DIVISOR PARA CAMPOS DE GALOIS EN UN PLD

Mario Alberto García Martínez
Instituto Tecnológico de Orizaba
Depto. de Ingeniería Eléctrica-Electrónica
Oriente 9 No. 852, Col. Emiliano Zapata. C.P. 94300
Tel. 012-7244016 ext. 24, Fax 012-7257056
"e-mail": marioag@prodigy.net.mx

Guillermo Morales-Luna
Sección de Computación
Departamento de Ingeniería Eléctrica
CINVESTAV-IPN
Av. IPN 2508, 07360 México, D.F.
"e-mail": gmorales@cs.cinvestav.mx

José Antonio Moreno Cadenas
Sección de Electrónica del Estado Sólido
Departamento de Ingeniería Eléctrica
CINVESTAV-IPN
Av. IPN 2508, 07360 México, D.F.
"e-mail": jmoreno@mail.cinvestav.mx

Resumen: Presentamos una implementación en un PLD de un divisor, que actúa en forma serial, sobre campos finitos de la forma GF(2^m). Utilizamos un algoritmo que calcula la división resolviendo un sistema de ecuaciones lineales. El divisor requiere sólo de tres procesadores básicos y de una señal de control. Las complejidades del circuito, en cuanto al número de componentes y al tiempo, son proporcionales a m^2 y a m respectivamente. La estructura es independiente del polinomio irreducible y se puede extender a diversos valores de m. Las áreas de aplicación de estos circuitos incluyen los códigos de corrección de errores y la criptografía.

INTRODUCCIÓN

Con el propósito de asegurar la información que se transmite a través de canales públicos de comunicación, los sistemas digitales de transmisión utilizan diferentes técnicas de criptografía. Muchas de ellas requieren de operaciones realizadas en campos finitos (o de Galois) GF(p^n). Otra aplicación importante de los campos de Galois se encuentra en los códigos de corrección de errores y en el procesamiento digital de señales.

Por lo general, la operación de la división en campos finitos se realiza calculando primero la inversa multiplicativa del divisor y después multiplicándola por el dividendo. Esto incrementa considerablemente el tiempo de ejecución y la complejidad de los divisores.

Diferentes métodos han sido propuestos para el cálculo de inversos multiplicativos basados ya sea en el algoritmo de Euclides [1], o en el teorema "pequeño" de Fermat [2]. Todos éstos utilizan operaciones recursivas para calcular cuadrados y multiplicaciones polinomiales sobre bases determinadas de GF(p^n).

Frente a estas propuestas, recientemente se han desarrollado mejores métodos para el cálculo de la división, basados en la solución de ecuaciones lineales sobre GF(2) [3][4]. El algoritmo de división (**AD**) que utilizaremos fue desarrollado en [5] y requiere de la solución de m ecuaciones lineales sobre GF(2^m), (m es la dimensión del campo, visto como un espacio vectorial sobre su subcampo primo, y 2^m es su cardinalidad, es decir, su número de elementos, llamado también, su *orden*). El algoritmo procede en dos etapas. En la primera se construye una matriz de coeficientes del divisor. En la segunda se resuelve el sistema de ecuaciones lineales previamente definido. El divisor es modular y muy apropiado para el manejo de valores grandes de m. Además, no depende del polinomio irreducible que se utilice para generar el campo. La duración del ciclo de tiempo, o de reloj, es independiente de m y el tiempo de procesamiento es proporcional a m.

Este documento está organizado como sigue: En la sección I presentamos los pormenores del algoritmo de división AD. En la sección II se muestra la estructura del divisor serial y los módulos que lo componen. En la sección III presentamos las simulaciones y las pruebas al circuito desarrollado y en la sección IV las conclusiones.

I. ALGORITMO DE DIVISIÓN

El algoritmo utilizado para el desarrollo del divisor es el que se encuentra en [5].

