(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 986 563**

(21) Número de solicitud: 202390120

(51) Int. CI.:

G06F 7/544 (2006.01)
G06N 3/065 (2013.01)

(12) PATENTE DE INVENCIÓN CON EXAMEN B2

(22) Fecha de presentación:

27.02.2023

(30) Prioridad:

25.02.2022 EP 22382166

(43) Fecha de publicación de la solicitud:

11.11.2024

(88) Fecha de publicación diferida del informe sobre el estado de la técnica:
26.12.2024

Fecha de modificación de las reivindicaciones:

26.02.2025

Fecha de concesión:

14.11.2025

(45) Fecha de publicación de la concesión:

21.11.2025

(73) Titular/es:

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA (100.00%)
Avda das Ciencias, 4 - 1D
15782 Santiago de Compostela (A Coruña) ES

(72) Inventor/es:

GARCIA LESTA, Daniel;
PARDO SECO, Fernando;
PEREIRA RIAL, Oscar;
LOPEZ MARTINEZ, Paula;
CABELLO FERRER, Diego y
BREA SANCHEZ, Victor Manuel

(74) Agente/Representante:

TORRENTE VILASÁNCHEZ, Susana

(54) Título: **PROCESADOR HIPERDIMENSIONAL DE SEÑAL MIXTA**

(57) Resumen:

Procesador hiperdimensional de señal mixta. Dispositivo para cálculo hiperdimensional que comprende: un conjunto de m x n unidades de procesamiento de 1 bit que comprenden una unidad lógica de 1 bit y una unidad de memoria; al menos un terminal de entrada para recibir un hipervector de entrada; n multiplexores de 1 bit para seleccionar la entrada. También, métodos para fabricar y utilizar el dispositivo.

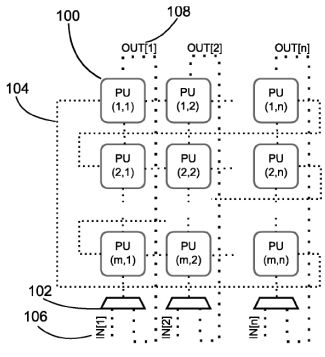


Figura 1

Aviso: Se puede realizar consulta prevista por el art. 41 LP 24/2015.
Dentro de los seis meses siguientes a la publicación de la concesión en el Boletín Oficial de la Propiedad Industrial cualquier persona podrá oponerse a la concesión. La oposición deberá dirigirse a la OEPM en escrito motivado y previo pago de la tasa correspondiente (art. 43 LP 24/2015).

ES 2 986 563 B2

DESCRIPCIÓN

PROCESADOR HIPERDIMENSIONAL DE SEÑAL MIXTA

SECTOR DE LA TÉCNICA

La invención se refiere a sistemas de hardware para tareas de computación hiperdimensional (en inglés Hyperdimensional Computing - HDC), más precisamente la invención se refiere a una arquitectura hardware multiprocesador de señal mixta para realizar tareas HDC.

ANTECEDENTES DE LA INVENCIÓN

La computación hiperdimensional (HDC) es un paradigma inspirado en el funcionamiento del cerebro que se basa en vectores de miles de dimensiones para representar datos. En los últimos años, la HDC ha abordado problemas de clasificación, mostrando resultados prometedores en comparación con los algoritmos de aprendizaje profundo más avanzados. La computación HDC implica operaciones simples, lo que la hace adecuada para implementaciones de hardware de bajo consumo de potencia.

Como ya se ha mencionado, la HDC es un paradigma inspirado en el funcionamiento del cerebro en el que los datos de baja dimensión, como las características de las imágenes, los valores de señales, texto, etc., se mapean en un vector con, por ejemplo, miles de componentes. De este modo, la información contenida en los datos de baja dimensión se distribuye a lo largo de las componentes del vector. El mapeo en un vector hiperdimensional (vector HD o HDV) se realiza a través de un codificador hiperdimensional (codificador HD) que proporciona, por ejemplo, la representación holística de los datos de baja dimensión, asignando HDV ortogonales a entradas de baja dimensión que son disimilares. El uso de la memoria asociativa (AM), como por ejemplo se explica en P. Kanerva, "Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors", Cognitive computation, vol. 1, nº 2, pp. 139-159, 2009, en el que los datos se recuperan comparando un HDV de consulta con el contenido de los datos de la memoria, es también un componente clave de los algoritmos de clasificación HDC. La HDC consta de tres primitivas básicas: vin7culación (en inglés binding), agrupación (en inglés bundling) y permutación. La vinculación de dos HDV binarios, V_1 y V_2, genera un HDV disímil, es decir, un HDV ortogonal a V_1 y V_2. La vinculación de HDV binarios se realiza con una puerta booleana XOR de bit a bit entre dos vectores HDC V_1 y V_2:

V_1 = 0 1 0 1 0 0 … 0 0
V_2 = 1 1 0 1 1 0 … 1 0

$V_1 * V_2$ = 1 0 0 0 1 0 … 1 0

La segunda operación básica es la agrupación de dos o más HDVs, que se ejecuta durante el entrenamiento para generar un vector de clase que represente el conjunto completo de HDVs de la clase. En el caso de los HDV binarios, la agrupación se realiza con la regla de la mayoría, que es una función bit a bit que asigna el valor lógico más frecuente para una posición de bit determinada entre los HDVs de entrada a la posición de bit correspondiente en el HDV de salida, como se muestra en el siguiente ejemplo:

V_1 = 0 1 0 1 0 0 … 0 0
V_2 = 1 1 0 1 1 0 … 1 0
V_3 = 0 0 1 0 0 1 … 0 1
V_4 = 1 1 1 1 0 0 … 1 0

$V_1 + V_2 + V_3 + V_4$ = 1 1 0 1 0 0 … 1 0

La última operación es la permutación, que rota los componentes del HDV un número determinado de posiciones adecuadas para codificar varios HDV en un HDV significativo para la tarea de clasificación. El proceso de clasificación se lleva a cabo comparando los HDV de consulta con los almacenados en una memoria asociativa (AM) mediante, por ejemplo, la distancia de Hamming (HD):

$$HD = \sum_{i=0}^{N} V_1(i)\ xor\ V_{am}(i)$$

donde N es el número de componentes del HDV, V_1 es el HDV de consulta y V_{am} es el HDV leído de la AM que representa una clase. El HDV V_{am} con la mínima distancia de Hamming proporciona la clase del HDV V_1.

