Implementación del Teorema de Cantor en la Solución del Direccionamiento de Memoria Mapeada

M. en C. Gustavo A. Mas Levario, M. en C. María A. Segura Corona, M. en C. Pablo Manrique Ramírez.
Profesores del CIC-IPN

El presente artículo propone una solución descriptiva al problema del mapeo computacional sobre la base del Teorema de Cantor. Para lograrlo se propone una serie de fundamentos heurísticos que permitan relacionar el Procedimiento de Cantor con el significado del direccionamiento de memoria mapeada que rebasa el tamaño del mapa de memoria de los procesadores.

INTRODUCCIÓN

Este trabajo sigue una línea pragmática principalmente, y hace uso de una serie de asociaciones entre la realidad del direccionamiento de memoria y la abstracción del Teorema de Cantor. El objetivo principal de esta asociación es proporcionar sustento a algo que está fuera del mundo real y que tiene su origen en el universo de los conceptos. El significado de sustentar aquí es materializar un concepto matemático en un campo de acción de la realidad mediante una idea o un pensamiento intuitivo, en este caso el campo de acción es el diseño de sistemas digitales para direccionar memoria mapeada.

Se sabe que en la generalización simple, la abstracción es sólo un instrumento de la operación mental entre otros, con el cual aislamos las cualidades que se desean obtener. Un pragmático debe partir de una condición de entidades con cualidades a una condición de cualidades sin entidades para plantear un problema de ingeniería bajo la abstracción, la precisión y el rigor lógico de ciertos procedimientos matemáticos, como en este caso el Procedimiento de Cantor[2].

Para poder materializar un concepto se debe entender que en la formación de los conceptos obtenidos por generalización simple aplicada sobre determinadas propiedades de los objetos reales o no, tales como el color de nuestra silla o el color de nuestra primera bicicleta, es posible distinguir tres fases de operaciones mentales. En la primera fase la propiedad se define comparando directamente los objetos: como en el caso de ciertas culturas primitivas en donde se asocian tantas manzanas como dedos de la mano. En la segunda aparece un adjetivo: cinco direcciones de memoria, o en casos un poco más abstractos: memoria mapeada, aquí el adjetivo es mapeada. En la tercera fase se abstrae la propiedad de los objetos y pueden aparecer cosas como verde, o el número abstracto 5 e incluso el adjetivo mapeado o mapeada.

Este estudio en sí trata con la segunda fase, ya que en lugar de confrontar dos hechos reales, se usa la técnica de la asociación de un hecho teórico ya estudiado con un hecho real sin estudio, desde la perspectiva de ese concepto. Para comprobar su funcionalidad se aplica la generalización simple a diferentes circunstancias del hecho real, postulando la obtención de su concepto que, como se verá, es el concepto del hecho teórico.

El hecho teórico es el Teorema de Cantor, que describe que la magnitud de una función y la magnitud de su variable siempre son inconmensurables. El hecho real que se desea estudiar es el originado al direccionar memoria mapeada en un nodo de procesamiento con memoria local. De este modo tenemos que la variable independiente es cada dirección absoluta del mapa de memoria del procesador, y su función es el mapeo de su dirección correspondiente sobre la memoria, en un momento dado.

En la actualidad hay sistemas de cómputo dedicados a aplicaciones específicas constituidos por nodos de procesamiento, en donde los procesadores tienen capacidades de direccionamiento limitado, pero que en ciertas aplicaciones requieren almacenar cantidades de datos que desbordan el límite del mapa de memoria de dichos procesadores.

El mapa de memoria de un procesador es el conjunto máximo de direcciones absolutas que puede referenciar desde su unidad de direccionamiento; de este modo, un procesador con 16 líneas de direccionamiento sólo puede alcanzar hasta 65536 direcciones absolutas, pero ciertas aplicaciones de diseño obligan a este procesador a demandar de su unidad de direccionamiento más de estas 65536 direcciones absolutas, que estarán en correspondencia biunívoca con 131072 direcciones mapeadas.