Sea *g(x)* un polinomio mónico irreducible de grado *m* sobre GF(2). Escribamos

$$g(x) = \sum_{i=0}^{m-1} g_i x^i + x^m$$

donde g_i Î GF(2), para i=0,1,...,m-1, y m es un entero positivo diferente de cero. GF(2^m) es una extensión del campo primo GF(2) y tiene 2^m elementos. Sea a Î GF(2^m) una raíz de *g(x)*. Entonces todos los elementos de GF(2^m) pueden ser representados en términos de la base {1, α, α^2,..., α^{m-1}}. Específicamente si a Î GF(2^m) es cualquier elemento, entonces existe una sucesión de escalares a_i Î GF(2), con $0 \le i \le m-1$, tal que:

$$a = a_0 + a_1 a + a_2 a^2 + \ldots + a_{m-1} a^{m-1}$$

La base {1, α, α^2, . . ., α^{m-1}} se dice ser la *base canónica* y los términos a_i las *coordenadas de a* con respecto a la base. Las 2m-1 potencias { α^0, α^1, α^2, ... , α^{2m-2} } son conocidas como los *elementos de soporte*. Denotemos por $\boldsymbol{A}^{(k)} = \left[a^i\right]_{i=0}^{k-1}$ al vector columna de dimensión k que consta de las primeras potencias a^i, i=0, …, k-1. Al expresar a las potencias hasta k=2m-2, en términos de la base canónica, podemos escribir

$$\boldsymbol{A}^{(2m-1)} = \boldsymbol{P}^T \boldsymbol{A}^{(m)} \qquad (1)$$

donde $\boldsymbol{P} = \left(p_{ij}\right)_{i=0,\ldots,m-1}^{j=0,\ldots,2m-2}$ es una matriz de orden $m \times (2m-1)$ con entradas en el campo primo GF(2). De hecho, p_{ij} es la *i-ésima* coordenada del elemento de soporte a^j. Y como la base canónica es, en efecto, una base, tenemos que

$$\boldsymbol{P} = \left[\boldsymbol{1}_{(m)} \quad \boldsymbol{P}_1\right] \qquad (2)$$

donde $\boldsymbol{1}_{(m)}$ es la matriz identidad de orden $m \times m$ y $\boldsymbol{P}_1$ es una matriz de orden $m \times (m-1)$ a calcularse mediante el polinomio irreducible *g*(*x*).

Sean a y c dos elementos en GF(2^m), a ¹ *0*. Sea

$$b = c/a.$$

Entonces, c=a b. En términos de la base canónica, escribamos

$$a = \boldsymbol{a}^T \boldsymbol{A}^{(m)} \quad , \quad b = \boldsymbol{b}^T \boldsymbol{A}^{(m)} \quad , \quad c = \boldsymbol{c}^T \boldsymbol{A}^{(m)}$$

donde $\boldsymbol{a}$, $\boldsymbol{b}$ y $\boldsymbol{c}$ son los vectores columnas consistentes de las coordenadas de a, b y c respecto a la base canónica. Un cálculo directo muestra que en GF(2^m) se han de cumplir las igualdades

$$\boldsymbol{c}^T \boldsymbol{A}^{(m)} = (\boldsymbol{a} * \boldsymbol{b})^T \boldsymbol{A}^{(2m-1)} = (\boldsymbol{a} * \boldsymbol{b})^T \boldsymbol{P}^T \boldsymbol{A}^{(m)}$$

donde $\boldsymbol{a} * \boldsymbol{b}$ denota a la convolución del vector $\boldsymbol{a}$ por el vector $\boldsymbol{b}$. Por consiguiente, hemos de tener

$$\boldsymbol{c} = \boldsymbol{P}(\boldsymbol{a} * \boldsymbol{b})$$

Explícitamente, para cada $i = 0, \ldots, m-1$, vale

$$c_i = \sum_{k=0}^{2m-2} p_{ik} \sum_{l+j=k} a_l b_j$$

Intercambiando el orden en el que se realizan las sumatorias y haciendo algunos renombramientos de sus índices obtenemos las expresiones equivalentes:

$$c_i = \sum_{j=0}^{m-1} \left(\sum_{k=0}^{m-1} p_{i,k+j} a_k \right) b_j$$

en otras palabras, resulta el sistema de ecuaciones

$$\boldsymbol{c} = \boldsymbol{Q}\boldsymbol{b} \qquad (3)$$

donde la matriz $\boldsymbol{Q} = \left(q_{ij}\right)_{i=0,\ldots,m-1}^{j=0,\ldots,m-1}$ tiene como entradas a los elementos

$$q_{ij} = \sum_{k=0}^{m-1} p_{i,k+j} a_k \qquad (4)$$

La ecuación (3) representa un sistema de m ecuaciones lineales cuyas incógnitas son las entradas en $\boldsymbol{b}$. Cuando las coordenadas de a, c y los elementos de soporte son conocidos, el cociente $b = c/a$ se obtiene pues como la solución de ese sistema en el campo GF(2^m) [5].