La vinculación y la agrupación incluyen operaciones entre las mismos componentes de los HDV, mientras que la permutación y el cálculo de la distancia de Hamming implican operaciones entre diferentes componentes del HDV.

Los sistemas de hardware HDC desarrollados hasta ahora realizan las operaciones de la regla de la mayoría y el cálculo de la distancia de Hamming mediante circuitos digitales, lo que lleva a diseños con gran consumo de área y potencia, como se describe, entre otros, en A. Rahimi,

S. Datta, D. Kleyko, E. P. Frady, B. Olshausen, P. Kanerva y J. M. Rabaey, "High-dimensional computing as a nanoscalable paradigm", IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 64, n.º 9, pp. 2508-2521, 2017, y M. Schmuck, L. Benini y A. Rahimi, "Hardware optimizations of dense binary hyperdimensional computing: Rematerialización de hipervectores, agrupación binarizada y memoria asociativa combinacional", J. Emerg. Technol. Comput. Syst., vol. 15, oct. 2019.

Mika Laiho, Jussi H. Poikonen, Pentti Kanerva, Eero Lehtonen High-Dimensional Computing with Sparse Vectors. IEEE Biomedical Circuits and Systems Conference, 2015 presenta una arquitectura en la que los HDV dispersos se trocean en segmentos para realizar permutaciones. Sin embargo, la memoria asociativa sigue siendo una memoria global, presentando los mismos inconvenientes de los trabajos mencionados.

Existe interés en proporcionar procesamiento HDC con un consumo reducido de potencia y área.

.**EXPLICACIÓN DE LA INVENCIÓN**

Para superar las limitaciones de las implementaciones hardware de sistemas HDC descritas anteriormente, la presente invención proporciona una arquitectura de señal mixta para implementar un procesador hardware HDC. La invención aborda los requisitos de bajo consumo de potencia y área del procesador HDC para tareas de clasificación o inferencia.

Un primer aspecto de la invención se refiere a un dispositivo que comprende m x n unidades de procesamiento (PUs) de 1 bit, donde m y n son números naturales iguales o mayores que 2. Cada una de las unidades de procesamiento (PUs) incluye una unidad de memoria, preferiblemente una memoria asociativa, y cada PU está conectada localmente a sus cuatro PUs vecinas más cercanas a través de respectivos buses de comunicación. Las PUs vecinas más cercanas son normalmente las PUs directamente adyacentes a la respectiva PU, al norte, este, oeste y sur de la respectiva PU. En el caso de las PU de la primera fila, primera columna, última fila y/o última columna, una o dos PU vecinas no son directamente adyacentes, sino que se encuentran en la fila y/o columna opuesta, en particular: cada PU de la primera fila está conectada localmente a la PU de la misma columna en la última fila (p. ej. la PU de la posición 1 x 3 está conectada a la UP de la posición m x 3; cabe señalar que aquí la primera fila se identifica con el índice 1, pero también podría identificarse con el índice 0, y que la

última fila se identifica con el índice m, pero también podría identificarse con el índice m - 1, lo mismo se aplica a los ID de fila); cada PU de la última columna está conectada localmente a la PU de la fila siguiente y de la primera columna (por ejemplo, la PU en la posición 2 x n está conectada a la PU en la posición 3 x 1), y la última PU está conectada a la primera PU (es decir, la PU en m x n está conectada a la PU en 1 x 1).

El dispositivo además comprende una pluralidad de multiplexores, en particular un multiplexor por columna. Cada uno de estos multiplexores tiene su puerto de salida conectado a una PU diferente de la última fila, y tiene dos puertos de entrada, uno de los cuales está conectado a la PU de la primera fila que comparte la columna de la PU de la salida, y un terminal de entrada del dispositivo; por lo tanto, los multiplexores son multiplexores de 1 bit. Con el uso de multiplexores en la fila inferior es posible introducir n componentes procedentes del exterior del dispositivo, a través de al menos un terminal de entrada del dispositivo, en la matriz de PUs (también denominada, en lo sucesivo, hiperprocesador o procesador HDC). A continuación, estos datos pueden trasladarse a la fila siguiente en el mismo ciclo de reloj en el que se inserta un nuevo flujo de n bits en la fila inferior. De este modo, el hiperprocesador se cargará con el hipervector deseado tras m ciclos de reloj como máximo. La misma estrategia puede utilizarse para leer desde la primera fila un hipervector almacenado.

Se observará que, aunque se utilicen los términos filas y columnas, éstos no deben entenderse de forma limitativa. En este sentido, las columnas pueden ser horizontales y las filas verticales. De hecho, girando la matriz o el dispositivo 90º en el sentido de las agujas del reloj o en sentido contrario, la matriz m x n puede considerarse que se convierte en una matriz n x m. También se observará que términos como primera y última filas y columnas también están sujetos a cómo se inspecciona la matriz o dispositivo; una rotación de 180º o un reflejo de la matriz o dispositivo dará como resultado la misma disposición pero con la primera fila siendo la última fila, la primera columna siendo la última columna y viceversa, Una matriz o dispositivo con todas estas posibles interpretaciones también entran dentro del alcance de la presente invención.

El dispositivo objeto de la presente invención resuelve dos problemas presentes en las arquitecturas y sistemas de otros dispositivos, como se ha descrito anteriormente. En primer lugar, una unidad de memoria, es decir, memoria local, en cada PU implementa la memoria asociativa necesaria en los algoritmos HDC para las tareas de inferencia, lo que hace que las operaciones dentro de la PU sean más rápidas que en sistemas con memoria fuera de la PU;

en este sentido, la memoria de cada PU reduce el flujo de datos, ya que los resultados del procesamiento sólo necesitan almacenarse en memoria una vez y no dos (por ejemplo los registros del procesador para una regla de mayoría no necesitan almacenarse de nuevo en una memoria, ya que la memoria de la PU es una entrada para la regla de mayoría; y/o los vectores de perfil en los registros del procesador pueden cargarse en paralelo cuando se calcula la distancia de Hamming con respecto a un vector de consulta) y, por lo tanto, se reduce el consumo de energía, y aumenta la velocidad ya que cada PU tiene su propia memoria. En segundo lugar, la arquitectura de mixta simplifica las operaciones HDC de, por ejemplo, la regla de la mayoría y el cálculo de la distancia de Hamming por medio de circuitos analógicos, lo que reduce el área en comparación con las implementaciones totalmente digitales.