Ese inconveniente exige la existencia de otra línea de dirección en la unidad de direccionamiento del procesador misma que es imposible de obtener, así que es aquí donde se recurre a un procedimiento que permita concebir este problema computacional como un problema a resolver de la matemática aplicada.

Este procedimiento indica que siempre habrá la posibilidad de direccionar de un modo u otro más de lo que aparentemente se permite a partir del mapa de memoria de un procesador.

Se presenta lo anterior como un teorema computacional a demostrar tomando como su lema el Teorema de Cantor.

MÉTODO

Lo que se pretende es llevar el plano cartesiano del mapeo al plano de un cuadrado, en donde el origen del plano cartesiano coincide con el vértice inferior izquierdo del cuadrado; de este modo, la función del mapeo se superpone a la diagonal que se traza del vértice inferior izquierdo al vértice superior derecho, el eje de las abscisas se superpone al segmento de la base del cuadrado, y el eje de las ordenadas se superpone al segmento vertical izquierdo del cuadrado.

En la **figura 1** se muestra computacionalmente en donde estarían ubicados el lema y el teorema que se desea demostrar. El Teorema de Cantor es el siguiente:

"Los puntos del plano siguen la potencia del continuo".

El método se fundamenta en los tres aspectos[1] de las abstracciones de la matemática, con ellos se formulan las asociaciones necesarias que dan la estructura del procedimiento que se desarrollará en el método:

1. *Las formas espaciales,* que en este caso son la geometría euclídeana necesaria para diseñar los esquemáticos de un diseño digital, así como los principios necesarios para el plano cartesiano y el cuadrado.

2. *Las relaciones cuantitativas* del hecho real, o sea la aritmética correspondiente para el nombramiento de las localidades del direccionamiento de un nodo de procesamiento.

3. *La sucesión de grados de abstracción creciente,* que en este caso en particular son ciertos principios de la geometría analítica.

Con el primer aspecto enumerado anteriormente se traza un bosquejo, en donde las líneas rectas representan a las señales que emite un procesador a las memorias y, en general, a cualquier dispositivo de entrada/salida. Los cuadriláteros representan a las unidades de cómputo como son los procesadores, los mapeadores y las memorias. Este aspecto, en conjunto con el segundo, permiten localizar e identificar cada entidad llamada dirección con el fin de poder nombrarlas con los números reales. En la **figura 2** se muestra un diagrama esquemático de un nodo de procesamiento.

Sin embargo, antes de entrar en detalle con relación a los tres aspectos es necesario resaltar la importancia del concepto de los números reales. Los números reales[1] representan un instrumento seguro y poderoso para la investigación matemática de

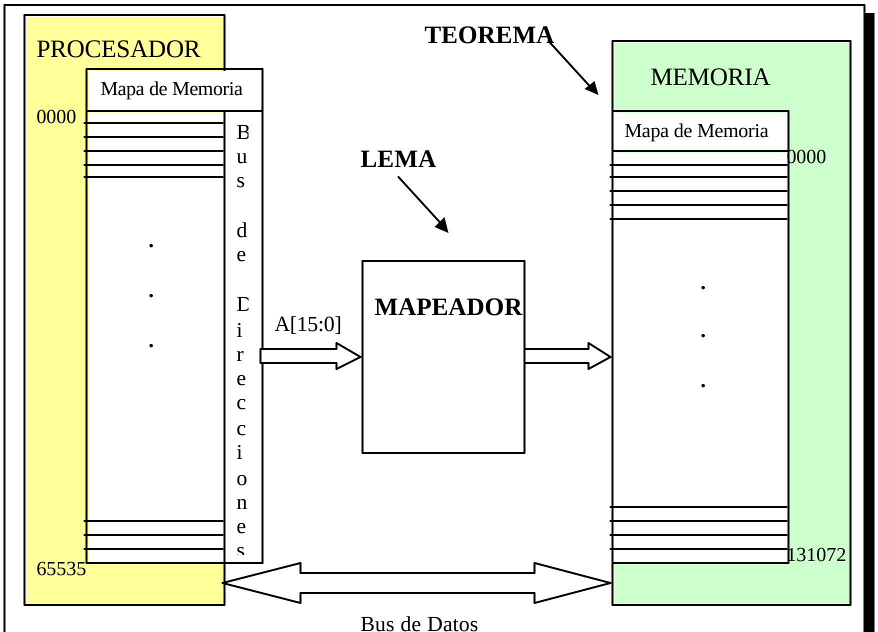

Figura 1. Diagrama del mapeo computacional donde se ilustra el Lema y Teorema de Cantor.

las magnitudes y procesos realmente continuos, tales como las direcciones del mapa de memoria y el mapeo de direcciones; la experiencia demuestra que el concepto de número real refleja con precisión las propiedades generales de las magnitudes continuas.

