

# Diseño de una biblioteca de compuertas MCML utilizando un algoritmo genético y optimización multiobjetivo

Design of an MCML gate library using a Genetic Algorithm and Multi-objective Optimization

*Roberto Pereira-Arroyo<sup>1</sup>  
Alfonso Chacón-Rodríguez<sup>2</sup>*

*Fecha de recepción: 25 de abril del 2014  
Fecha de aprobación: 17 de junio del 2014*

Pereira-Arroyo, R; Chacón-Rodríguez, A. Diseño de una biblioteca de compuertas MCML utilizando un algoritmo genético y optimización multiobjetivo. *Tecnología en Marcha*. Vol. 27, N° 4, Octubre-Diciembre. Pág 41-48.

- 1 Escuela de Ingeniería Electrónica, Instituto Tecnológico de Costa Rica. Cartago, Costa Rica. Correo electrónico: rpereira@itcr.ac.cr
- 2 Escuela de Ingeniería Electrónica, Instituto Tecnológico de Costa Rica. Cartago, Costa Rica. Correo electrónico: alchacon@itcr.ac.cr

## Palabras clave

Algoritmo genético; optimización multiobjetivo; lógica en modo de corriente; automatización del diseño electrónico.

## Resumen

En este documento se alude al problema de dimensionamiento de circuitos MCML (*MOS Current Mode Logic*). Se introduce el Frente de Pareto como una herramienta de análisis útil para explorar el espacio de diseño de las distintas compuertas que conforman nuestra biblioteca básica MCML. Un algoritmo genético (GA) es implementado para detectar automáticamente este frente, en un proceso que busca eficientemente las parametrizaciones óptimas del diseño y sus correspondientes valores en un espacio de aptitudes. Las mediciones del consumo de potencia, el retardo de propagación y el rango del voltaje de salida se usan como funciones de aptitud, puesto que el problema se trata como una tarea de optimización multiobjetiva. Finalmente, se presentan los resultados de las simulaciones *postlayout*, usando la tecnología de fabricación AMS 0,35  $\mu\text{m}$ .

## Key words

Genetic algorithm; multi-objective optimization; current mode logic; electronic design automation.

## Abstract

In this paper, the problem of sizing MOS Current Mode Logic (MCML) circuits is addressed. The Pareto front is introduced as a useful analysis tool to explore the design space of each gate that is part of our MCML basic library. A genetic algorithm (GA) is employed to automatically detect this front in a process that efficiently finds optimal parameterizations and their corresponding values in an aggregate fitness space. Measures of the power consumption, propagation delay and output voltage swing are used as fitness functions, since the problem is treated as a multi-objective optimization task. Finally, the results of postlayout simulations, using the AMS 0.35  $\mu\text{m}$  technology are presented.

---

## Introducción

Este documento introduce una estrategia de optimización automatizada que se aplica para el diseño de circuitos *MOS Current Mode Logic* (MCML, por sus siglas en inglés), aprovechando el poder y versatilidad de los Algoritmos Genéticos (GA). En otros enfoques ya publicados (Ayman, Sharifkhani y Elmasry, 2004) se ataca el problema de optimización de la topología del circuito a sus parámetros, lo cual hace necesaria una búsqueda exhaustiva en el espacio de parámetros. Los GA, por el contrario, trabajan a un nivel de abstracción mayor en el que no se requiere información específica del circuito por optimizar. El GA solo recibe un conjunto de valores de aptitud (por ej., números reales) que representan las métricas de calidad del circuito como el retardo, el consumo de potencia, el área del silicio, etc. El usuario puede definir otras métricas o tipos de circuitos analógicos o digitales.

El optimizador propuesto utiliza el algoritmo genético conocido como PESA (*Pareto Envelope-based Selection Algorithm*) (Corney y Knowles, 2000), y depende de un simulador de circuitos estándar (por ej., Spectre, Spice, etc.) para tratar la complejidad de los modelos físicos de los transistores MOS y la topología del circuito. Los parámetros del circuito, como los voltajes de alimentación y polarización, ancho y largo de los transistores, etc., son generados por el algoritmo genético y enviados al simulador donde se calculan los parámetros de aptitud. Así, el diseñador tiene la posibilidad de cambiar el algoritmo de optimización, o bien, los modelos de simulación sin mucho esfuerzo.

