

IMPLEMENTACIÓN DE UN ALGORITMO ICA CON CIRCUITOS CMOS ANALÓGICOS PARA EL PROBLEMA DE SEPARACIÓN CIEGA DE FUENTES

L. Noé Oliva Moreno, Oliverio A. Cárdenas, José A. Moreno Cadenas, Luis M. Flores Nava y Felipe Gómez Castañeda

Departamento de Ingeniería Eléctrica, CINVESTAV-IPN, Av. IPN, 2508
México D.F. , 07360, México
E-mail: noliva@sees.cinvestav.mx

RESUMEN

El presente artículo muestra la implementación de un algoritmo de Análisis de Componentes Independientes (ICA *Independent Component Analysis*) en un circuito integrado analógico con tecnología CMOS operando en saturación, para la solución del problema de separación ciega de fuentes (BSS *Blind Source Separation*). Se implementa una red neuronal de 3x3, basada en el algoritmo de Jutten y Herault (H-J) [1]. El diseño fue desarrollado para una tecnología de 1.2 μ m, con una área empleada de 630 μ m x 500 μ m. En este trabajo se muestran los resultados de simulación de cada etapa, así como el desempeño íntegro de la red.

La arquitectura de la red consiste de un filtro adaptable lineal recursivo continuo en el tiempo; y su algoritmo de aprendizaje, desarrollado por Jutten y Herault, se efectúa en tiempo real y en línea. La baja complejidad del algoritmo permite implementar dentro de un mismo chip un gran número de celdas y, en consecuencia, contar con más de dos entradas de fuentes.

1. INTRODUCCIÓN

La separación ciega de fuentes es un problema importante en el procesamiento de señales y varios métodos han sido empleados [2]. El algoritmo propuesto para la implementación busca separar un número importante de fuentes independientes, razón por la cual se implementa el de Jutten y Herault. Este algoritmo contiene una baja complejidad en sus funciones; sin embargo, para aplicaciones de audio presenta buenas características en la separación. Para simplificar la explicación del algoritmo se asumirá un arreglo de 2x2.

El problema esencial consiste en separar y estimar cada una de las fuentes de señal independientes a través de su mezcla. El modelo de mezcla está descrito por:

$$e = Ax \Leftrightarrow \begin{pmatrix} e_i \\ e_j \end{pmatrix} = \begin{pmatrix} a_{ii} & a_{ij} \\ a_{ji} & a_{jj} \end{pmatrix} \begin{pmatrix} x_i \\ x_j \end{pmatrix} \quad (1)$$

En donde: x es la señal desconocida, emitida por las fuentes originales, A es la combinación lineal o matriz de mezcla que se produce en el ambiente en donde se propagan dichas señales y e es la mezcla de las señales recibidas por los transductores.

El problema se concentra en determinar las fuentes originales, conociendo tan sólo las señales mezcladas (del transductor). La solución debe aplicarse a la siguiente expresión.

$$s = We \Leftrightarrow \begin{pmatrix} s_i \\ s_j \end{pmatrix} = \begin{pmatrix} 1 & -w_{ij} \\ -w_{ji} & 1 \end{pmatrix} \begin{pmatrix} e_i \\ e_j \end{pmatrix} \quad (2)$$

donde W es la matriz de pesos sinápticos. La matriz de pesos W debe tender a los valores de la matriz A^{-1} . Dos diferentes soluciones teóricas de los pesos sinápticos son posibles:

$$w_{ij} = a_{ij}/a_{jj} \text{ y } w_{ji} = a_{ji}/a_{ii} \quad (3)$$

$$w_{ij} = a_{ii}/a_{ji} \text{ y } w_{ji} = a_{jj}/a_{ij} \quad (4)$$

El algoritmo de aprendizaje empleado fue propuesto por Jutten y Herault [1]. Este ajusta el valor de los pesos sinápticos de la red en forma continua, basándose en el método del gradiente descendiente [3][4]:

$$\Delta w_{ij} = \frac{dw_{ij}}{dt} = \eta \cdot \mu(s_i) \nu(s_j) \quad (5)$$

$$\Delta w_{ji} = \frac{dw_{ji}}{dt} = \eta \cdot \mu(s_j) \nu(s_i) \quad (6)$$

En donde η es la razón de aprendizaje, μ y ψ son funciones no lineales impares. Algunas de las funciones propuestas por Jutten y Herault son del tipo: $\mu(s) = s^3$ y $\psi(s) = \tanh(s)$, no obstante $\psi(s) = (s)$ también conduce a la separación.