II. EL DIVISOR SERIAL

Formación de la matriz $\boldsymbol{Q}$

Recordemos la expresión (2) de la matriz $\boldsymbol{P}$. Tenemos, mediante el polinomio $g(x)$ y la expresión $a^m = \sum_{i=0}^{m-1} p_{im} a^i$, que para $i \le m-1$, $p_{im} = g_i$. Sucesivamente tenemos

$$\begin{aligned} \boldsymbol{a}^{k+1} &= \boldsymbol{a} \cdot \boldsymbol{a}^k = \boldsymbol{a} \sum_{i=0}^{m-1} p_{ik} \boldsymbol{a}^i \\ &= \sum_{i=1}^{m-1} p_{i-1,k} \boldsymbol{a}^i + p_{m-1,k} \boldsymbol{a}^m \\ &= \sum_{i=1}^{m-1} p_{i-1,k} \boldsymbol{a}^i + p_{m-1,k} \sum_{i=0}^{m-1} p_{im} \boldsymbol{a}^i \\ &= (p_{m-1,k}\, p_{0m}) + \sum_{i=1}^{m-1} \left(p_{i-1,k} + p_{m-1,k}\, p_{im}\right) \boldsymbol{a}^i \end{aligned}$$

Así pues resultan las recurrencias

$p_{0,k+1} = p_{m-1,k}\, p_{0m}$ y $p_{i,k+1} = p_{i-1,k} + p_{m-1,k}\, p_{im}$

las cuales pueden ser escritas, de manera sintética como sigue: Para cada $i = 0,1,\ldots, m-1$

$$p_{i,k+1} = (1-d_{0i})\, p_{i-1,k} + p_{m-1,k}\, g_i$$

De la ecuación (4) tenemos que los elementos de la columna 0 de la matriz ***Q*** han de ser:

$$q_{i0} = \sum_{k=0}^{m-1} p_{ik} a_k = a_i$$

Y para $1 \le j \le m-1$ tenemos:

$$\begin{aligned} q_{ij} &= \sum_{k=0}^{m-1} p_{i,k+j} a_k \\ &= \sum_{k=0}^{m-1} \left((1-d_{0i})\, p_{i-1,k+j-1} + p_{m-1,k+j-1} g_i\right) a_k \\ &= (1-d_{0i}) \sum_{k=0}^{m-1} p_{i-1,k+j-1} a_k + g_i \sum_{k=0}^{m-1} p_{m-1,k+j-1} a_k \\ &= (1-d_{0i})\, q_{i-1,j-1} + g_i a_{m-j} \\ &= \begin{cases} g_0 q_{m-j,0} & \text{si} \quad i = 0 \\ g_i q_{m-j,0} + q_{i-1,j-1} & \text{si} \quad i = 1,\ldots, m-1 \end{cases} \end{aligned}$$

Estructura del arreglo que genera la matriz Q

El arreglo para generar la matriz ***Q*** (*Gen-Mat)* se muestra en la figura 1.

El arreglo consiste de *m*-1 celdas rectangulares etiquetados como Cel_1, Cel_2, Cel_3,... Cel_{m-1}. Las coordenadas de *a* se introducen a Cel_1 en forma serial siendo a_{m-1} la primera en presentarse.

La columna 0 se obtiene directamente de las coordenadas de *a* que entran a Cel_1. Las columnas 1 a *m*-1 son generadas por Cel_1, Cel_2 ... Cel_{m-1} respectivamente. La salida de Cel_{j-1} se introduce en la celda correspondiente Cel_j.

En esta figura las señales indicadas se definen como sigue:

g_{in} = *Polinomio irreducible*
q_{in} = *Señal de control*
a_{in} = *Coordenadas del divisor*

En la figura 2 se muestra la estructura interna de cada celda Cel_j.

