

PROTOTIPO DIDÁCTICO DE SUMADOR BINARIO

DIDACTIC PROTOTYPE OF A BINARY ADDER

Cerón, Víctor^a, Pérez, Óscar Jesús^b, Rendón, Hugo Emmanuel^c, Reyes, Evelyn Monserrat^d, Rufino, Luis Osvaldo^e, Santiago, Jessica Aketzali^f.

^{a,b,c,d,e,f} Tecnológico Nacional de México/ITS del Occidente del Estado de Hidalgo. División de Ingeniería en Tecnologías de la Información y Comunicaciones. Mixquiahuala de Juárez Hidalgo. México. 42700. xxxxx@itsoeh.edu.mx*.

RESUMEN. Este informe presenta el diseño e implementación de un sumador binario completo de 1 bit, tanto en simulación digital como en hardware físico, así como el desarrollo de una herramienta en Java que convierte números decimales a binario y viceversa. El proyecto integra conocimientos de Fundamentos de Programación (algoritmos y Java), Matemáticas Discretas (lógica booleana), Cálculo Diferencial (modelado de señales), Introducción a las TICs (electrónica digital) y Ética profesional. Se modeló la señal de salida $V(t)$ del sumador como una onda cuadrada (función escalón). La simulación en Tinkercad permitió validar la lógica del circuito, y el programa Java, usando divisiones repetidas o métodos nativos (p. ej. `Integer.parseInt`), realiza la conversión decimal-binario correctamente. Se empleó control de versiones con GitHub para coordinar el trabajo en equipo, mejorando la productividad y calidad del software.

Palabras clave: lógica booleana, simulación digital, algoritmos.

ABSTRACT. This report presents the design and implementation of a 1-bit full binary adder, both in digital simulation and physical hardware, as well as the development of a Java tool that converts decimal numbers to binary and vice versa. The project integrates knowledge from Programming Fundamentals (algorithms and Java), Discrete Mathematics (Boolean logic), Differential Calculus (signal modeling), Introduction to ICT (digital electronics), and Professional Ethics. The adder's output signal $V(t)$ was modeled as a square wave (step function). The simulation in Tinkercad enabled the validation of the circuit logic, and the Java program, using repeated divisions or native methods (e.g., `Integer.parseInt`), performs the decimal-binary conversion correctly. Version control with GitHub was used to coordinate teamwork, improving software productivity and quality.

Key words: boolean logic, digital simulation, algorithms.

INTRODUCCIÓN

La aritmética binaria es fundamental en los sistemas digitales modernos y constituye la base del procesamiento computacional actual. A pesar de su importancia, la comprensión de cómo interactúan las señales físicas con la lógica abstracta suele representar un desafío en la formación de ingeniería. Bajo este panorama, se plantea la hipótesis de que la integración de un prototipo físico con modelado matemático y herramientas de software facilita la asimilación de estos conceptos complejos³.

Un sumador binario completo es el circuito encargado de agregar tres bits de entrada (A, B y acarreo entrante C_{in}) produciendo un bit de suma S y un acarreo saliente C_{out} . Para su análisis riguroso, la señal eléctrica digital asociada puede modelarse idealmente como una onda cuadrada.

En este proyecto, se propone modelar matemáticamente la función de salida $V(t)$ utilizando la función escalón unitario, propia del Cálculo Diferencial, conectando así la teoría de señales con la implementación física. Adicionalmente, se aborda la conversión numérica mediante una herramienta en Java que emplea algoritmos modulares y métodos nativos.

El desarrollo integra conocimientos multidisciplinarios: Fundamentos de Programación (estructuras de control y arreglos), Matemáticas Discretas (lógica booleana), Introducción a las TICs (diseño lógico) y Taller de Ética, aplicando además control de versiones con GitHub para la gestión colaborativa.

El objetivo de este trabajo es diseñar e implementar un prototipo didáctico de sumador binario completo

de 1 bit, validado mediante simulación y modelado matemático, junto con una herramienta de software para la conversión de sistemas numéricos⁴

METODOLOGÍA

El desarrollo del proyecto se realizó bajo un enfoque de **Investigación Tecnológica Aplicada**, siguiendo las etapas del **Ciclo de Diseño en Ingeniería**. Esta metodología estructura el trabajo en fases secuenciales e interdependientes: fundamentación teórica y modelado matemático, validación mediante simulación digital, implementación física del prototipo y desarrollo de software complementario. A continuación, se detalla el procedimiento realizado en cada una de estas etapas.

Diseño lógico y modelado matemático

En la fase inicial, se estableció la tabla de verdad del sumador completo de 1 bit. A partir de ella, se dedujeron las expresiones booleanas para la salida de suma S y el acarreo de salida C_{out} , presentadas en las Ecs. (1) y (2):

$$S = A \oplus B \oplus C_{in} \quad (1)$$

$$C_{out} = (A \wedge B) \vee (C_{in} \wedge (A \oplus B)) \quad (2)$$

Para vincular la lógica digital con el análisis de señales, se modeló la señal de salida $V(t)$ como una función escalón unitario que alterna entre los niveles lógicos de 0 V y 5 V. Considerando una señal ideal de onda cuadrada con ciclo de trabajo del 50%, el comportamiento se define en la Ec. (3):

$$V(t) = 5 \cdot u(t \bmod T < T/2) \quad (3)$$

Donde $u(t)$ representa la función escalón unitario. Este modelo permitió contrastar teóricamente las mediciones físicas posteriores.