El ajuste de los pesos se realiza en forma continua de acuerdo a la siguiente expresión, conocida como la regla de Hebb o regla de adaptación, dada por [3][4]:

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij} \quad (7)$$

$$w_{ji}(t+1) = w_{ji}(t) + \Delta w_{ji} \quad (8)$$

La figura 1 muestra la arquitectura completa de la red neuronal de Jutten y Herault.

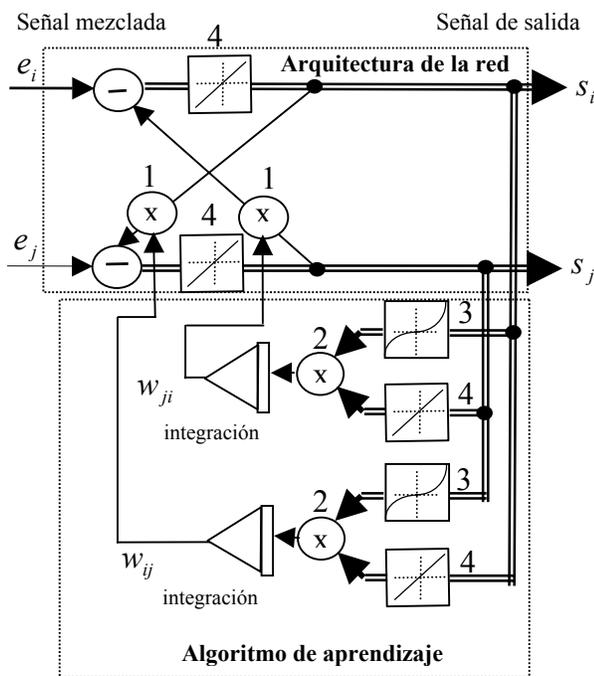


Fig. 1.- Arquitectura de la red neuronal y su Algoritmo de aprendizaje.

- 1.- Multiplicador de dos cuadrantes.
- 2.- Multiplicador de Gilbert de cuatro cuadrantes
- 3.- Generador de la función cúbica $\mu(s) = s^3$
- 4.- Función lineal $\psi(s) = (s)$

2. IMPLEMENTACIÓN EN SILICIO

La red neuronal se organiza por bloques, detallados a continuación.

El primer bloque está constituido por la arquitectura de la red y en segundo termino su algoritmo de aprendizaje. Se comenzará por explicar la arquitectura de la red.

ARQUITECTURA DE LA RED

La función de activación de la red neuronal, es una función lineal. El elemento empleado para este propósito es un amplificador de ganancia unitaria, con entrada y salida en modo diferencial. La figura 2 muestra el diagrama eléctrico de este bloque.

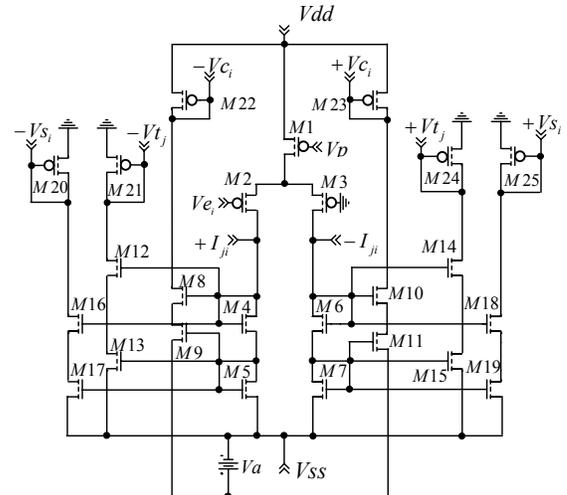


Fig. 2.- Circuito eléctrico de la función de activación lineal.

El transistor $M1$ proporciona la corriente del circuito, y opera en la región de saturación. En el transistor $M2$ se ingresa la señal de mezcla proveniente del transductor o micrófono. La señal es convertida a modo de corriente diferencial, facilitando la operación de sustracción requerida en la arquitectura de la red. El resultado de esta corriente diferencial es llevada a través de los espejos de corriente ($M4 - M19$) y posteriormente es convertida a voltajes diferenciales (Vs_i, Vt_j y Vc_i).

