determinar las tablas de verdad y diagramas de tiempo; **sinteticen** circuitos prácticos basados en latches y puertas básicas; **evalúen** sus progresos y entendimientos; y **valoren** tanto su esfuerzo como sus logros de aprendizaje.

Esta aproximación permite formar una base común de conocimientos, habilidades y entendimientos en los estudiantes enseñando el diseño lógico mediante varias representaciones y de acuerdo a los estilos de aprendizaje e intereses de los propios estudiantes. Con un constante entrenamiento en este método, los estudiantes pueden desarrollar transversalmente sus habilidades de comunicación, razonamiento lógico-matemático, diseño en ingeniería, cultura tecnológica, indagación científica.

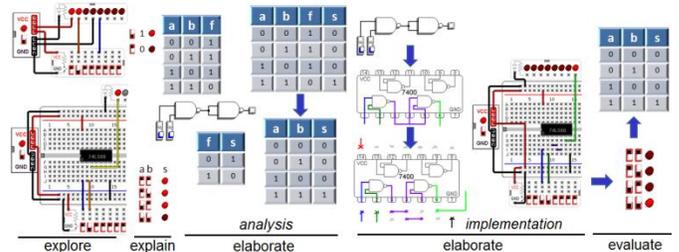


Fig. 5. Descubrimiento de la función NAND y diseño básico de circuitos.

B. Aprendizaje basado en proyectos

Las actitudes de los estudiantes hacia el aprendizaje y emprendimientos responsables pueden ser impulsadas a través del aprendizaje con enfoque en proyectos. Para los cursos de ingeniería puede utilizarse un proceso de diseño en ingeniería de diez pasos [9] que inicia con una búsqueda de oportunidades de diseño reales, del propio entorno local, y concluye con una presentación pública y demostración del funcionamiento del producto o prototipo.

El programa facilita la incorporación de este método en un primer curso de diseño digital. Por ejemplo, el docente puede diseñar entornos virtuales de las situaciones reales para que los estudiantes identifiquen y analicen las condiciones operativas. Cuando una especificación esté lista, el docente puede crear circuitos ASIC para validar las soluciones propuestas por los estudiantes. Luego los estudiantes pueden diseñar sus soluciones con varias tecnologías (estándar, FPGA, ASIC, microcontrolador), métodos de descripción (modular, algorítmica, simbólica, textual) y de verificación. Por ejemplo, la Fig. 6 muestra un diseño con chips TTL para el control del nivel de agua en un tanque pero la misma función puede ser realizada con circuitos FPGA o microcontroladores. En casos así, las restricciones de diseño determinan la solución idónea.

Para ayudar a los estudiantes a entender más sobre la naturaleza del problema y sobre las tecnologías apropiadas, el docente puede desarrollar tutoriales interactivos. Por ejemplo, la Fig. 7 muestra un tutorial para deducir la ley de caída libre y otro para descubrir leyes de circuitos eléctricos. Actualmente estos tutoriales pueden incorporarse en el programa accediendo al código fuente.

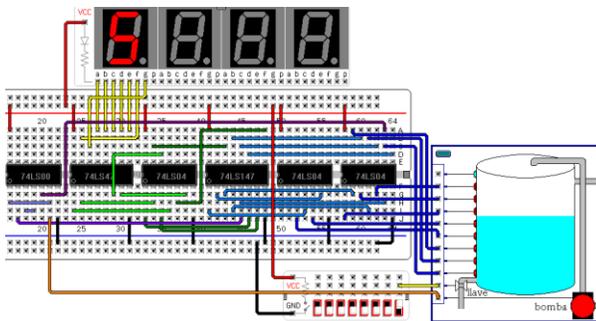


Fig. 6. Un controlador del nivel de agua en un tanque.

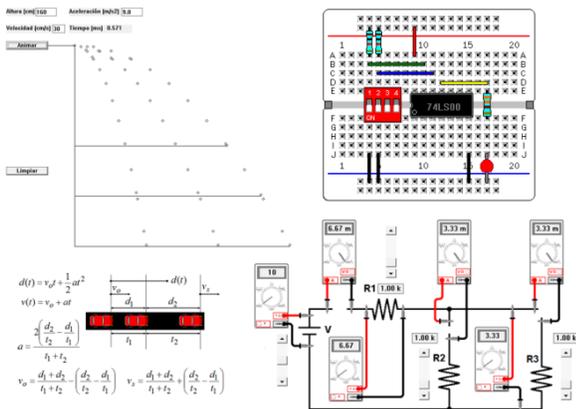


Fig. 7. Tutoriales para experimentos de cinemática y de electricidad.

IV. CONCLUSIÓN Y TRABAJO FUTURO

Se ha descrito un programa de computadora para construir y simular circuitos lógicos con varios tipos de representaciones de circuitos: diagramas esquemáticos, circuitos integrados estándares, circuitos ASIC, dispositivos lógicos programables y diagramas de flujo. Además de las aplicaciones típicas de un primer curso de circuitos digitales este programa facilita el estudio de circuitos más complejos tales como máquinas de estados, microprocesadores y microcontroladores.

Se han indicado características especiales del programa para entender la temporización y el retardo de propagación gracias a los retardos por los símbolos lógicos y a los analizadores lógicos con modos de captura síncrono y asíncrono. También, las diferentes representaciones de funciones lógicas y las distintas tecnologías permiten reforzar los conceptos de diseño, abordar los problemas desde varias perspectivas y probar soluciones alternas.

La herramienta facilita a los instructores las evaluaciones formativas y la aplicación de métodos de aprendizaje basados en proyectos a lo largo del curso y en indagación en el aula de clase, y también ayuda a ajustar los contenidos y actividades de acuerdo al ritmo de aprendizaje y a los estilos individuales y grupales de aprendizaje.

Actualmente se está trabajando en una versión para incorporar circuitos analógicos y para ampliar el número de escenarios y de componentes para simulaciones y pruebas interactivas a través de los puertos del computador y mediante la Internet.

REFERENCIAS

- [1] National Research Council. 2004. *The Engineer of 2020: Visions of Engineering in the New Century*. Washington, DC: The National Academies Press, pp 14-15. www.nap.edu/catalog/10999.html
- [2] Herman, Geoffrey and Loui, Michael. "Identifying the Core Conceptual Framework of Digital Logic". In 119th ASEE Annual Conference. 10-13 June 2012. www.asee.org/public/conferences/8/papers/4637/view
- [3] Goldman, K., Gross, P., Heeren, C., Herman, G., Kaczmarczyk, L., Loui, M., and Zilles C. "Identifying Important and Difficult Concepts in Introductory Computing Courses using a Delphi Process". SIGCSE 2008, www.cs.illinois.edu/~zilles/papers/delphi.uiuc-tr2917.pdf
- [4] National Research Council. 2000. *How People Learn: Brain, Mind, Experience, and School: Expanded Edition*. Washington, DC: The National Academies Press. www.nap.edu/catalog/9853.html
- [5] Intel Education, intel.com/education
- [6] Science Buddies, www.sciencebuddies.org
- [7] Simulador de circuitos digitales, www.tourdigital.net
- [8] BSCS 5E Instructional Model, bscs.org/bscs-5e-instructional-model
- [9] Intel Design and Discovery, intel.com/education/design