Este documento comienza con un breve estudio de los circuitos MCML y los múltiples parámetros que afectan su comportamiento. Se incorpora también una introducción de los conceptos básicos relacionados con el Frente de Pareto, la arquitectura de nuestro optimizador y la forma en la que se empleó



La figura 2 muestra un ejemplo de un circuito MCML de dos niveles, es decir, sirve para implementar funciones lógicas de dos entradas. Se ha demostrado en otras investigaciones (Hassan, Anis y Elmasry, 2004; Hassan, Anis y Elmasry, 2005) que la optimización de compuertas de dos niveles es incluso más compleja y ésta crece conforme aumenta el número de niveles del circuito, en vista del incremento en el número de parámetros que pueden variarse. Esto implica que la optimización haciendo los cálculos a mano (de forma no automatizada) de los circuitos MCML es una tarea de poca utilidad práctica. Por ello, en esta investigación se vio la necesidad de contar con una estrategia de optimización que fuese tanto automatizada como independiente de la topología del circuito.

### Algoritmo genético para optimizaciones multiobjetivo de circuitos

Como se mencionó en la sección anterior, en el curso de esta investigación fue evidente la necesidad de desarrollar una herramienta denominada Automatización del Diseño Electrónico (EDA, por sus siglas en inglés) para optimizar circuitos MCML. Se decidió utilizar una estrategia con base en algoritmos genéticos, aprovechando la experiencia previa documentada en Alvarado (2004).

Los algoritmos genéticos se han utilizado previamente en la optimización de circuitos (MacEachern, 1999), aunque en esa investigación no se realizó una optimización multiobjetivo ya que solo se consideró una medida de aptitud (rendimiento). Esta medida se obtuvo por medio de una combinación lineal de otras tres medidas de rendimiento. El enfoque del presente proyecto ha sido utilizar múltiples valores de aptitud, que se calculan independientemente y luego se pasan, como entradas, al algoritmo genético para ser evaluados. La optimización multiobjetivo ha sido propuesta para el diseño de circuitos integrados analógicos, usando tanto métodos deterministas (Srinivastan, Samg Ha y Sulisty, 2004) como métodos heurísticos (algoritmo genético) (Smedt y Gielen, 2003). Aunque estos trabajos son similares al nuestro, éste ha sido propuesto en forma independiente y, a diferencia de los otros, fue concebido para optimizar circuitos digitales.

También se han publicado desarrollos de estrategias de optimización específicamente dirigidas a los

circuitos MCML (Müller-Gritschneider, 2005; Gielen, 2005), pero únicamente con base en métodos deterministas (seguidores de gradiente), que tienen la desventaja de ser proclives a quedar atrapados en mínimos locales.

### Evaluación de circuitos usando el Frente de Pareto

El Frente de Pareto es un concepto matemático de gran utilidad para evaluar la optimalidad de una serie de puntos en un espacio de aptitud multidimensional. Seguidamente se dará una explicación de los conceptos involucrados.

La función de aptitud agregada  $F$  para un circuito  $A$  con una parametrización  $u$ , se define como:

$$F(Au) = \Phi(f_1(Au), \dots, F_n(Au)) \quad (4)$$

Donde las funciones de aptitud individuales  $f_i(A_u)$  se definen como crecientes monótonamente con la aptitud de algún aspecto particular del comportamiento del circuito. Por ejemplo, en el caso del inversor MCML presentado anteriormente, los valores de aptitud están relacionados directamente con la excursión del voltaje de salida y son inversamente