Cada voltaje es derivado en forma independiente, y alimenta cada una de las etapas posteriores. Vs_i proporciona la señal de salida de la red, en tanto que Vc_i y Vt_j proporcionan las señales requeridas en los bloques del algoritmo de aprendizaje. La fuente de voltaje Va es un voltaje de referencia que permite acoplar al circuito generador de la función cúbica con esta etapa, a través de Vc_i .

- Multiplicador de dos cuadrantes

Este multiplicador de dos cuadrantes realiza el producto del voltaje equivalente al peso sináptico (Vw_i) y el voltaje

de salida (V_{S_j}), generando su salida en modo de corriente diferencial ($-I_{ji}$ e $+I_{ji}$). La figura 3 muestra el diagrama eléctrico del multiplicador.

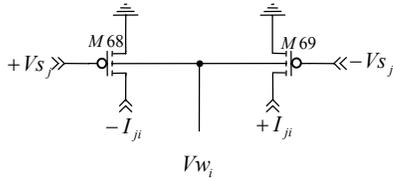
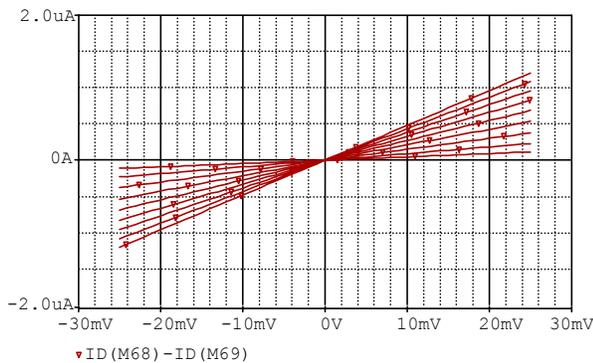


Fig. 3.- Multiplicador de dos cuadrantes

La expresión que describe a éste circuito es:

$$I_o = +I_{ji} - I_{ji} \approx \beta \gamma (V_{TO} + \sqrt{2|\phi_F|}(\gamma - 1)) \cdot [(+V_{T_j} - V_{T_j}) \cdot V_{W_i}] \quad (9)$$

La gráfica 1 muestra la corriente diferencial de salida del bloque de sinapsis. El intervalo dinámico de V_{S_j} fue de $\pm 25\text{mV}$; para valores de V_{W_i} de 0.2 a 1.1V, en donde cada recta representa un cambio de 0.1V.



Gráfica 1.- Salida del circuito multiplicador

La corriente diferencial de salida ($-I_{ji}$ y $+I_{ji}$) efectúa una operación de resta en el bloque de entrada de la función de activación de acuerdo a la figura 2.

ALGORITMO DE APRENDIZAJE

En este bloque se tratará primero el elemento que genera la función no-lineal y posteriormente se tratará el multiplicador de cuatro cuadrantes y la etapa de integración.

- Generador de la función cúbica

La función de éste circuito es la de generar una señal no-lineal. Esta función se emplea en el bloque del algoritmo de aprendizaje de la red, su salida y la entrada de este circuito se presentan en modo voltaje diferencial. El diagrama eléctrico se muestra en la figura 4. El voltaje V_{C_i} es el voltaje diferencial de entrada y $V_{S_{C_i}}$ es el voltaje diferencial de salida.

El voltaje V_{C_i} genera las corrientes que circulan a través de las ramas de transistores ($M38-M41$, $M42-M45$) los cuales actúan como divisores de voltaje, suministrando la alimentación para realizar el producto por medio de la arquitectura de dos multiplicadores de Gilbert [5]. Este multiplicador es analizado más adelante.

El primer multiplicador está formado por los transistores ($M26-M31$) y se realiza una operación cuadrática de V_{C_i} por medio de la alimentación del divisor de voltaje. El segundo multiplicador ($M32-M35$) toma las corrientes del primero y efectúa el producto con V_{C_i} . Obteniendo la función cúbica, en modo de corriente diferencial ($+I, -I$).

Las corrientes $+I$ y $-I$ son enviadas a los espejos de corriente formado por los transistores ($M46-M48$, $M49-M51$), generando el voltaje diferencial $V_{S_{C_i}}$.