Simulación e implementación física

Previo al montaje, el circuito se validó en un entorno de simulación digital (Tinkercad), verificando que las salidas coincidieran con la tabla de verdad para todas las combinaciones de entrada⁵. Posteriormente, se procedió a la construcción física en protoboard utilizando componentes lógicos (compuertas discretas o CI 74HC83), interruptores para las entradas y LEDs para visualizar los estados lógicos. El funcionamiento se corroboró midiendo los niveles de tensión resultantes con un voltímetro digital, asegurando la correspondencia con el modelo teórico.

Desarrollo de software y colaboración

Paralelamente, se desarrolló una herramienta de consola en Java para la conversión entre sistemas numéricos. El algoritmo implementado aplica el método de divisiones sucesivas por 2, almacenando los residuos en un arreglo para conformar la cadena binaria⁶. El código incluye validación de entradas y el uso de métodos nativos como Integer.parseInt para la conversión inversa.

La gestión del ciclo de vida del software siguió buenas prácticas de control de versiones mediante un repositorio en GitHub. El equipo trabajó sobre ramas individuales, realizando confirmaciones frecuentes (*commits*) y revisiones de código cruzadas (*pull requests*), lo que optimizó la integración de funciones y la calidad del código⁷. Finalmente, la organización del equipo se rigió por principios éticos profesionales, estableciendo roles definidos y fomentando la responsabilidad compartida hacia el cumplimiento de los objetivos del proyecto.

RESULTADOS Y DISCUSIÓN

Funcionamiento del Circuito y Simulación

El circuito sumador completo diseñado operó de acuerdo con las especificaciones teóricas. La simulación digital validó la integridad lógica del diseño; por ejemplo, al aplicar las entradas $A=1$, $B=1$, $C_{in}=0$, se obtuvo consistentemente una salida de Suma $S=0$ y Acarreo $C_{out}=1$, coincidiendo con la tabla de verdad esperada¹². Como se observa en la Figura 1, la implementación mediante compuertas XOR para la suma aritmética y la red AND/OR para el acarreo permitió minimizar el retardo de propagación lógico, verificando los fundamentos expuestos en la literatura especializada¹³. ShutterstockExplorar

Figura 1. Diagrama lógico del sumador completo de 1 bit implementado. La configuración de compuertas XOR y AND/OR garantiza la generación correcta de S y C_{out} según la aritmética binaria.

Modelado de Señales

El análisis de la señal de salida $V(t)$ demostró congruencia con el modelo matemático propuesto de onda cuadrada ideal. La Figura 2 ilustra el comportamiento de la tensión, que alterna entre niveles de 0 V y 5 V con un ciclo de trabajo del 50%. Este resultado confirma que la función escalón unitario es una herramienta matemática adecuada para describir transiciones en circuitos digitales, tal como se establece en la teoría de señales¹⁴. La

estabilidad observada en los estados alto y bajo valida la robustez del prototipo frente a ruido eléctrico menor.

(Aquí deberías insertar tu gráfica de la onda cuadrada)

Figura 2. Gráfica de la señal de salida $V(t)$. Se aprecia la alternancia entre niveles lógicos alto y bajo, correspondiendo al modelo matemático de funciones escalón desplazadas.

Herramienta de Software

El algoritmo de conversión implementado en Java demostró ser preciso y eficiente. En las pruebas de escritorio, la conversión del decimal 13 a binario mediante divisiones sucesivas arrojó correctamente la secuencia de residuos inversa (1, 0, 1, 1), resultando en el binario 1101_2 . La comparación de este método algorítmico con la función nativa Integer.parseInt arrojó resultados idénticos, lo que valida la lógica de bucles implementada por los estudiantes. El código, estructurado y documentado, facilita la comprensión del proceso aritmético subyacente.

Gestión del Proyecto y Ética

La metodología de trabajo colaborativo mediante GitHub resultó en un historial de desarrollo transparente y organizado. El uso de ramas separadas para nuevas características y la revisión por pares mediante pull requests incrementaron la calidad del código final, alineándose con las mejores prácticas de la industria. Finalmente, la dinámica de equipo reflejó los principios de ética profesional de la ACM, evidenciando una distribución equitativa de responsabilidades y un compromiso con la integridad académica y el bien público, sin que se presentaran conflictos significativos durante el desarrollo.

CONCLUSIONES

Redactar las conclusiones más significativas sin repetir literalmente el contenido del resumen.

AGRADECIMIENTOS Y/O RECONOCIMIENTOS

Puede escribir aquí la fuente de financiamiento de su investigación o reconocer la contribución de personas o instituciones que aportaron en la investigación. Punto a considerar según el autor.

REFERENCIAS

Las citas se escribirán en letra normal arial tamaño 8. Debe incluir trabajos científicos publicados evitando incorporar el material no disponible abiertamente como tesis doctorales no publicadas y actas de congresos o similares. En esta sección se detallarán los trabajos a los que se hizo referencia en el artículo y deberán ir numerados en orden de aparición.

Los ejemplos a continuación son para artículo, capítulo de libro, memorias de congreso y sitios web, respectivamente. Los autores son responsables de la precisión de las citas utilizadas.

1. Ghisi, E., & Tinker, J. A. (2005). An ideal window area concept for energy efficient integration of daylight and artificial light in buildings. *Building and Environment*, 40(1), 51-61.
2. Jeantet, R., Croguennec, T., Schuck & P., Brulé, G. (2006). *Ciencia de los alimentos*. Acribia. Zaragoza, España. pp 22.
3. Nava-Gómez, B., Valenzuela-Espinoza, E., Sampedro-Ávila, J.E., Bonett-Calzada, B. y León-Mancilla, S. (2013). *XX Congreso Nacional de Ciencias y Tecnología del Mar*. DGE CyT. Los Cabos, B.C. México.
4. Saunders, L. (1994). Beverage creation. Design elements. Disponible en: www.foodproductdesign.com/archive/1994/0494DE.html. Accesado: 24 agosto 2005