Como se ha mencionado anteriormente, el hiperprocesador del dispositivo consiste en m veces n PUs distribuidas en una matriz m x n, donde todas las PUs reciben las mismas señales de control, desde, por ejemplo, un circuito o unidad de control dentro del dispositivo, o desde fuera del dispositivo a través de uno o más terminales de entrada del dispositivo, con procesamiento SIMD (Single Instruction Multiple Data). Una posible aplicación de la conectividad 4 de la PU es la entrada/salida de datos al procesador HDC a través de la pluralidad de multiplexores.

Cada PU tiene capacidad de procesamiento local a través de una unidad lógica (LU) de 1 bit, que puede considerarse una unidad lógica aritmética, ya que puede realizar operaciones aritméticas de 1 bit, y está conectada a sus cuatro vecinas más cercanas, como se ha descrito anteriormente, siguiendo una arquitectura toroidal x-y. Este esquema permite mover datos a lo largo de los dos ejes perpendiculares, por ejemplo los ejes horizontal y vertical, que pueden utilizarse para realizar diferentes operaciones implicadas en HDC. Por ejemplo, una de las operaciones posibles son los desplazamientos circulares en ambas direcciones, muy utilizados en algoritmos comunes de cálculo hiperdimensional, gracias a la conexión de la última PU de cada fila con la primera de la fila siguiente. Además, la conectividad en el eje y permite realizar desplazamientos aleatorios más rápidamente que con sólo comunicación en el eje x.

La memoria local de cada PU almacena las componentes locales de los hipervectores correspondientes. De este modo, la LU y la memoria local permiten ejecutar cualquier función digital, y por tanto operaciones HDC, siempre que se disponga de la secuencia correcta de

señales de control.

Preferentemente, cada PU incluye toda la circuitería necesaria para las operaciones comunes de cálculo hiperdimensional. Para ello, cada PU comprende uno o más circuitos analógicos para realizar operaciones de regla de la mayoría y distancia Hamming.

En algunas realizaciones, m veces n es igual o mayor que 1000 (o 1024 si m y n son potencias de dos, lo cual es preferible ya que así se aprovecha al máximo la electrónica del dispositivo). Sin embargo, como será evidente para el experto en la materia, m veces n es típicamente mayor en HDC, ya que puede ser del orden de miles, decenas de miles o incluso mayor, y el dispositivo de la presente divulgación está configurado para realizar HDC con valores de m veces n de, por ejemplo, 4096 o mayor, 8192, o incluso mayor. En este sentido, m veces n debe ser igual o mayor que una dimensionalidad de un hipervector de entrada. Por ejemplo, en algunas realizaciones, m o n es 128 y el otro de m y n es 64.

En algunas realizaciones, la LU de cada PU se implementa con una puerta lógica. Por ejemplo, una puerta NOR o una puerta NAND, que proporciona funcionalidad completa. Preferiblemente, la puerta lógica tiene entradas con registros de memoria y una salida triestado conectadas al bus de comunicación de la propia PU o a cualquiera de los buses de comunicación de sus cuatro vecinas más cercanas.

En algunas realizaciones, las entradas de los registros de memoria recogen los datos del bus o una versión invertida de los mismos, en este último caso cada puerta lógica comprende además un inversor y un multiplexor.

En algunas realizaciones, la memoria local es una memoria SRAM. La comunicación entre PUs se lleva a cabo a través de los buffers triestado manejados por la puerta lógica, por ejemplo, puerta NOR, puerta NAND. Cada uno de ellos está conectado al bus de una vecina. Las direcciones este y oeste se utilizan para desplazamientos circulares de una posición habilitando sus correspondientes buffers. Cuando esto ocurre, el bus vecino seleccionado se carga con la salida de la puerta lógica, que puede escribirse en un registro de memoria del procesador o en la memoria local del vecino. Toda la información entre las PUs se comparte a través de un bus común de 1 bit, conectado a bloques funcionales como la unidad lógica, el circuito de regla de mayoría y un bloque de salida en modo corriente.

En algunas realizaciones, la unidad de memoria de cada PU comprende una memoria SRAM.

Cada columna y fila de la SRAM puede recibir una señal de control de, por ejemplo, un circuito o unidad de control dentro del dispositivo, o desde fuera del dispositivo a través de uno o más terminales de entrada del dispositivo, en un formato de un solo bit, en cuyo caso puede evitarse el uso de descodificadores a nivel de la PU (si los hubiera); de lo contrario, las señales de control pueden requerir descodificadores para convertir las palabras codificadas en una pluralidad de señales de control. Como cada PU es un procesador de 1 bit, sólo se puede escribir o leer 1 bit por ciclo de reloj.

Además, algunos bits, columnas o filas de la memoria pueden tener un doble propósito. El primero es mantener información general como los otros bits de la memoria, y el segundo es almacenar la entrada al bloque de regla de mayoría cuando se requiera.

En algunas realizaciones, cada unidad de procesamiento comprende un circuito analógico para el cálculo de la regla de la mayoría y un circuito analógico para el cálculo de la distancia de Hamming. La provisión de un circuito analógico para el cálculo de la regla de la mayoría simplifica la circuitería necesaria en comparación con una implementación completamente digital.

Para calcular el resultado de la regla de la mayoría en un conjunto de n hipervectores, el bloque de regla de la mayoría escribe los valores almacenados en las unidades de memoria. En las realizaciones en las que las unidades de memoria comprenden memoria SRAM, el bloque de cálculo de la regla de la mayoría escribe, mediante transistores, los valores almacenados en el SRAM en una pluralidad de condensadores. Los condensadores tienen preferentemente una respuesta lineal o una respuesta lo más lineal posible, en este sentido los condensadores podrían ser condensadores Metal-Aislante-Metal (MiM). También se podrían utilizar condensadores de otro tipo con respuesta no lineal, pero producirían que la precisión de las operaciones de cálculo de la mayoría fuese menor produciendo que el resultado de la operación fuese incorrecto.

Una operación común en computación hiperdimensional es la distancia de Hamming entre dos hipervectores para encontrar el perfil más cercano a un hipervector de consulta. La circuitería analógica permite realizar esta operación de forma sencilla. Una forma posible sería ejecutar la operación XOR entre cada componente del hipervector en cada PU y enviar el hipervector resultante a la circuitería digital situada dentro del dispositivo en la periferia del

hiperprocesador para contar el número de unos.