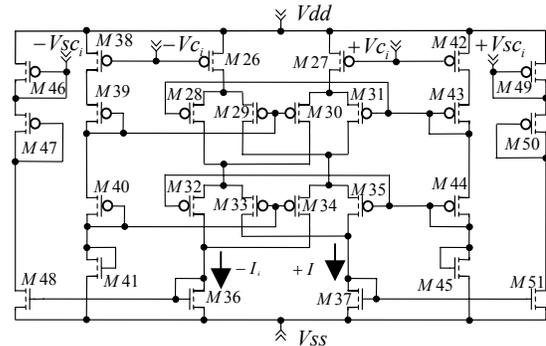
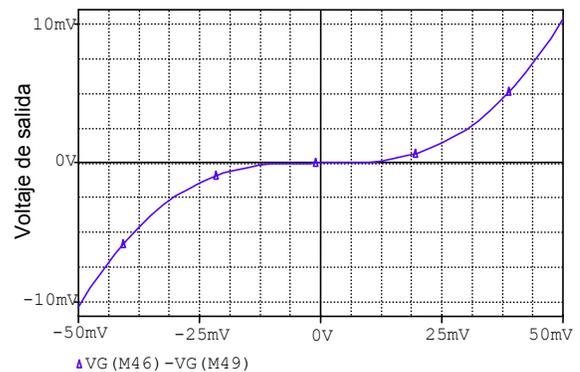


Fig. 4.- Circuito de la función cúbica.

La expresión que rige el comportamiento del circuito es:

$$V_1^T - V_1^T = \frac{\beta}{2\sqrt{2}} (V_1^{C+} - V_1^{C-})^3 \quad (10)$$

siempre y cuando los transistores estén trabajando en saturación. La gráfica 2 muestra la respuesta de salida del circuito. El intervalo dinámico de V_{C_i} fue de $\pm 50\text{mV}$.



Gráfica 2.- Salida del circuito cúbico

Una vez que se obtuvo la señal no-lineal deseada, la siguiente etapa trata con el diseño de un circuito multiplicador de Gilbert.

- Multiplicador de Gilbert de cuatro cuadrantes

En esta etapa se emplea un multiplicador de Gilbert de cuatro cuadrantes [5] para efectuar el producto de las dos funciones ($\psi(s) = (s)$ y $\mu(s) = s^3$). La señal de entrada (V_{t_i}, V_{sc_j}) se genero en modo de voltaje diferencial en las etapas previas. La figura 5 muestra el diagrama eléctrico del multiplicador.

El voltaje diferencial V_{sc_j} y V_{t_i} ingresa al multiplicador, a través de las compuertas de los transistores ($M52 - M57$).

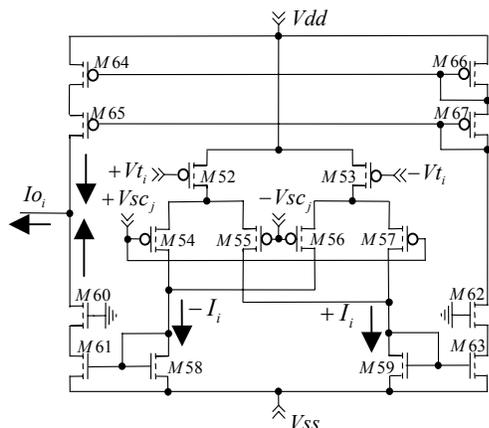
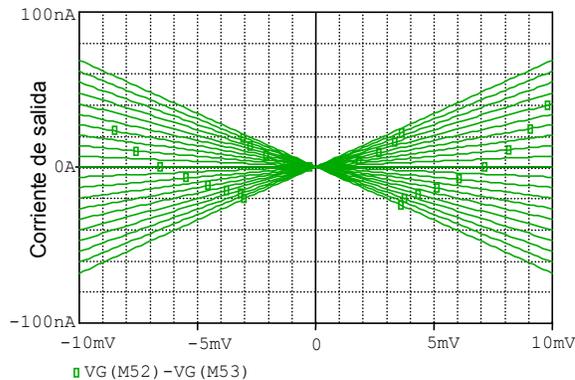


Fig. 5.- Multiplicador de Gilbert de cuatro cuadrantes.

Las corrientes resultantes ($-I_i, +I_i$) son enviadas a través de los espejos de corriente ($M58 - M67$) efectuándose la diferencia de corriente en el nodo I_{o_i} . La corriente de salida I_{o_i} se expresa como:

$$I_{o_i} = (i_{M54} + i_{M56}) - (i_{M55} + i_{M57}) = \frac{\beta}{2\sqrt{2}} (V_{t_i} - V_{t_i})(V_{sc_j} - V_{sc_j}) \quad (11)$$

La gráfica 3 muestra la respuesta de corriente del circuito multiplicador. El intervalo dinámico de V_{t_i} y V_{sc_j} son de $\pm 10\text{mV}$ y $\pm 50\text{mV}$ respectivamente, en donde cada recta representa un cambio de 2.5mV en V_{sc_j} .