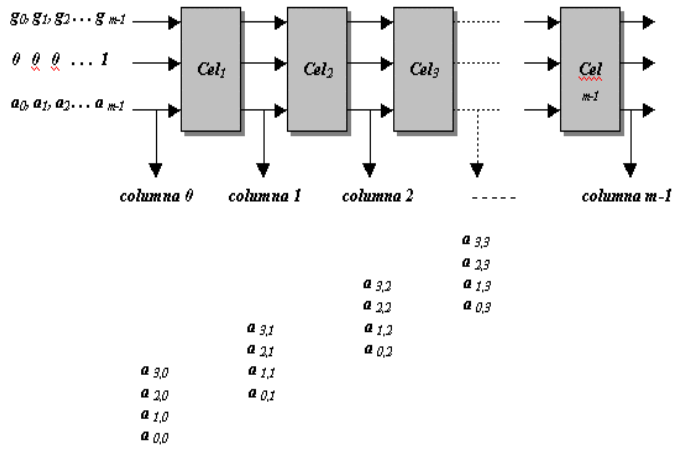


Fig. 1. Arreglo generador de la matriz (Gen-Mat)

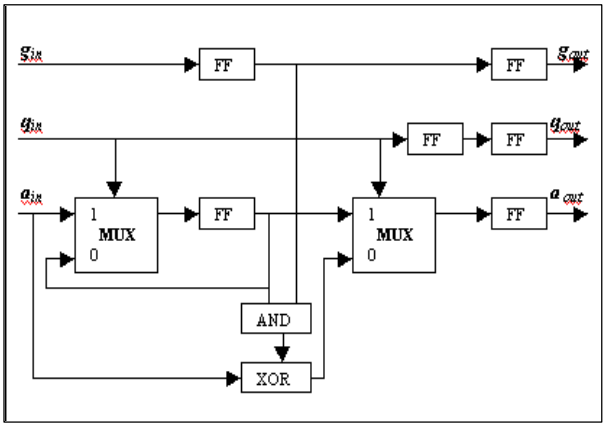


Fig. 2. Estructura interna de la celda Cel_j

Solución del sistema de ecuaciones

Observamos que el sistema de ecuaciones (3) posee una única solución ***b***, pues GF(2^m) es en efecto un campo. Por tanto el determinante de la matriz ***Q*** es distinto de cero. Siendo un elemento de GF(2) necesariamente su valor ha de ser 1.

Lema*: Si a es un elemento diferente de cero en el campo GF(2^m), entonces el determinante de la matriz formada por Gen-Mat es* 1.

Consideremos la matriz :

$$\boldsymbol{P}_{i,j} = \begin{pmatrix} 1 & & & & & & & & \\ & \ddots & & & & & & 0 & \\ & & 1 & & & & & & \\ & & & f^{-} & & f & & & \\ & & & & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & 1 & & \\ & & & h & & & f^{-} & & \\ & & & & & & & 1 & \\ & 0 & & & & & & & \ddots \\ & & & & & & & & 1 \end{pmatrix}$$

Los elementos de $\boldsymbol{P_{i,j}}$ en *(i,j), (i,i), (j,i) y (j,j)* son h, f^{-}, f y f^{-}, respectivamente, donde:

$$h = q_{i,j} \qquad \text{y} \qquad f = q^{-}_{j,j}\, q_{i,j}$$

y f^{-} representa al complemento de f. Los elementos restantes de $\boldsymbol{P_{i,j}}$ son cero excepto los demás en la diagonal principal que son 1. De la definición de $\boldsymbol{P}_{i,j}$ se obtiene:

$$\begin{aligned} \det(\boldsymbol{P}_{i,j}) &= f^{-}f^{-} + hf \\ &= 1 + f + hf \\ &= 1 + h^{-}f \\ &= 1 + a^{-}_{i,j}\, a^{-}_{j,j}\, a_{i,j} \\ &= 1 \end{aligned} \qquad (5)$$

Nos referiremos como la *matriz operando* a aquella premultiplicada por $\boldsymbol{P}_{i,j}$. La premultiplicación sólo afecta los renglones i y j en la matriz operando. Hay tres posibles cambios y éstos quedan enlistados en la tabla I.

h	f	***Cambios***
0	0	Sin cambio
1	0	i := renglón i + renglón j
1	1	reng. i y reng. j se intercambian

Tabla I: Cambios en la matriz operando después de la premultiplicación.