En algunas realizaciones, el dispositivo está configurado para recibir uno o más hipervectores de entrada, procesar el hipervector o hipervectores y proporcionar un resultado. El dispositivo puede utilizarse, por ejemplo, para la clasificación de imágenes y vídeos, el procesamiento de señales EGM, el reconocimiento de textos, etc. En algunas realizaciones, a modo ilustrativo y no limitativo, el dispositivo está configurado para recibir y procesar uno o más hipervectores de entrada correspondientes a una imagen digital o fotograma de un vídeo digital, y datos de salida indicativos de uno o más objetos presentes en la imagen digital o fotograma.

En un segundo aspecto la invención se refiere a un método para fabricar un dispositivo de cálculo hiperdimensional, que comprende: disponer una pluralidad de unidades de procesamiento de 1 bit formando una matriz m x n, siendo m y n cada una un número natural igual o mayor que 2; disponer una pluralidad de n multiplexores de 1 bit; conectar cada unidad de procesamiento a sus cuatro unidades de procesamiento vecinas más cercanas con la condición de: una unidad de procesamiento vecina para cada unidad de procesamiento de la primera fila de la matriz sea la unidad de procesamiento de la última fila que comparte la misma columna de la matriz; una unidad de procesamiento vecina para cada unidad de procesamiento de la última columna de la matriz sea la unidad de procesamiento de la primera columna y de la fila siguiente; y una unidad de procesamiento vecina para la unidad de procesamiento en la posición m x n sea la unidad de procesamiento en la posición 1 x 1; conectar una salida de cada multiplexor de 1 bit a una unidad de procesamiento diferente de la primera fila de la matriz; y conectar una primera entrada de cada multiplexor de 1 bit a la unidad de procesamiento de la última fila que tiene la misma columna que la unidad de procesamiento conectada a la salida, y una segunda entrada de cada multiplexor de 1 bit a un terminal de entrada del dispositivo. Los terminales de entrada permiten la comunicación del dispositivo con otros dispositivos mediante conexiones eléctricas.

En una realización de la invención, el dispositivo fabricado es un dispositivo según el primer aspecto de la invención.

Un tercer aspecto de la invención se refiere a un método para utilizar un dispositivo según el primer aspecto de la invención o un dispositivo fabricado con el método del segundo aspecto de la invención.

En algunas realizaciones de la invención, el método comprende: recibir, en el dispositivo, uno o más hipervectores correspondientes a una imagen digital o fotograma de un vídeo digital; procesar el uno o más hipervectores en el dispositivo; y emitir, en el dispositivo, datos indicativos de uno o más objetos presentes en la imagen digital o fotograma. El hipervector o los hipervectores pueden recibirse a través de uno o más terminales de entrada del dispositivo.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

Las modalidades detalladas en las Figuras se ilustran a modo de ejemplo y no a modo de limitación:

Las Figuras 1 y 2 muestran el esquema de E/S y la arquitectura del sistema de los procesadores HDC de acuerdo con las realizaciones.
La Figura 3 muestra una Unidades de Procesamiento (PU) de un procesador HDC de acuerdo con las realizaciones.
La Figura 4 muestra una circuitería para implementar la operación de regla de la mayoría en unidades de procesamiento de un dispositivo de acuerdo con las realizaciones.

REALIZACIÓN PREFERENTE DE LA INVENCIÓN

La Figura 1 muestra un hiperprocesador de un dispositivo para la computación hiperdimensional de acuerdo con las realizaciones.

El hiperprocesador incluye una pluralidad de unidades de procesamiento (PU) 100 dispuestas en una matriz o matriz m x n. Todas las unidades de procesamiento 100 están dispuestas, por ejemplo, en uno o varios circuitos integrados. La posición de cada PU 100 dentro del conjunto o matriz se muestra en el interior de la PU únicamente con fines ilustrativos.

Cada PU 100 está conectada a sus cuatro PUs 100 directamente adyacentes con buses de datos 104. La adyacencia puede considerarse como las PUs al norte, este, oeste y sur de cualquier PU 100 dada, donde cada una de estas coordenadas se considera en términos de la posición de las PUs 100 en el conjunto o matriz. Sin embargo, y como se representa en la Figura 1, una de las cuatro PUs 100 directamente adyacentes para las PUs 100 de la primera fila es la PU 100 de la misma columna pero de la fila inferior. Del mismo modo, una de las cuatro PUs 100 directamente adyacentes para las PUs 100 de la última columna es la PU 100

de la primera columna pero de la fila siguiente; en el caso de la última PU 100, se considera que la fila siguiente es la primera fila.

El hiperprocesador también incluye tantos multiplexores 102 como columnas tenga la matriz. Cada multiplexor 102 está conectado con las PUs 100 tanto de la primera como de la última fila de la misma columna; la conexión con la PU 100 de la primera fila es una salida 108 del multiplexor 102 respectivo, y la conexión con la PU 100 de la última fila es como una entrada 106 del multiplexor 102 respectivo. Además, cada multiplexor 102 tiene como entrada 106 directa o indirectamente conectada con un terminal de entrada del dispositivo de modo que pueda recibir hipervectores de entrada. Con la pluralidad de multiplexores 102, el hiperprocesador puede recibir n bits por ciclo de reloj, y proporcionar como salida n bits por ciclo de reloj.

Como se apreciará, el hiperprocesador representado en la Figura 1 también podría girarse o reflejarse, alterando así la disposición visual de los elementos mostrados en ella, pero manteniendo la misma disposición entre los elementos. Por ejemplo, el hiperprocesador podría girarse 90º como en la Figura 2.

La Figura 2 muestra un hiperprocesador de un dispositivo de cálculo hiperdimensional de acuerdo con las realizaciones.

En este ejemplo, el hiperprocesador incluye una pluralidad de unidades de procesamiento 100 dispuestas en una matriz o matriz m x n, con m columnas horizontales dispuestas y n filas verticales. Las mismas explicaciones proporcionadas con referencia a las realizaciones de la Figura 1 se aplican igualmente a las realizaciones de la Figura 2.

Alternativamente, el mismo hiperprocesador podría definirse como un array o matriz de n x m, con n filas horizontales y m columnas verticales. En ese caso, el número de multiplexores sería n y las conexiones entre las UP 100 seguirían también el esquema ilustrado en la Figura 2.