Gráfica 3.- Salida del circuito multiplicador

La siguiente etapa integra esta corriente a través de un capacitor externo generando un voltaje (V_{w_i}), el cual es proporcional al valor del peso sináptico.

Una vez analizada la implementación de cada circuito que compone la red y su algoritmo de aprendizaje, se procede a las pruebas de simulación en el tiempo, de la red neuronal.

3. PRUEBAS DE SIMULACIÓN

Las pruebas de simulación fueron desarrolladas tomando el caso más simple, es decir para una red neuronal de 2×2 . En nuestro caso el diseño del chip que fue de 3×3 , la tercera rama de la red se inhabilito asignando un valor predeterminado al peso de dicha rama de tal forma que no afectara el desempeño de las restantes.

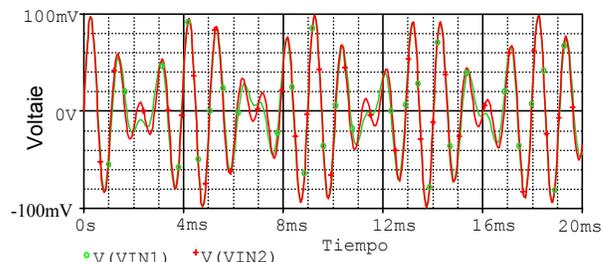
Las señales empleadas para estas pruebas fueron:

- 1er caso: x_1 .- Señal senoidal con frecuencia de 1KHz
- x_2 .- Señal senoidal con frecuencia de 800Hz
- 2do caso x_1 .- Señal de Voz No. 1
- x_2 .- Señal de Voz No. 2

Para el primer caso, las señales fueron mezcladas con la siguiente matriz de mezcla.

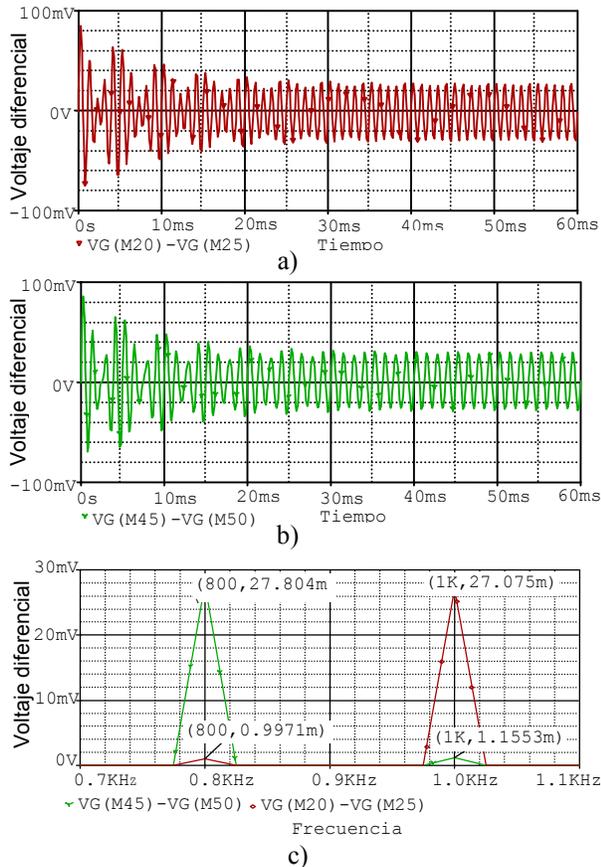
$$A = \begin{bmatrix} 1 & 0.83 \\ 0.83 & 1 \end{bmatrix} \quad (12)$$

Generándose las siguientes señales mezcladas.



Gráfica 4.- Señales mezcladas

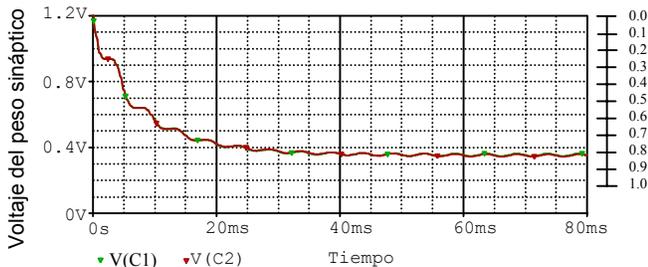
La salida de la red se puede observa en la gráfica 5.



