



UNIVERSIDAD DISTRITAL  
FRANCISCO JOSÉ DE CALDAS

# PROCESADOR DE APLICACIÓN ESPECÍFICA PARA SIMULAR UNA COLONIA DE HORMIGAS SOBRE FPGA

Cristian D. Rodriguez  
cridrodriguezr@correo.udistrital.edu.co

Miguel A. Melgarejo  
mmelgarejo@ieee.org

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS



LAMIC  
Laboratorio de Automática, Microelectrónica  
e Inteligencia Computacional

## Introducción

Según [1], Los sistemas bio-inspirados son sistemas construidos por medio de hardware configurable que emulan el modo de procesar información y resolución de problemas de un sistema biológico. El proyecto propone modelar una colonia de hormigas (Inspirada en modelo de Netlogo) [2]. Las colonias de hormigas son sistemas sociales complejos, su comportamiento emerge de la cooperación entre la especie, una sola hormiga no es inteligente [3]. El proyecto aborda una propuesta arquitectural hardware y su correspondiente implementación para un microprocesador que emula el comportamiento emergente de una colonia de hormigas. En el centro de la propuesta se encuentra que el comportamiento de una sola hormiga puede ser capturado por medio de una máquina de estados algorítmica. La colonia de hormigas se concibe entonces como un conjunto de máquinas de estado que interactúan entre sí por medio de una memoria distribuida asociada al espacio de recolección de comida de las hormigas. En este

sentido se estaría hablando de una arquitectura de procesamiento con capacidades de cómputo emergente.

Para implementar la colonia se utiliza la tecnología FPGA (Field Programmable Gate Array). Esta provee beneficios importantes tales como no necesitar un sistema operativo, una considerable mejora en el rendimiento aprovechándose del paralelismo del hardware, tiempos de respuesta más veloces y la facilidad del mantenimiento o actualización de la implementación [4].

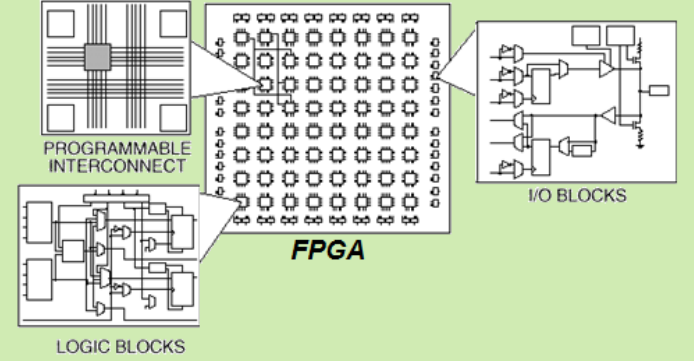
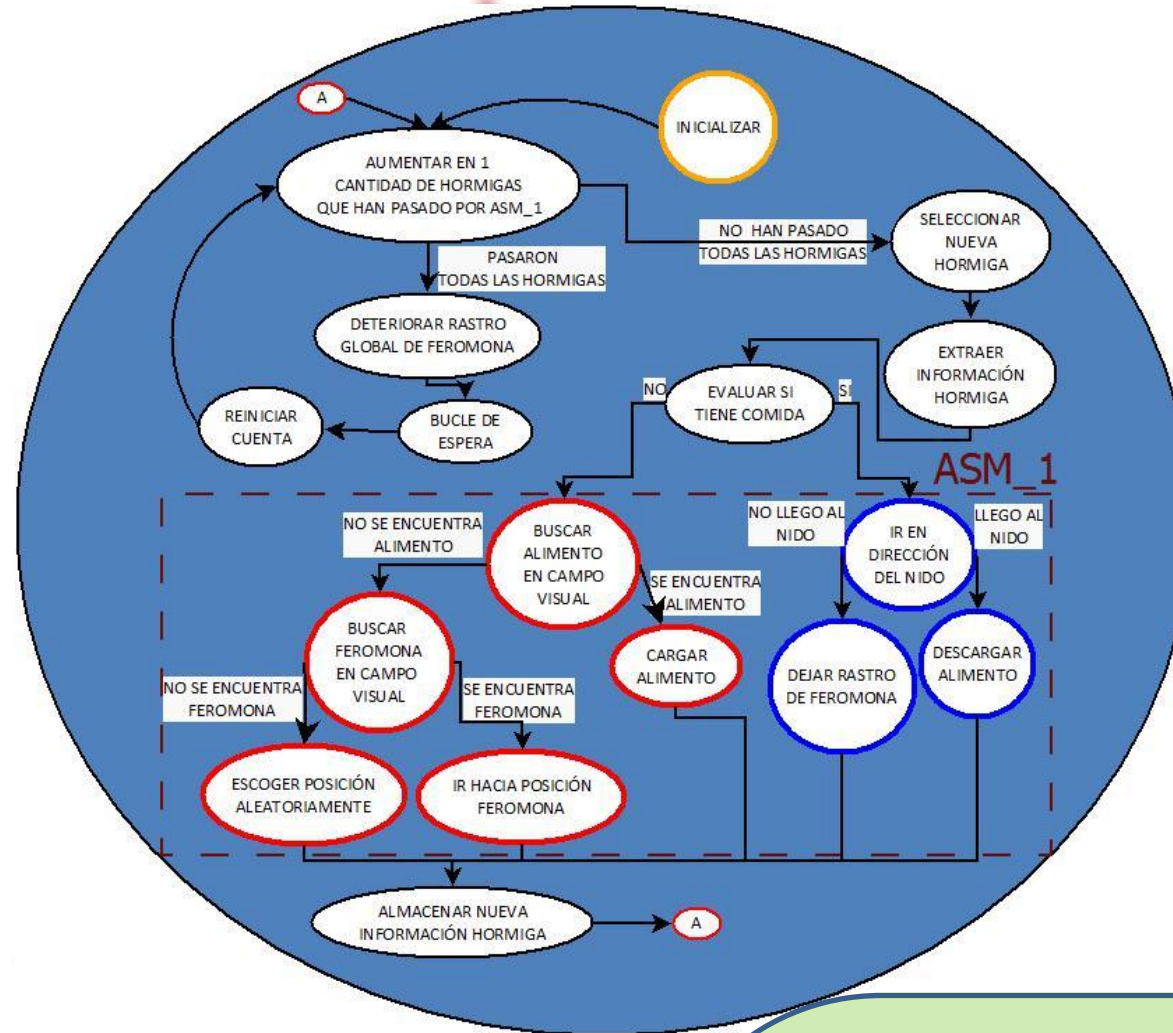