La Figura 3 muestra una unidad de procesamiento, como una unidad de procesamiento 100 de cualquiera de las Figuras 1 y 2, de un dispositivo de acuerdo con algunas realizaciones.

La unidad de procesamiento incluye una unidad de memoria 200, en este ejemplo una

memoria SRAM de 16 bits aunque será fácilmente aparente que otros tipos y tamaños de memorias son posibles sin apartarse del alcance de la presente invención.

La unidad de memoria 200 está conectada tanto al circuito analógico para el cálculo de la regla de mayoría 212 como a un bus de datos propio 216. El bus de datos 216 está conectado a una unidad lógica 204, y a una fuente de corriente controlada 220.

En este ejemplo, la unidad lógica 204 incluye una puerta lógica 202 en la forma de una puerta NOR, pero una puerta diferente podría ser utilizada en su lugar, como por ejemplo una puerta NAND. La puerta lógica 202 incluye entradas con latch 208, 210 y una salida triestado conectadas al bus de datos 216, o a través de las salidas 214 (donde N, E, W y S denotan norte, este, oeste y sur) a cualquiera de los buses de las cuatro PUs 100 adyacentes como se describe, por ejemplo, en la Figura 1. Las entradas 208, 210 pueden recoger datos del bus tal cual o en su versión invertida gracias a un inversor 206 y un multiplexor 218.

Para evitar el bloque digital adicional y el movimiento de todos los datos del hiperprocesador a la periferia del hiperprocesador al realizar los cálculos de distancia de Hamming, lo que aumentaría el área, el tiempo de procesamiento y el consumo de energía del dispositivo, en algunas realizaciones se lleva a cabo lo siguiente. La operación de distancia de Hamming se proporciona añadiendo una fuente de corriente 220 con un transistor NMOS, por ejemplo, polarizado con un espejo de corriente para obtener una corriente proporcional al número de unos después de ejecutar la operación XOR. Esta fuente de corriente habilita transistores controlados por la señal de control e_l y por la tensión del BUS, y tiene su salida conectada a un bus de columna que recoge la corriente de todas las PUs de una columna. Estos buses de columna también están conectados entre sí, sumando todas las corrientes del conjunto de PUs. Así, tras la operación XOR esta fuente de corriente queda habilitada y sólo aquellas en las que el resultado haya sido uno sumarán su corriente.

La Figura 4 muestra un circuito, como el circuito 212 de la Figura 3, para implementar la operación de regla de mayoría en unidades de procesamiento de un dispositivo de acuerdo con las realizaciones.

La regla de la mayoría puede calcularse sobre un conjunto de, por ejemplo, p hipervectores. El valor de p, que es un número natural, es menor que un tamaño k de la unidad de memoria, por lo que $p < k$; por ejemplo, p es 8 cuando el tamaño de la unidad de memoria es de 16 bits.

El valor de p es igual a un número de condensadores 300.

El bloque de regla de la mayoría escribe los valores almacenados de la unidad de memoria en los condensadores 300 a través de transistores 302, como por ejemplo transistores NMOS, utilizando la señal write_MA. Al mismo tiempo, un nodo V_A 304 se carga a un voltaje de precarga v_precharge 306 para mejorar el rendimiento del circuito cargando una capacitancia de entrada parásita de un primer inversor con un voltaje seleccionado; el primer inversor es el par de transistores NMOS y PMOS controlados por las señales e_MA y e_MAB. Esta tensión puede utilizarse también para polarizar el valor de salida, lo que resulta útil para decidir el resultado del caso de empate (mismo número de unos y ceros). Después de escribir la entrada en los condensadores 300, todos ellos se cortocircuitan con la señal e_MA 308 y su valor medio se almacena en el nodo V_A 304. Además, se activa el primer inversor triestado, ajustando la salida a la salida deseada invertida. Este inversor se utiliza como comparador; el inversor puede hacer frente al efecto de interruptores no ideales implementados con, por ejemplo, transistores NMOS. En algunas realizaciones, debido a las tensiones de umbral de los interruptores, éstos pueden desplazar la representación lógica "1" de una tensión alta, por ejemplo 1,8 V, a una tensión más baja. Como será evidente para el experto, los umbrales pueden tener que configurarse para tener en cuenta estas variaciones. Por último, para evitar la baja ganancia del primer inversor causada por la modificación de la tensión umbral, se añade preferentemente un segundo inversor triestado con bucle de realimentación a V_A 304, que se enciende con la señal MA de lectura después de que se estabilice la señal de salida.

En este texto, el término "comprende" y sus derivaciones (tales como "que comprende", etc.) no deben entenderse en un sentido excluyente, es decir, estos términos no deben interpretarse como excluyentes de la posibilidad de que lo descrito y definido pueda incluir otros elementos, pasos, etc.

Por otra parte, es evidente que la invención no se limita a la(s) realización(es) específica(s) aquí descrita(s), sino que también abarca cualquier variación que pueda considerar cualquier experto en la materia (por ejemplo, en cuanto a la elección de materiales, dimensiones, componentes, configuración, etc.), dentro del alcance general de la invención tal como se define en las reivindicaciones.

REIVINDICACIONES

1. Un dispositivo para computación hiperdimensional que comprende:
    a. una matriz de m x n unidades de procesamiento de 1 bit (100) dispuestas en m filas y n columnas, donde m y n son números naturales igual o mayores que 2;
    b. unidades de procesamiento (100) que comprenden una unidad lógica de 1 bit y una unidad de memoria;
    c. al menos un terminal de entrada para recibir un hipervector de entrada;
    d. n multiplexores de 1 bit (102), cada uno de los cuales tiene su puerto de salida conectado a una unidad de procesamiento diferente de la última fila, y tiene dos puertos de entrada, estando un puerto de entrada conectado a una unidad de procesamiento de la primera fila que comparte la columna de la unidad de procesamiento de la salida, y el segundo puerto de entrada está conectado a un terminal de entrada.

    en el que la matriz está dispuesta de tal manera que cada unidad de procesamiento está conectada a sus cuatro unidades de procesamiento vecinas más cercanas, donde:

        una unidad de procesamiento vecina para cada unidad de procesamiento (100) de la primera fila del conjunto es la unidad de procesamiento de la última fila que comparte la misma columna de la matriz, donde dicha conexión se hace a través de un multiplexor;

        una unidad de procesamiento vecina para cada unidad de procesamiento (100) de la última columna del conjunto es la unidad de proceso de la primera columna y de la siguiente fila;

        una unidad de procesamiento vecina para la unidad de procesamiento (100) en la posición m x n es la unidad de procesamiento en la posición 1 x 1.