Gráfica 5.- Señal de salida de la red a) V_{S_i} , b) V_{S_j} y c) Transformada rápida de Fourier de V_{S_i} y V_{S_j} después de 40ms.

Se observa a través de la salida de la red, la evolución de la separación de las señales; la separación se obtiene a partir de los 35ms aproximadamente.

La gráfica 6 muestra la evolución de los pesos sinápticos en el tiempo (a través del voltaje), en donde se observa que para este caso particular, el valor de ambos pesos siguen una misma trayectoria, debido a la matriz de mezcla empleada.

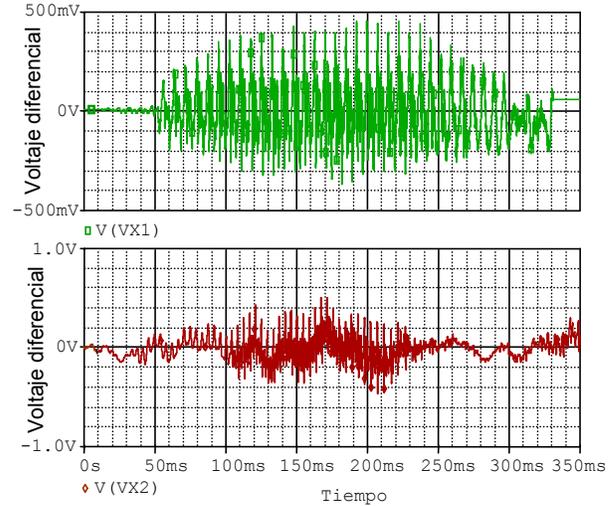


Gráfica 6.- Voltaje equivalente al peso sináptico

La relación señal a ruido que se obtuvo de la gráfica 5-c en este primer caso fue de 28dB y de 27dB para V_{S_i} y

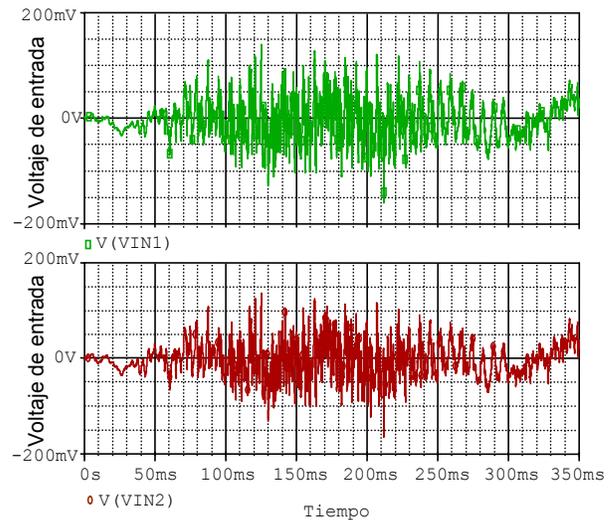
V_{S_j} respectivamente. El error de los coeficientes de la matriz de pesos fue de 0.4%.

Para el segundo caso, se grabaron dos señales de voces. La gráfica 7 muestra las señales independientes de dichas fuentes.

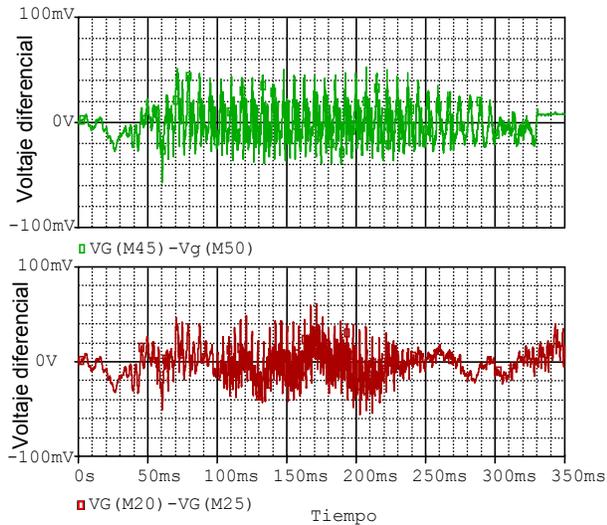


Gráfica 7.- Señales independientes de dos voces

Las señales fueron mezcladas con la matriz de mezcla del caso anterior (12). En la gráfica 8 se muestran las señales mezcladas. En la gráfica 9 se muestran las señales de salida de la red neuronal.