Figura 1. Componentes de una FPGA [5].

## Planteamiento del problema

Las hormigas buscan alimento estocásticamente. Seleccionan su trayectoria en base a la información de feromona contenida en el espacio de búsqueda. Como resultado de este proceso, indirectamente cada hormiga puede comunicarse con las demás. Esta comunicación les permite ser atraídas hacia las mejores soluciones. La búsqueda eficiente se logra gracias a la intensificación del camino más corto por la realimentación positiva de la información de feromona por selección estocástica. Esta realimentación positiva se produce porque el camino solución se refuerza de feromona más rápido que los otros al requerir un menor tiempo para ser completado [6]. Se debe diseñar un procesador que permita simular este proceso.

Figura 2. Diagrama funcional para las hormigas obreras.



## Metaheurística ACO

La Metaheurística ACO está inspirada en la observación del comportamiento de hormigas reales considerándose un algoritmo de inteligencia colectiva [7]. En tabla 1, se describen las ecuaciones utilizadas para modelar el proceso de recolección de alimento y sus correspondientes condiciones.

Tabla I. Metaheurística ACO para implementación sobre FPGA		
ITEM	ACO	CONDICIONES
Probabilidad para escoger nueva posición	$P^k(i,j) = \frac{[T(i,j)]^\alpha * [\eta(i,j)]^\beta}{\sum_{l \in N^k} [T(i,l)]^\alpha * [\eta(i,l)]^\beta}$ Para $n(i,j) = \frac{1}{d(i,j)}$	$T \in N, (1 \leq T \leq 8)$ $d \in N, (2 \leq d \leq 3)$ $i \in N, (i \leq 255)$
Actualización de feromona	$T_f(i,j) = (1-p) * T_f(i,j) + \sum_{k=1}^m \Delta T^k(i,j)$	$j \in N, (j \leq 255)$ $k \in N, (1 \leq k \leq 32)$ $\alpha = \beta = 1, p = 0.125$