2. El dispositivo según la reivindicación 1, en el que cada unidad de procesamiento (100) además comprende:
    a. un circuito analógico para el cálculo de la regla de mayoría (212); y
    b. un circuito analógico para el cálculo de la distancia de Hamming.
3. El dispositivo según la reivindicación 2, en el que el circuito analógico para el cálculo de la regla de mayoría de cada unidad de procesamiento (100) comprende:

a. una pluralidad de condensadores (300) que almacenan componentes de un vector hiperdimensional sobre el que se calcula la regla de mayoría;

b. elementos de control (302) para cargar un componente del vector hiperdimensional de la unidad de memoria en el condensador (300) correspondiente; y

c. un elemento de precarga (306) para cargar un nodo (304) en el que la pluralidad de condensadores (300) está conectada para sesgar el resultado de salida un caso de empate en el cálculo de la regla de la mayoría.

4. El dispositivo según cualquiera de las reivindicaciones anteriores, en el que la unidad lógica de cada unidad de procesamiento (100) comprende:

a. una puerta lógica (202) con entradas con latch (208, 220);

b. una salida triestado conectada a un bus (216) de la unidad de procesamiento (100); y

c. búferes para cada conexión con una de las cuatro unidades de procesamiento vecinas más próximas.

5. El dispositivo según la reivindicación 4, en el que la puerta lógica (202) de cada unidad lógica es una puerta NOR.

6. El dispositivo según la reivindicación 4, en el que la puerta lógica (202) de cada unidad lógica es una puerta NAND.

7. El dispositivo según cualquiera de las reivindicaciones anteriores, en el que la unidad de memoria de cada unidad de procesamiento (100) es una memoria SRAM.

8. El dispositivo según una cualquiera de las reivindicaciones anteriores, en el que m por n es igual o mayor que 1024.

9. El dispositivo según la reivindicación 8, en el que m por n es igual o mayor que 8196.

10. El dispositivo según una cualquiera de las reivindicaciones anteriores, en el que el dispositivo está configurado para recibir y procesar uno o más hipervectores de entrada correspondientes a una imagen digital o fotograma de un vídeo digital, y configurado además para emitir datos indicativos de uno o más objetos presentes en la imagen digital o fotograma.

11. El dispositivo según una cualquiera de las reivindicaciones anteriores, en el que las filas son filas horizontales y las columnas son columnas verticales.

12. El dispositivo según cualquiera de las reivindicaciones 1 a 10, en el que las filas son filas verticales y las columnas son columnas horizontales.

13. Un método para fabricar un dispositivo para computación hiperdimensional, que comprende:

disponer una pluralidad de unidades de procesamiento de 1 bit y una unidad de memoria, formando una matriz m x n, siendo m y n cada uno un número natural igual o mayor que 2;

disponer una pluralidad de n multiplexores de 1 bit;

conectar cada unidad de procesamiento a sus cuatro unidades de procesamiento vecinas más cercanas con la condición de:

una unidad de procesamiento vecina para cada unidad de procesamiento de la primera fila de la matriz sea la unidad de procesamiento de la última fila que comparte la misma columna de la matriz, donde dicha conexión se hace a través de un multiplexor;

una unidad de procesamiento vecina para cada unidad de procesamiento de la última columna del conjunto es la unidad de procesamiento de la primera columna y la siguiente fila; y

una unidad de procesamiento vecina para la unidad de procesamiento en la posición m x n es la unidad de procesamiento en la posición 1 x 1;

conectar una salida de cada multiplexor de 1 bit a una unidad de procesamiento diferente de la última fila de la matriz; y

conectar una primera entrada de cada multiplexor de 1 bit a la unidad de procesamiento de la primera fila que tiene una misma columna que la unidad de procesamiento conectada a la salida, y una segunda entrada de cada multiplexor de 1 bit a un terminal de entrada del dispositivo.

14. El método según la reivindicación 13, en el que el dispositivo es un dispositivo según una cualquiera de las reivindicaciones 1 a 12.

15. Un método de utilización de un dispositivo según cualquiera de las reivindicaciones 1 a 12, que comprende:

recibir, en el dispositivo, uno o más hipervectores correspondientes a una imagen digital o fotograma de un vídeo digital;

procesar uno o más hipervectores en el dispositivo; y

proporcionar una salida, en el dispositivo, de datos indicativos de uno o más objetos presentes en la imagen o fotograma digital.