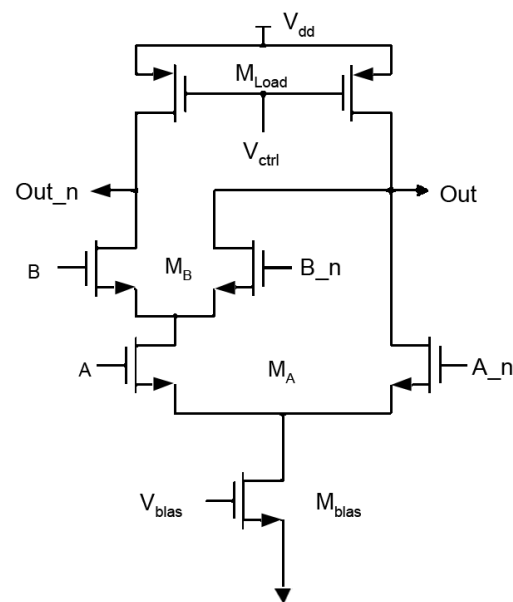


Figura 2. Ejemplo de una compuerta MCML de nivel 2 que implementa las funciones lógicas NAND/AND.

proporcionales a su consumo de potencia y a su retardo de propagación.

Las funciones  $f_i$  abarcan un espacio de aptitudes multidimensional, donde cada punto representa el rendimiento de un circuito parametrizado con un punto  $u$  en el espacio de parámetros. La forma general de  $\Phi$  se supone desconocida, pero debe crecer monótonamente si todas las funciones  $f_i$  crecen. Con esta condición se asegura que un punto en el espacio de aptitudes se pueda considerar más apto o mejor que todos los demás puntos con valores menores en todas las dimensiones. En la figura 3, por ejemplo, el punto  $q_1$  es más apto que el punto  $q_4$  y todos los otros elementos dentro del rectángulo gris. En este contexto, se dice que el punto  $q_1$  domina a  $q_4$ . Todos los puntos no dominados en un conjunto definen el Frente de Pareto de ese conjunto. En el ejemplo de la Fig. 3, el frente estaría definido por los puntos  $q_1$ ,  $q_2$  y  $q_3$ . Escoger una parametrización que no pertenezca al Frente de Pareto siempre es una mala selección pues habrá otro punto en el frente que tiene una mejor aptitud agregada.

Los conceptos anteriores se pueden expresar matemáticamente con la siguiente ecuación:

$$\hat{P} = \{ \langle u \in P_A, f(A_u) \rangle \mid \neg \exists v \in P_A : f(A_v) \prec f(A_u) \} \quad (5)$$

Donde  $P$  es el Frente de Pareto,  $f$  es el vector de funciones de aptitud  $[f_1, \dots, f_n]^T$  y  $P_A$  es el espacio de parámetros del circuito  $A$ . La relación de ordenamiento parcial " $\prec$ " en  $f$  describe la propiedad de dominación y se define como:

$$f(A_v) \succ f(A_u) \Leftrightarrow \forall i : f_i(A_v) \geq f_i(A_u) \wedge \exists i : f_i(A_v) > f_i(A_u) \quad (6)$$

El proceso de evaluación puede entonces considerarse como un proceso de mapeo que transforma el espacio de parámetros válidos  $P_A$  en una región conectada en el espacio de aptitudes  $[f_1, \dots, f_n]^T$ . El Frente de Pareto es el borde de esta región delimitada por los óptimos parciales (Müller-Gritschneider, 2005).

Cualquier algoritmo que encuentre el Frente de Pareto para un conjunto de puntos de aptitud implementa las ecuaciones (5) y (6). Puesto que el espacio de parámetros  $P_A$  usualmente contiene un número infinito de parametrizaciones, el siguiente problema consiste en escoger un conjunto

representativo de muestras de  $P_A$ , tal que su Frente de Pareto pueda suponerse como una aproximación confiable del frente exacto, extraído del espacio completo.

Una forma ingenua de enfocar el problema sería muestrear regularmente los valores de cada parámetro. En este caso, el número de evaluaciones crecería exponencialmente con el número de parámetros. Por ejemplo, un circuito con siete parámetros (variables de diseño), cada uno muestreado cinco veces, requeriría  $5^7 = 78\,125$  evaluaciones.

Para evitar hacer esta búsqueda exhaustiva de parámetros, se ha empleado el algoritmo evolutivo multiobjetivo llamado PESA (Corne y Knowles, 2000). Con este enfoque se suprimen los cálculos de parametrizaciones inútiles y el análisis se concentra en aquellas regiones que prometen mejores resultados. A pesar de que este algoritmo también discretiza el espacio de parámetros, la resolución usada en cada parámetro puede ser tan alta como sea necesario, sin el riesgo de un crecimiento exponencial del espacio de búsqueda. Entonces, el número de evaluaciones requeridas es proporcional al número de bits usados para representar una parametrización.