## Propuesta a desarrollar

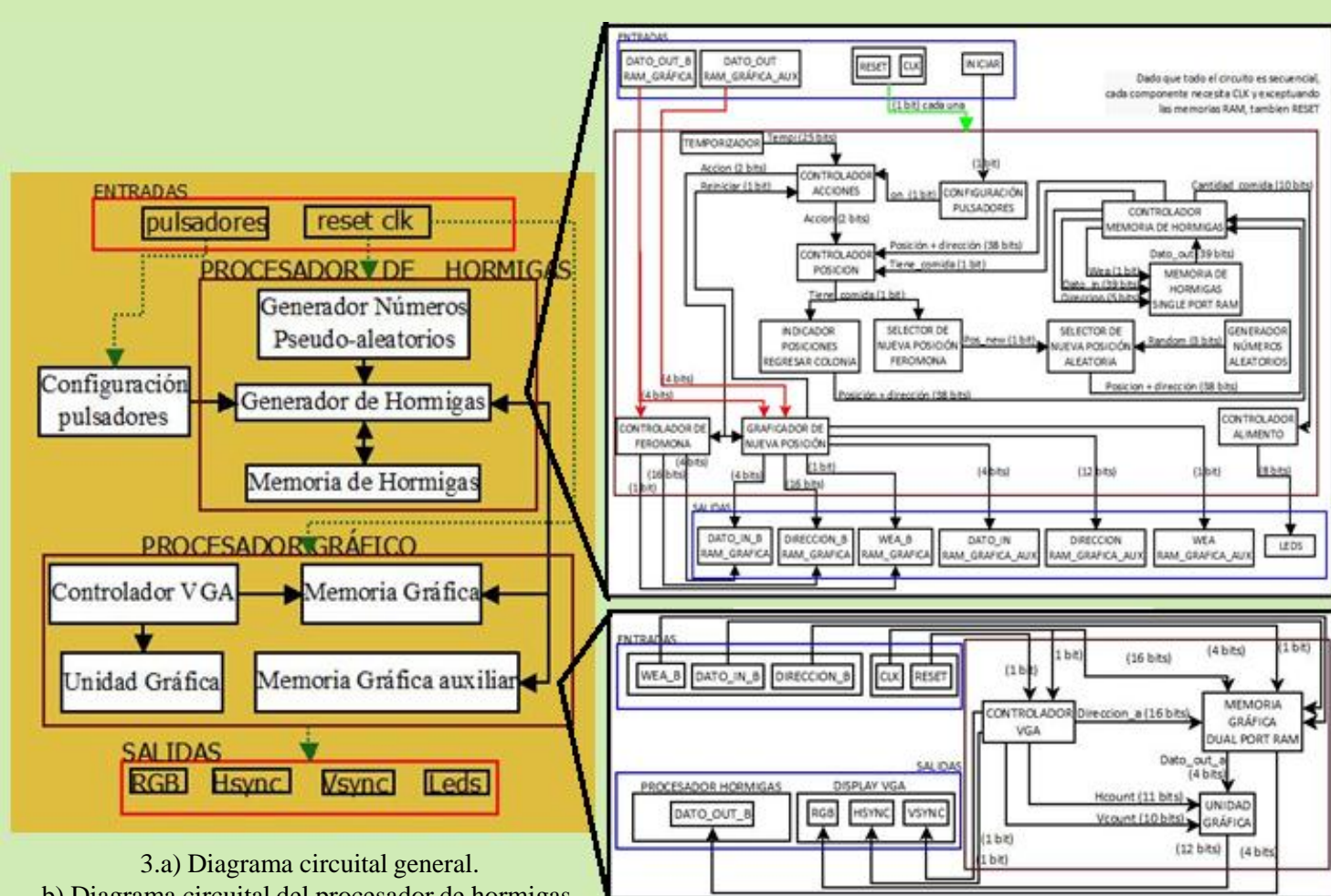
Se diseña un ASM [8] en dos etapas. La primera, se encarga de modelar a una hormiga obrera y la segunda, nos permite usar la ASM anterior para cada una de las demás, es decir, la primera ASM tipifica el cerebro de la hormiga y la segunda ASM reutiliza este cerebro para cada hormiga, almacenando la información necesaria de cada una, como posición, dirección y objetivo actual.

## Desarrollo en Hardware

**Procesador gráfico.** Se diseña un GPU para controlar una pantalla VGA. El proceso de graficación se realiza por medio de visualización dinámica. La GPU se forma por tres componentes básicos que son el controlador VGA, la unidad gráfica y la memoria RAM de doble puerto[9].