Eliminación de Gauss-Jordan para GF(2)

Hagamos

$$\boldsymbol{U_i = P}_{m\text{-}1,i}\, \boldsymbol{P}_{m\text{-}2,i} \ldots \boldsymbol{P}_{i+1,i} \;\; ; \;\; \boldsymbol{Q^{(0)} = U_0 Q.}$$

Entonces de acuerdo a lo visto anteriormente, todos los elementos debajo de $q^{(0)}_{0,0}$ de la columna 0 de $\boldsymbol{Q}^{(0)}$ son cero. Realizando el proceso de eliminación de las columnas 0 a la m-2, se tiene:

$$\boldsymbol{[T,c']} = \boldsymbol{U}_{m\text{-}2}\, \boldsymbol{U}_{m\text{-}3} \ldots \boldsymbol{U}_0 \boldsymbol{[Q,c]} \qquad (6)$$

donde la matriz $\boldsymbol{T}$ de $m \times m$ sobre GF(2) es triangular superior. Esto corresponde al método de eliminación gaussiana con pivoteo parcial sobre GF(2). Ya que $\boldsymbol{Q}$ es no-singular, de la ec. (4), $\boldsymbol{T}$ no es singular y sus elementos diagonales principales son todos 1. Similarmente, hagamos

$$\boldsymbol{L_i = P}_{i,m\text{-}1}\, \boldsymbol{P}_{i,m\text{-}2} \ldots \boldsymbol{P}_{i,i+1}$$

y de ahí

$$\boldsymbol{[I,c'']} = \boldsymbol{L}_{m\text{-}2}\, \boldsymbol{L}_{m\text{-}3} \ldots \boldsymbol{L}_0 \boldsymbol{[T,c']} \qquad (7)$$

donde **I** es la matriz identidad de orden $m \times m$. Entonces el vector columna $\boldsymbol{c''}$ es la solución requerida, es decir, $\boldsymbol{b = c''}$.

Estructura del arreglo para el procesamiento $\boldsymbol{Q}$

La estructura propuesta para la aplicación del método de Gauss-Jordan para triangular la matriz $\boldsymbol{Q}$ queda mostrada en la figura 3.

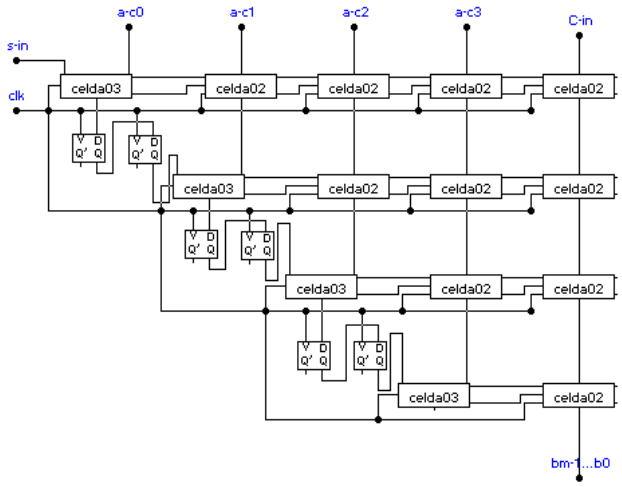


Fig. 3. Estructura para la Solución de la Matriz

Las señales indicadas en la figura son:

ac0-ac3 = Columnas de la matriz
S-in = Señal de control
C-in = Dividendo
clk = reloj del sistema
$\boldsymbol{b_{m\text{-}1}}$ **-** $\boldsymbol{b_0}$ = Solución del sistema de ecuaciones

En la estructura se utilizan dos tipos de celdas, la *celda02* y la *celda03*. Los biestables son utilizados para sincronización. El inicio para la introducción de los elementos de la matriz queda determinado por la señal de control. Las coordenadas de la solución $\boldsymbol{b}$ inician su salida de la $celda02_{m\text{-}1,m}$ en un tiempo $3m$. Posteriormente las coordenadas salen a un promedio de una coordenada por ciclo de tiempo, resultando en un tiempo de cómputo de 4m-1 ciclos. Las estructuras de las celdas se muestran enseguida:

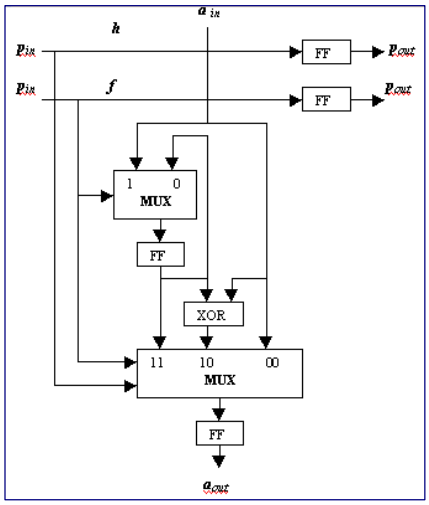


Fig. 4. Estructura de la celda02

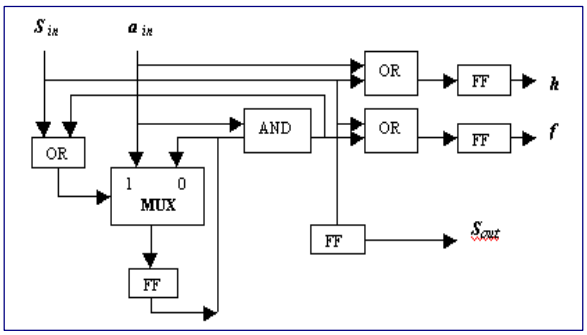


Fig.5. Estructura de la celda03

El esquema final del divisor se muestra en la figura 6.

Entre las dos etapas se puede apreciar un banco de biestables. Esto se introduce con el propósito de sincronizar la operación entre ambas etapas. El tiempo de operación para realizar la división es de 5*m*-1 ciclos, y el sistema permite una operación en línea *("pipeline")*.

III. SIMULACIONES Y PRUEBAS

Las simulaciones se hicieron utilizando el paquete computacional MAXPLUS-II de ALTERA. Enseguida se presentan la estructura *Gen-Mat* así como la estructura para la *Solución* desarrolladas con MAXPLUS-II.

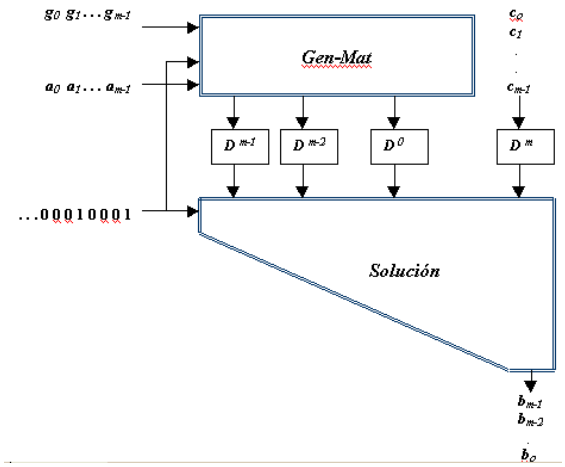


Fig.6. Estructura completa del Divisor

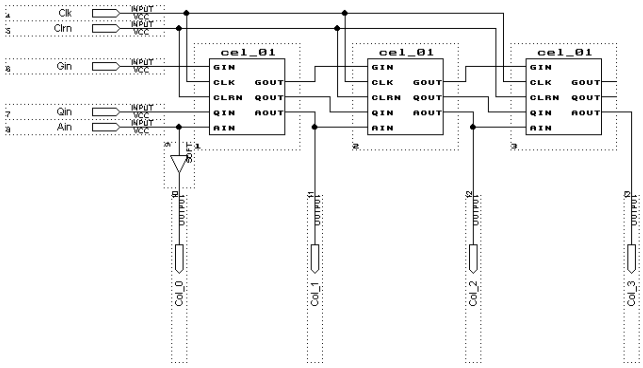


Fig. 7. Circuito Gen-Mat para GF(2^4)

Todas las simulaciones se realizaron considerando el campo GF(2^4) con polinomio irreducible

$g(x) = x^4 + x^3 + 1$

Potencia de α	Representación polinomial
α^0	1
α^1	α
α^2	α^2
α^3	α^3
α^4	$\alpha^3 + 1$
α^5	$\alpha^3 + \alpha + 1$
α^6	$\alpha^3 + \alpha^2 + \alpha + 1$
α^7	$\alpha^2 + \alpha + 1$
α^8	$\alpha^3 + \alpha^2 + \alpha$
α^9	$\alpha^2 + 1$
α^{10}	$\alpha^3 + \alpha^2$
α^{11}	$\alpha^3 + \alpha^2 + 1$
α^{12}	$\alpha + 1$
α^{13}	$\alpha^2 + \alpha$
α^{14}	$\alpha^3 + \alpha^2$
α^{15}	1

Tabla 1. Campo GF(2^4) con $g(x) = x^4 + x^3 + 1$

La simulación del Circuito Gen-Mat para GF(2^4) es la que se indica:

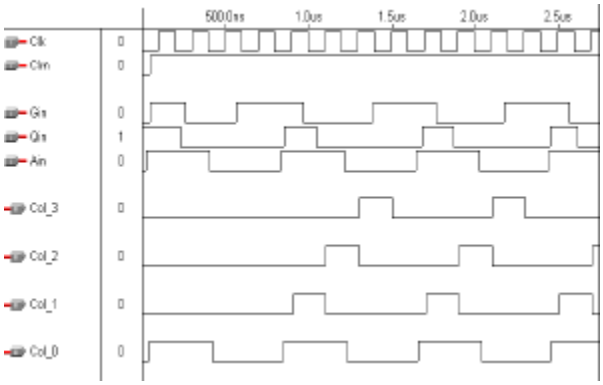


Fig.8. Simulación de Gen-Mat para GF(2^4)

Esta simulación corresponde al siguiente ejemplo:

Gin= $a^3 + 1$ = 1 0 0 1
Qin= 1 0 0 0
Ain= $a^{14} = a^3 + a^2$ = 1 1 0 0

y la matriz esperada es:

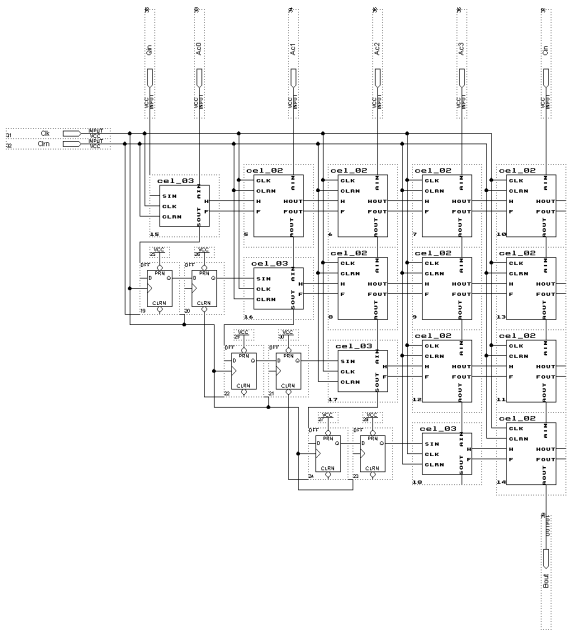
$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

es conveniente observar aquí que en la gráfica la columna 0 inicia en la subida del pulso 1, la columna 1 en el pulso 2, la columna 2 en el pulso 4 y la columna 3 en el pulso 6.

La estructura y simulación del circuito *Solución* se muestra en las figuras 9 y 10.

En este ejemplo se tienen los siguientes valores:

Ain = $a^6 = a^3 + a^2 + a + 1$ = 1 1 1 1
Cin = $a^{14} = a^3 + a^2$ = 1 1 0 0 (Después del 4º pulso)



Bout = a^8 = 1 1 1 0 (Después del pulso 12)
Fig. 9 Estructura del circuito Solución4

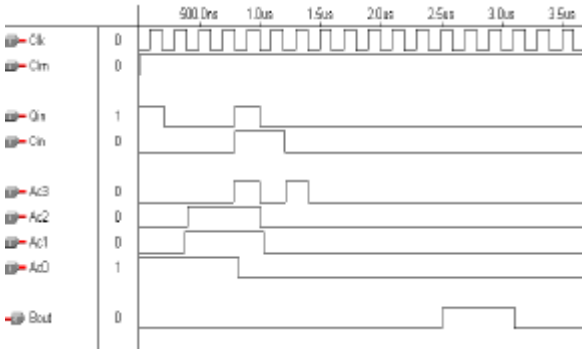


Fig.10. Simulación del circuito Solución4

En el esquema siguiente se pueden observar las etapas que integran al Divisor completo, es decir, el circuito *Gen-Mat*, la etapa intermedia de Flip-Flops para sincronizar las señales y el circuito *Solución.*