## Arquitectura de la herramienta CAD

El objetivo de la herramienta Diseño Asistido por Computador (CAD) que se ha desarrollado consiste en generar el Frente de Pareto de una celda dada, es decir, el conjunto de todas las parametrizaciones

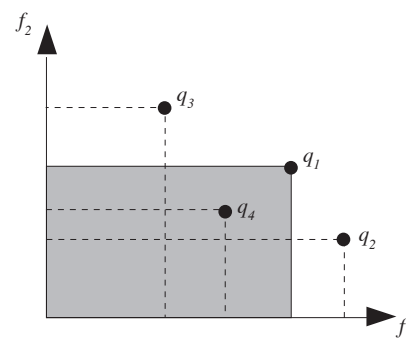


Figura 3. Frente de Pareto, el punto  $q_1$  domina la región oscurecida. Las líneas punteadas delimitan las regiones dominadas de los puntos  $q_2$ ,  $q_3$  y  $q_4$ . La línea gruesa representa el Frente de Pareto de los 4 puntos.

no dominadas, pues ellas representan a los mejores individuos que se buscan. Esta información se usa posteriormente para encontrar el punto de operación deseado del circuito, uno que satisfaga las especificaciones. En la figura 4 se muestra el diagrama de bloques correspondiente.

La funcionalidad de la herramienta está dividida en dos procesos bien definidos e independientes: una fase que sirve para capturar la representación del circuito y otra que se encarga de la optimización.

En la versión actual de la herramienta, una lista de conectividad (*netlist*) tipo Spice (simulador estándar de circuitos) captura el esquemático del circuito y se usa un simulador comercial (Spectre™) para calcular las medidas de rendimiento (o valores de aptitud) especificadas. Por otra parte, el núcleo del proceso de optimización está hecho con base en la LTI-LIB (LTI Image Processing Library, 2003), una biblioteca de código abierto originalmente concebida para realizar investigaciones de procesamiento de imágenes. Esta biblioteca provee la implementación del algoritmo PESA y genera el Frente de Pareto. En la figura 4, la línea gruesa de puntos ilustra la separación de las tareas, comunicándose entre ellas por medio de *sockets* TCP/IP.

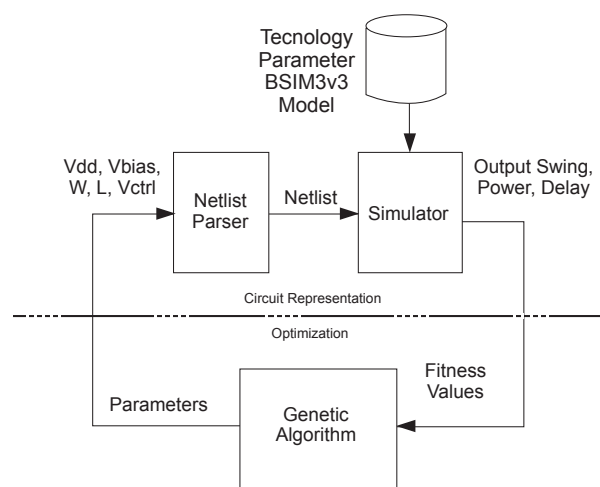


Figura 4. Arquitectura del Optimizador Genético de Circuitos. Desde el punto de vista de la optimización, los valores de aptitud y los parámetros son solo números, que representan variables físicas y geométricas del circuito.

## Resultados

Además del inversor y la compuerta NAND que se han mostrado en las figuras 1 y 2, respectivamente, también se ha implementado una biblioteca de componentes lógicos fundamentales. Esta biblioteca contiene los siguientes componentes: compuerta XOR de tres entradas, un *latch* D, un *flip-flop* D y un circuito generador de acarreo de salida. Estos son los bloques básicos para implementar circuitos aritméticos.