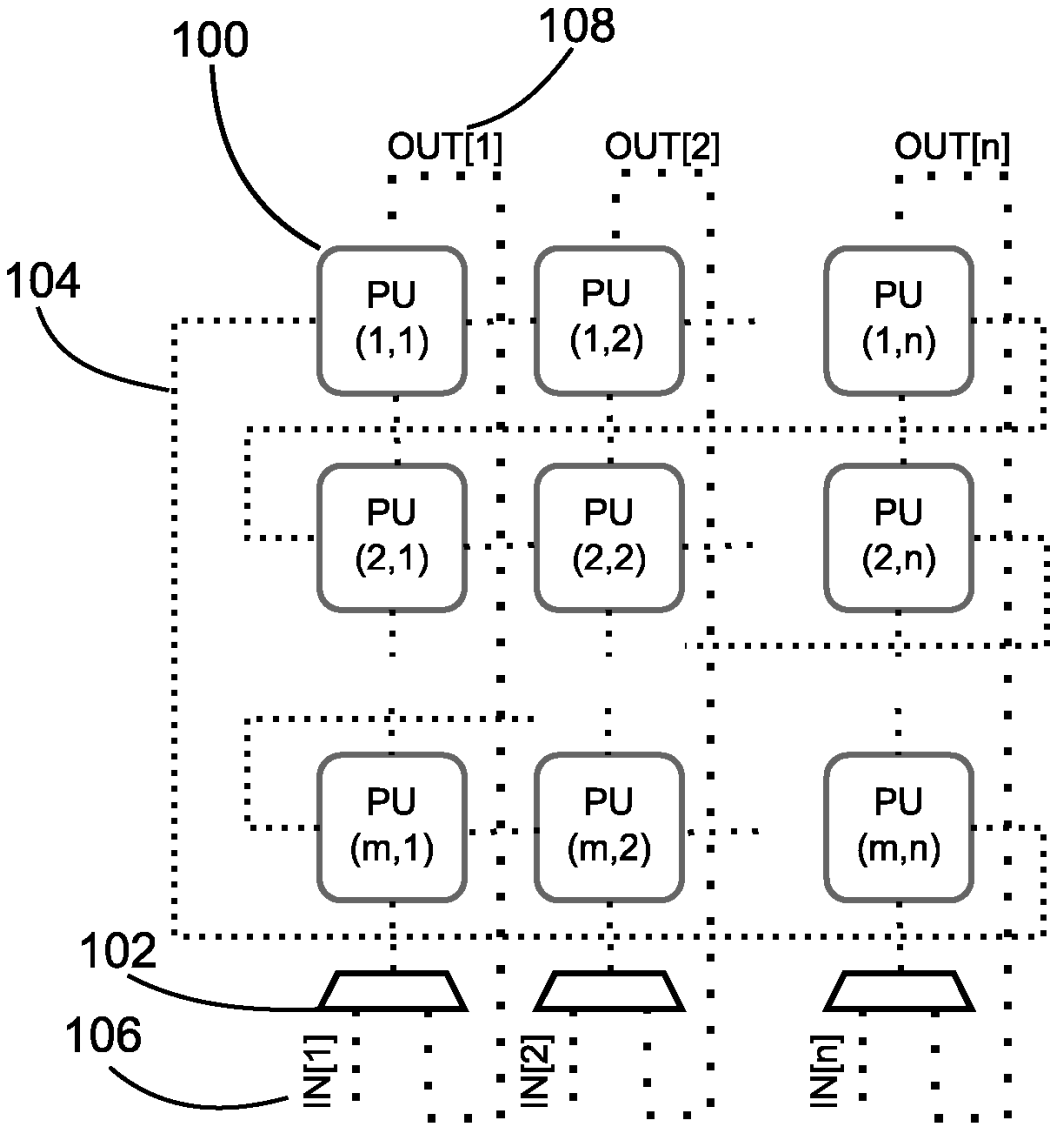

100
108
OUT[1]
OUT[2]
OUT[n]
104
PU (1,1)
PU (1,2)
PU (1,n)
PU (2,1)
PU (2,2)
PU (2,n)
PU (m,1)
PU (m,2)
PU (m,n)
102
106
IN[1]
IN[2]
IN[n]

Figura 1

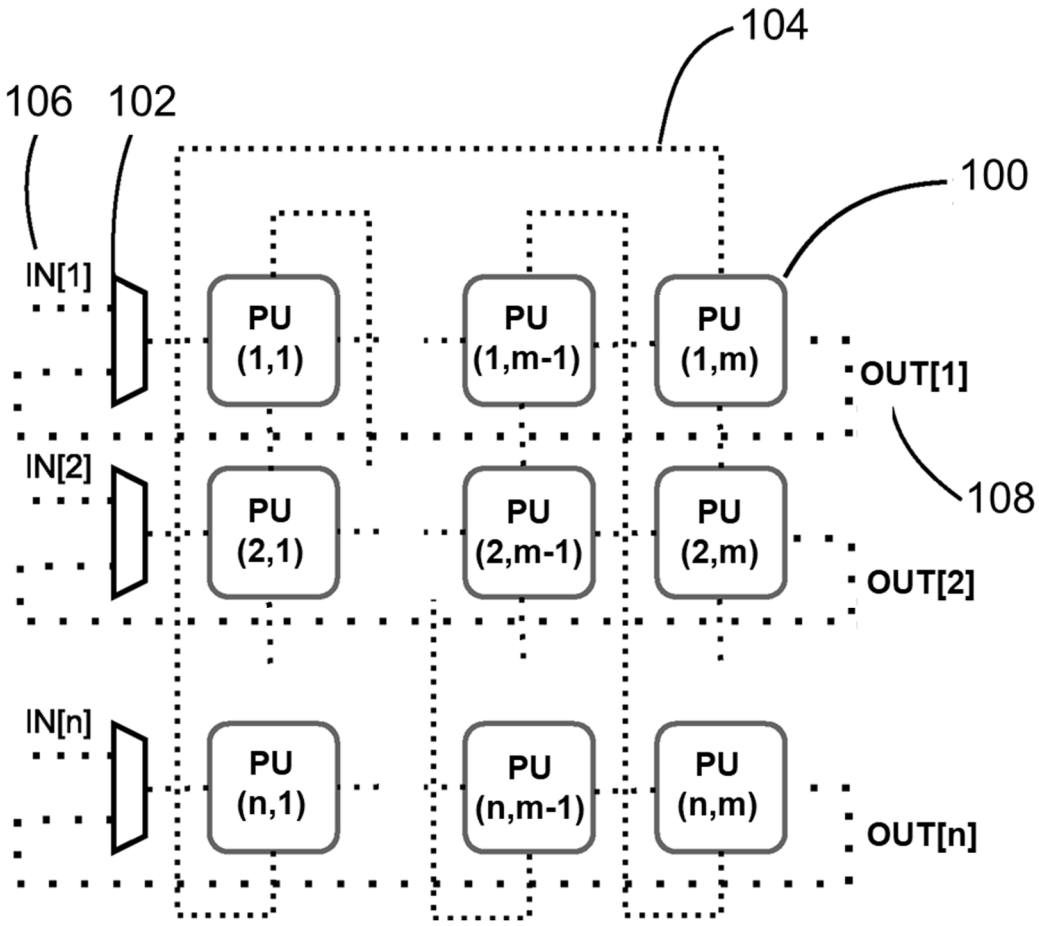

104
106
102
100
IN[1]
PU (1,1)
PU (1,m-1)
PU (1,m)
OUT[1]
108
IN[2]
PU (2,1)
PU (2,m-1)
PU (2,m)
OUT[2]
IN[n]
PU (n,1)
PU (n,m-1)
PU (n,m)
OUT[n]