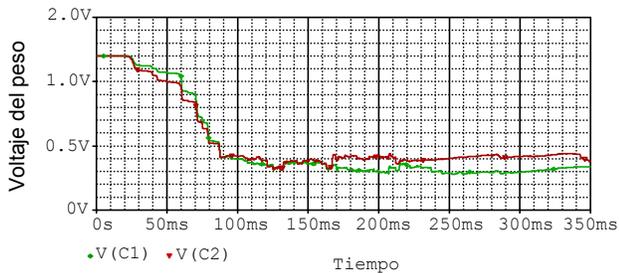


Gráfica 8.- Señales mezcladas de voces



Gráfica 9.- Señales de salida de la red V_{S_i} y V_{S_j}

La adaptación de los pesos sinápticos se muestra en la gráfica 10. En esta gráfica se observa la evolución de los pesos hasta estabilizarse, lo cual ocurre aproximadamente en 100ms.



Gráfica 10.- Voltaje equivalente al peso sináptico

La relación señal a ruido de esta etapa fue de 18dB y de 20dB, para V_{S_i} y V_{S_j} respectivamente.

En esta sección se analizó el desempeño en el tiempo de la red neuronal, aplicando dos casos diferentes. A continuación se muestra el diseño topológico del circuito.

4. DISEÑO TOPOLÓGICO DEL CIRCUITO

En este punto se muestra el diseño geométrico correspondiente a los circuitos eléctricos de la red neuronal. La figura 6 muestra el diseño geométrico del bloque de la red neuronal.

En este diseño topológico se buscó que la distancia entre transistores pares fuera la mínima permitida para reducir algunos de los fenómenos como el *mismatch* [6][7] y alguna variación por temperatura. La figura 7 muestra el diseño topológico del generador de la función cúbica.

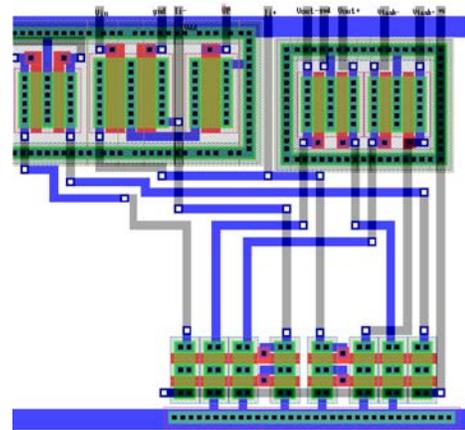


Fig. 6 Diseño topológico de la red neuronal

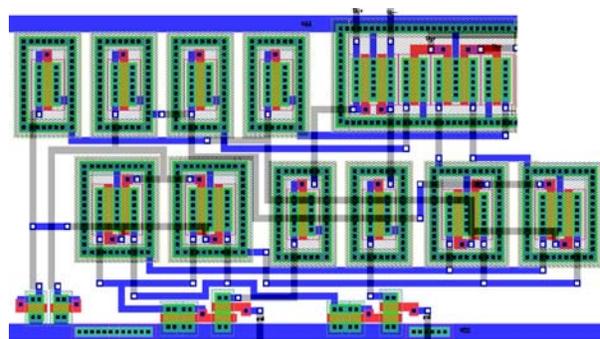


Fig. 7 Diseño topológico del generador cúbico

Las dos etapas restantes son el multiplicador de Gilbert de cuatro cuadrantes y el multiplicador de dos cuadrantes.

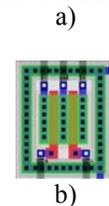
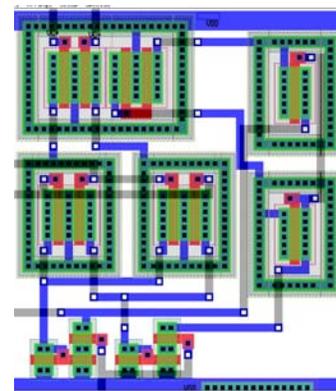


Fig. 8 Diseño de los multiplicadores a) de Gilbert de cuatro cuadrantes b) de dos cuadrantes

La arquitectura completa de la red se muestra en la figura 9, el diseño completo de este trabajo se realizó para una red de 3x3. El área ocupada por la red es de 630u x 500u.