En la figura 5 se muestra un Frente de Pareto en tres dimensiones, correspondiente a la compuerta XOR implementada en este proyecto. Este frente en particular lo conforman 2500 individuos (parametrizaciones) y fue generado por el algoritmo genético PESA. Esta figura ilustra el compromiso que se tiene al querer obtener circuitos de alta velocidad: al reducirse el retardo de propagación se incrementará el consumo de potencia y se reducirá la excursión del voltaje de salida.

Una pregunta clave a la hora de diseñar circuitos MCML es: ¿Qué valores deberían tener las variables de un circuito dado para lograr un cierto nivel de rendimiento? Como se mencionó en la sección III, la cantidad de variables, el efecto que tienen sobre las métricas de rendimiento y la interdependencia

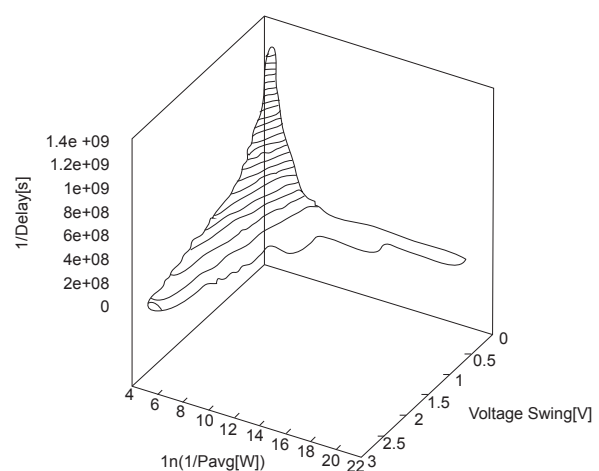


Figura 5. Frente de Pareto de una XOR MCML. Para este ejemplo se definieron tres métricas de diseño: el rango de voltaje, el consumo de potencia promedio y el retardo de propagación, aunque es posible definir otras métricas.

entre estas últimas, hacen que la respuesta al problema no sea obvia o se pueda obtener directamente de una ecuación. Por esta razón, la utilización de herramientas para automatización del diseño electrónico (EDA) puede ser de gran ayuda para encontrar una respuesta.

En el cuadro 1 se presenta un ejemplo de las parametrizaciones obtenidas para cada componente de la biblioteca que se ha implementado. En este caso concreto, la meta era encontrar circuitos con un retardo cercano a 1 ns, al mismo tiempo que la excursión de voltaje no fuese menor que 0,5V. En el cuadro 2 se muestran los resultados después de haber hecho simulaciones post *layout*, usando la tecnología para fabricación de 0,35  $\mu\text{m}$  de Austria Microsystems.

Es necesario señalar que del Frente de Pareto no se obtiene solo un elemento ganador; eso depende de lo que el diseñador de circuitos requiera lograr. De hecho, el frente mostrado en la figura 5 está compuesto por 2500 elementos ganadores, es decir, el conjunto de los individuos no dominados.

### Conclusiones

En este trabajo se ha presentado la aplicación de una estrategia automatizada para diseñar circuitos de lógica en modo de corriente (MCML). Específicamente, se ha creado una biblioteca de compuertas MCML optimizadas, necesarias para la implementación de circuitos aritméticos.

Cuadro 1. Parámetros extraídos del Frente de Pareto de cada compuerta.

Parameter\Gate	XOR	Carry_Out	NAND	Dflip-Flop
$V_{dd}$ [V]	2,7	2,7	2,7	2,7
$V_{bias}$ [V]	2,7	2,7	2,7	2,7
$V_{cntrl}$ [V]	0	0	0	0
$W_{bias}$ [ $\mu\text{m}$ ]	2,9	4,8	4,1	5,83
$L_{bias}$ [ $\mu\text{m}$ ]	0,65	1,5	1,22	0,43
$W_a$ [ $\mu\text{m}$ ]	4,8	2,58	3,45	3,95
$L_a$ [ $\mu\text{m}$ ]	2,48	0,75	1,22	0,43
$W_b$ [ $\mu\text{m}$ ]	2,9	5,15	3,1	5,65
$L_b$ [ $\mu\text{m}$ ]	0,75	0,65	0,9	1,53
$W_c$ [ $\mu\text{m}$ ]	5,14	4,2	-	-
$L_c$ [ $\mu\text{m}$ ]	1,30	0,51	-	-
$W_{load}$ [ $\mu\text{m}$ ]	6	4,8	5,83	5,83
$L_{load}$ [ $\mu\text{m}$ ]	2,72	0,83	0,75	0,83

Cuadro 2. Mediciones de desempeño (calidad o aptitud) de las compuertas.