Figura 2

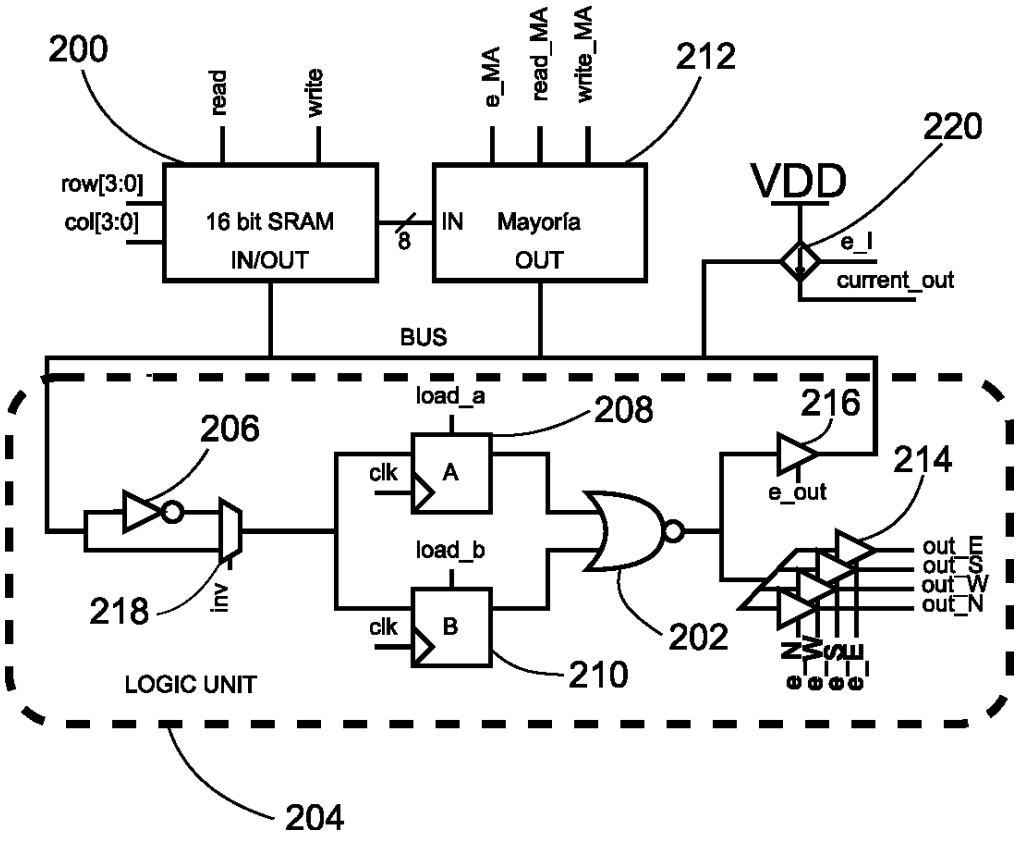

200
read
write
e_MA
read_MA
write_MA
212
220
VDD
row[3:0]
col[3:0]
16 bit SRAM
IN/OUT
8
IN
Mayoría
OUT
e_I
current_out
BUS
load_a
208
206
216
clk
A
e_out
214
load_b
out_E
out_S
out_W
out_N
218
inv
clk
B
202
210
LOGIC UNIT
204

Figura 3

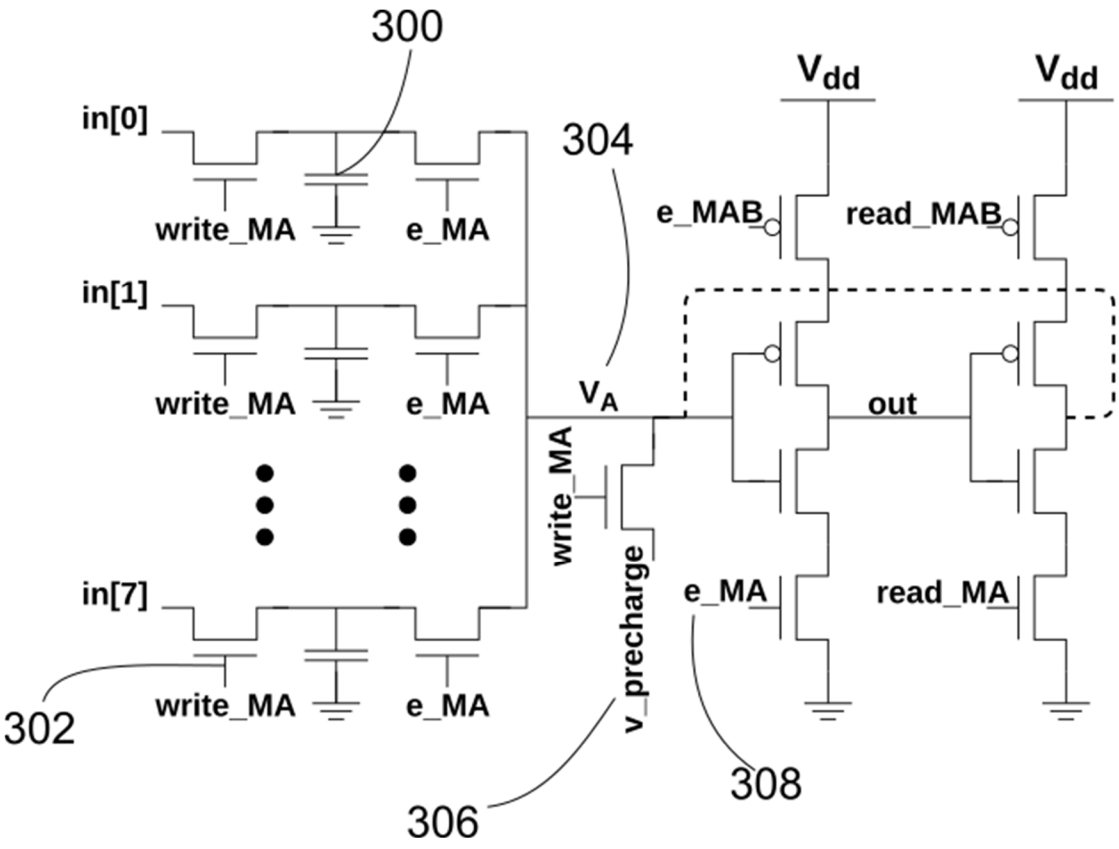

300
304
in[0]
write_MA
e_MA
in[1]
write_MA
e_MA
in[7]
write_MA
e_MA
302
306
308
V_{dd}
V_{dd}
e_MAB
read_MAB
V_A
out
write_MA
v_precharge
e_MA
read_MA

Figura 4