**Procesador de hormigas.** Este procesador tiene como propósito escribir la memoria RAM gráfica utilizando dos memorias RAM auxiliares y valiéndose de un generador de números pseudo-aleatorios para tomar decisiones. La memoria gráfica auxiliar hace pequeños backups de la memoria gráfica, según se requiera. Es una memoria de 128 posiciones con una longitud por posición de 4 bits, RAM 128X4. La memoria de hormigas permite guardar información característica de cada hormiga en 32 posiciones de una longitud por posición de 39 bits, RAM 32X39.

Este procesador se dirige por un controlador de acciones, el cual decide en que momento opera el controlador de hormigas, selector de posición, generador de gráfica de la hormiga, controlador de alimento o el controlador de feromona. Todo el proceso es sincrónico.



3.a) Diagrama circuital general.  
b) Diagrama circuital del procesador de hormigas.  
c) Diagrama circuital del procesadores gráfico.

Como se evidencia en la tabla 1, el uso de los recursos de la FPGA es mínimo, exceptuando la RAM debido al procesador gráfico. A pesar de la complejidad del circuito diseñado, se logra implementar usando unidades básicas como Slices, Flip Flops, Luts, IOBs y RAM. La interfaz gráfica es de resolución 256X256.

Como se evidencia en la figura 4, la primera fuente de alimento en ser hallada se desabastece exponencialmente. Al acabarse la comida e introducirse una nueva fuente de esta, se repite el proceso de desabastecimiento exponencial. La selección heurística de las nuevas posiciones permite a las hormigas localizar rápidamente las fuentes de alimento encontradas por compañeras. En la implementación es posible variar el número de hormigas en la colonia desde 1 hasta 32. La cantidad de alimento recolectado en un tiempo determinado crece exponencialmente conforme a la cantidad de hormigas que estén involucradas en el proceso. Estos fenómenos son totalmente emergentes, y son comprobables experimentalmente con colonias reales.



## Conclusiones

El procesador realiza la simulación de manera eficiente en uso de hardware, replicando correctamente las características principales de la colonia. Se comprueba el uso de la feromona para encontrar el camino más corto hacia los alimentos y la rapidez con la que se abastece la colonia luego de que una hormiga encuentra alimento por primera vez. Se evidencia además una propiedad emergente en el entorno, el espiral de la muerte, fenómeno en el cual las hormigas se quedan encerradas en un círculo direccionadas por su propia feromona.

Se demuestra entonces que un sistema estructuralmente sencillo es capaz de simular la complejidad del proceso de recolección de alimento de una colonia de hormigas y que el hardware configurable es una tecnología indicada para implementar un procesador de propósito especial.

## Referencias

- [1] Xin-She Yang, Mehmet Karamanoglu, Swarm Intelligence and Bio-Inspired Computation: An Overview. Elsevier, 1º Ed. Mayo 2013, 3 - 23.
- [2] Wilensky, U. (1997). NetLogo Ants model. <http://ccl.northwestern.edu/netlogo/models/Ants>. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.
- [3] Patricia J. Folgarait; Alejandro G. Farji-brener, Un mundo de hormigas. 1ª edición, Buenos Aires, Argentina, 2005, 15-18.
- [4] Clive Max Maxfield, FPGAs: Instant Access. Elsevier, 2008, 1-12.
- [5] Imagen disponible en <http://www.ni.com/tutorial/6097/en/>.
- [6] P.-P. Grassé, Recherches sur la biologie des termites champignonnistes (Macrotermitinae). Annales des Sciences Naturelles, Zoologie, 1944, 11(6):97-171.
- [7] M. Dorigo. Optimization, Learning and Natural Algorithms. PhD thesis, Dipartimento di Elettronica, Politecnico di Milano, Milan, 1992.
- [8] D. Gajski, Principios de diseño digital, 7ª edición, Prentice Hall, Madrid, España, 1999, Cap 8, sec 8.1-8.4.
- [9] Breitbart, J., Data structure design for GPU based heterogeneous systems. IEEE, High Performance Computing & Simulation. HPCS '09, Junio, 2009, 44-51.
- [10] Imagen de fondo disponible en <http://www.yalotenemos.com/media/catalog/product/cache/1/base/9/d/7/8/9/d78eab33525d08d6e5fb8d27136e95/k/i/kit-hormigas-1-100.png>