Gate	Power[mW]	Out Swing[V]	Delay[ns]
Xor	0,4	1,45	1,3
Carry_Out	0,7	0,5	0,96
Nand	0,78	0,7	1,7
D Flip-Flop	0,97	1,46	0,25



Al tenerse varias métricas de rendimiento que compiten entre sí, el problema de la optimización ha sido tratado como una optimización multiobjetivo. Por ello, se ha utilizado el concepto de Frente de Pareto como un instrumento para hallar los puntos optimizados o, dicho de otra manera, el o los puntos no dominados que cumplen con una especificación de diseño.

## Bibliografía

- Alvarado, J. P. (julio 2004). *Segmentation of color images for interactive 3D object retrieval*. (Tesis doctoral). Germany: Rheinisch-Westfälischen Technischen Hochschule Aachen.
- Ayman, H. I., Sharifkhani M. & Elmasry M. I. (Mayo 2004). *On the Design of Low Power MCML Based Ring Oscillators*. Ingeniería en Electrónica y Computación 2004. Conferencia Canadiense, 4, 2383-2386.
- Corne, D. & Knowles, J. (2000). *The Pareto envelope-based selection algorithm for multiobjective optimization*. En *PPSN VI: Proceedings of the International Conference on Parallel Problem Solving from Nature* (pp. 839-848).
- Gielen, G. (2005). Performance space modeling for hierarchical synthesis of analog integrated circuits. En *ACM/IEEE DAC 2005*.
- Hassan, H., Anis, M. & Elmasry, M. (septiembre 2004). *MOS current mode circuits: analysis, design, and variability*. Proceedings of IEEE International SOC Conference. pp. 247-250.
- Hassan, H., Anis, M. & Elmasry, M. (Agosto 2005). MOS current mode circuits: analysis, design, and variability. *IEEE Trans. VLSI Syst.*, 13(8), 885-898.
- LTI Image Processing Library: Developer's Guide. Version 29.10.2003. Obtenido de <http://www.techinfo.rwth-aachen.de> de <http://ltilib.sourceforge.net>
- MacEachern, L. A. (1999). Constrained circuit optimization via library table genetic algorithms. En *Proc. IEEE Int. Symp. Circuits and Systems ISCAS '99*, 6, 310-313.
- Mizuno, M. (junio 1996). A GHz MOS, Adaptive Pipeline Technique Using MOS Current-Mode Logic. *IEEE Journal of Solid-State Circuits*, 31(6), 784-791.
- Müller-Gritschneider, D. (2005). Deterministic approaches to analog performance space exploration (PSE). En *ACM/IEEE DAC 2005*.
- Musicer, J. M. (2004). *An Analysis of MOS Current Mode Logic for Low Power and High Performance Digital Logic*. (Tesis de Maestría). Department of Electrical Engineering and Computer Sciences, University of California at Berkeley.
- Musicer, J. M. & Rabaey, J. (julio 2000). MOS Current Mode Logic for Low Power, Low Noise, CORDIC Computation in Mixed-Signal Environments. En *Proceedings of the 2000 International Symposium on Low Power Electronics and Design, ISLPED '00*, 102-107.
- Smedt, B. D. & Gielen, G. E. (2003). *HOLMES: Capturing the Yield-Optimized Design Space Boundaries of Analog and RF Integrated Circuits*. En *Proceedings of IEEE DATE 2003*.
- Smedt, B. D. & Gielen, G. E. (2003). *WATSON: Design space boundary exploration and model generation for analog and RF IC design*. IEEE Trans.
- Srinivastan, V., Samg Ha, D. & Sulistyo, J. B. (2004). *Gigahertz-range MCML Multiplier Architectures*. ISCAS 2004.