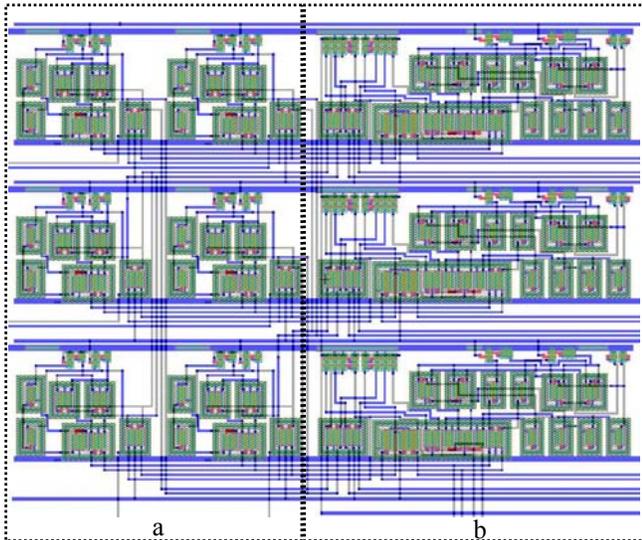


Fig. 9 Red Neuronal de 3x3

De la figura 9, la sección “a” de la arquitectura de esta red neuronal crece linealmente con el número de entradas, mientras que la parte “b” crece a una razón de (n^2-n) donde “n” es el número de entradas de la red.

5. CONCLUSIONES Y TRABAJOS A FUTURO

En este trabajo se presentó el diseño y la implementación analógica de una red neuronal que, a pesar de las limitaciones que pueda tener el algoritmo tratado, presenta resultados de simulación con buenas características para una aplicación de separación de fuentes audibles en tiempo real. La red fue simulada con dos señales de entrada audibles, previamente grabadas y mezcladas linealmente, presentándose para la matriz de mezcla (12) relaciones señal a ruido de alrededor de 20dB por salida. En un trabajo futuro se pretende incluir los pesos sinápticos en el chip a fabricar así como aumentar el número de entradas del mismo, contándose además con terminales que proporcionen información de algunos parámetros como el valor de los propios pesos sinápticos, resultando en una arquitectura multichip con múltiples entradas.

En la actualidad el diseño propuesto en este artículo se encuentra en proceso de fabricación y se espera que pronto se pueda realizar las mediciones experimentales correspondientes para su caracterización.

Actualmente, se está trabajando con un proyecto de tesis doctoral [7] en el desarrollo de un sistema el cual permite implementar un algoritmo de ICA para la visualización multidimensional de datos encefalográficos (EEG *Electroencephalographic*) con alta selectividad. El

algoritmo ha sido probado preliminarmente utilizando DSP's (TMS320C6713 y TMS320F2812), aplicando a las entradas varias señales audibles, demostrando así la efectividad del algoritmo en tiempo real para arreglos superiores a dos entradas.

6. REFERENCIAS

- [1] Jutten, C. Herault, J. “Blind separation of sources”, part I: An adaptive algorithm based on neuromimetic architecture *Signal Processing*, Vol. 24, pp. 1-10, 1991.
- [2] K. Fanghänel, K. Köllmann and H. Christoph “Parallel Bit-Stream Neurohardware for Blind Separation of Sources” *IEEE – ICASSP’97*.
- [3] A. Cichocki and R. Unbehauen, *Neural Networks for Optimization and Signal Processing*, John Wiley and Sons, 1993, pp. 461-471.
- [4] S. Haykin. *Neural Networks: a comprehensive foundation*. Prentice Hall, inc. 1999 pp 680-690.
- [5] Gunhee Han and Edgar Sánchez, “CMOS Transconductance Multipliers: A Tutorial”. *IEEE Transactions on circuits and systems-II: Analog and Digital Signal Processing*, Vol. 45, No. 12, December 1998.
- [6] J. Bastos, M. Steyaert, R. Roovers, P. K., W.S, “Mismatch characterization of small size MOS transistor”. *Proc. IEEE 1995 Int. Conference on microelectronic test structures*, Vol 8, Mar 1995.
- [7] J. M. Pelgrom, C. J. Duinmaijer, P. G. Welbers, “Matching Properties of MOS Transistors”. *IEEE journal of solid-state circuits*, vol. 24, n°5, Oct 1989.
- [8] L. N. Oliva “Sistemas de Procesamiento Neuronal para el análisis de información Multidimensional” trabajo de doctorado actualmente en desarrollo