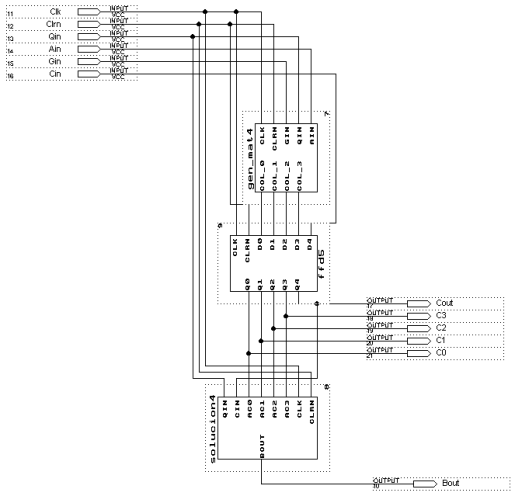
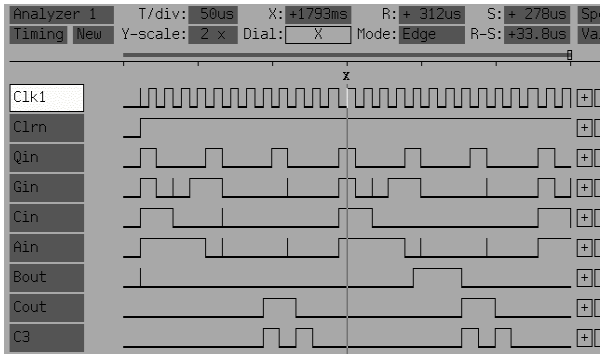


Fig.11. Estructura final del Divisor GF(2^4)

Una vez realizada la simulación del divisor, se grabó en un PLD EPM5128JC-2 de ALTERA.

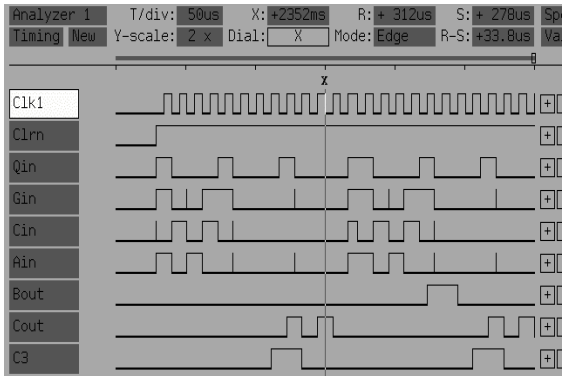
Las pruebas finales al PLD se realizaron aplicando diferentes vectores de prueba y tomando las salidas del circuito con la ayuda de un Analizador Lógico de Señales. Se muestran enseguida dos ejemplos de estas pruebas.

Ejemplo 1: $A_{in} = a^6 = a^3 + a^2 + a + 1 = 1\ 1\ 1\ 1$;
$C_{in} = a^{14} = a^3 + a^2 = 1\ 1\ 0\ 0$
$B_{out} = a^8 = a + a^2 + a^3 = 0\ 1\ 1\ 1$



Observe la salida en el pulso 16.

Ejemplo 2 : $A_{in} = a^{10} = a^3 + a = 10\ 1\ 0$
$C_{in} = a^9 = a^2 + 1 = 0\ 1\ 0\ 1$
$B_{out} = a^{14} = a^2 + a^3 = 0\ 0\ 1\ 1$



IV. CONCLUSIONES

En el presente trabajo desarrollamos una estructura serial para un Divisor sobre GF(2^4). Se ha requerido sólo de una señal de control y de interconexiones simples y regulares. Aprovechando estas ventajas, el circuito ha sido implantado en un PLD EPM5128JC-2 de Altera utilizando el paquete MAX PLUS que la misma empresa proporciona. En cuanto a sus características de operación, el ciclo de reloj no depende de *m*, el cálculo de la división requiere de 5*m*-1 ciclos de reloj y opera con característica "*pipeline*", por lo que el divisor es una muy buena propuesta para ser utilizado en aplicaciones de *códigos de corrección de errores* y en *criptografía.*

REFERENCIAS

[1] K. Akari, I.Fujita and M. Morisue, "Fast Inverter over finite field based on Euclid´s algorithm", *Trans. IEICE*, vol. E 72, pp. 1230-1234, Nov. 1989.

[2] C.C. Wang, T.K. Truong, H.M. Shao, L.J. Deutsch, J.K. Omura and I.S. Reed. "VLSI architecture for computing multiplications and inverses in GF(2)", *IEEE Trans. Comput*., vol C-34, pp. 709-717, Aug. 1985

[3] G.I. Davida, "Inverse of elements of a Galois Field", *Electron. Letters*, vol.8, pp. 518-520, Oct. 1972.

[4] M. Morii, M. Kasahara and D.L. Whiting, "Efficient bit-serial multiplication and the discrete- time Wiener-Hopf equations over finite fields", *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 1177-1183, Nov. 1989.

[5] M.A. Hasan and V.K. Bhargava "División and bit-serial multiplication over GF(q^m)", *IEEE Proc*. Part E, vol. 139, No. 3, pp. 230-236, May 1992.