Así, con el segundo aspecto se propone cuantificar el direccionamiento realizado por un procesador mediante la aritmética. Para iniciar se parte de un axioma que se considera el más evidente en el direccionamiento. Se observa que en el direccionamiento, el nombramiento de las direcciones absolutas de los procesadores y memorias actuales va de modo consecutivo en incrementos unitarios de una dirección a otra hacia direcciones más altas, o en decrementos unitarios hacia otras más bajas, y este patrón es constante. Además, el mapeo básicamente es un problema de correspondencias entre las direcciones absolutas y las direcciones mapeadas, siendo esta relación lineal. De acuerdo a esto, se conciben el primero y el segundo de los postulados que se presentarán más adelante.

En general, se utilizan números enteros positivos para nombrar a cada localidad direccionada; no se nombra con números negativos ni imaginarios. Según este axioma se puede concebir parcialmente el lugar geométrico para el mapeo que se describe con el tercer postulado.

Postulado 1: la relación entre las direcciones absolutas y direcciones mapeadas es lineal, y por lo tanto una línea recta es la definición de esa relación.

Postulado 2: la razón entre cualquier par de nombramientos unitarios de las direcciones absolutas y mapeadas se supone igual a 1.

Postulado 3: por lo tanto, el plano cartesiano en su primer cuadrante

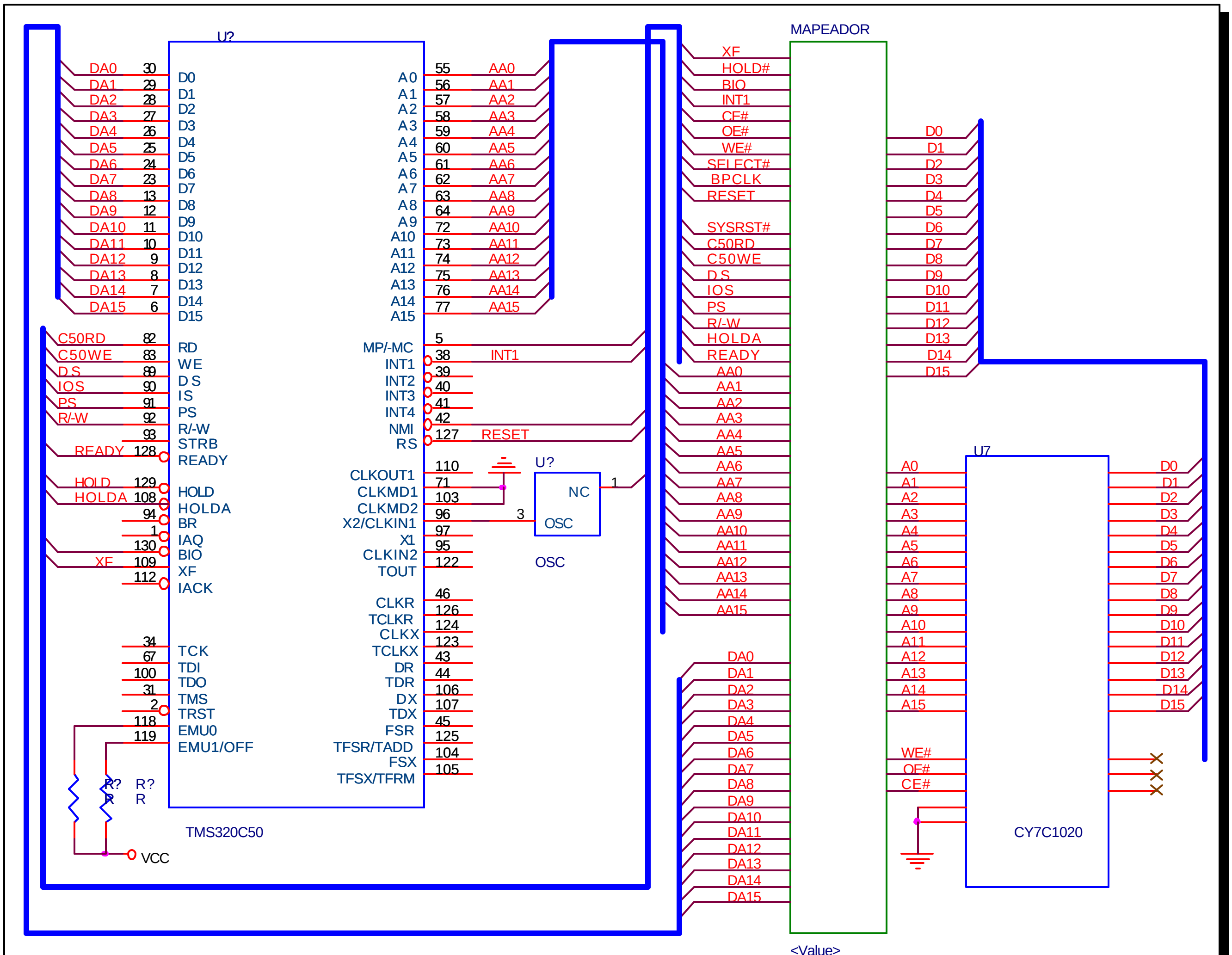

Figura 2. Esquemático de un nodo de procesamiento con su memoria mapeada.

se presenta como la mejor opción, en donde el eje vertical es la dirección mapeada, el eje horizontal es la dirección absoluta a mapear y la línea recta define al mapeo.

En el mapeo se permite que un procesador proyecte su dirección absoluta a una localidad de una memoria no inmediata a él, llamada memoria mapeada, cuyo nombramiento es de acuerdo al nombre de la dirección absoluta, lo cual genera el siguiente postulado:

Postulado 4: la dirección absoluta del procesador es la variable independiente *x* de la ecuación cartesiana y la dirección mapeada es la variable dependiente *y*.

Aplicando una vez más el tercer aspecto, pero ahora bajo la perspectiva de los cuatro postulados anteriores, se puede abstraer el direccionamiento al primer cuadrante del plano cartesiano, en donde la función continua es una línea recta con una pendiente de 45 grados desde el origen del plano. La **figura 3** muestra la abstracción del direccionamiento típico de un nodo de procesamiento, con la siguiente consideración: en el origen del plano, el cero, se encuentran las direcciones absolutas y mapeadas nombradas como las direcciones cero, ya que normalmente en los mapas de memoria se considera el cero como el primer nombramiento[1] de las direcciones (es inusual ver mapas de memoria en donde la primera dirección sea nombrada con 1 u otro valor diferente a cero).

La **figura 3** muestra el mapeo aparentemente imposible con el procesador cuyo mapa de direcciones va de 0 a 65535. A partir de aquí empezamos a considerar al direccionamiento como un ente más abstracto con el motivo de darle una demostración más rigurosa. Por tanto, se entrará en un nuevo nivel de abstracción al considerar un cuadrado en el plano cartesiano. Esta nueva etapa de razonamiento matemático busca demostrar que la comparación entre la variable independiente y el mapeo es irracional, y para ello se sigue la siguiente hipótesis[1]:

"Se dice que el lado y la diagonal de un cuadrado son, de hecho, inconmensurables"

A partir de este momento se entra en un razonamiento matemático que hace más tenue la conexión con el origen computacional del problema; por lo tanto, el nivel de abstracción es tal que el fenómeno real del mapeo ha dado paso a la comprobación de que la comparación entre el lado inferior horizontal del cuadrado y su diagonal es irracional. La **figura 4** muestra el cuadrado obtenido a partir de la **figura 3**.

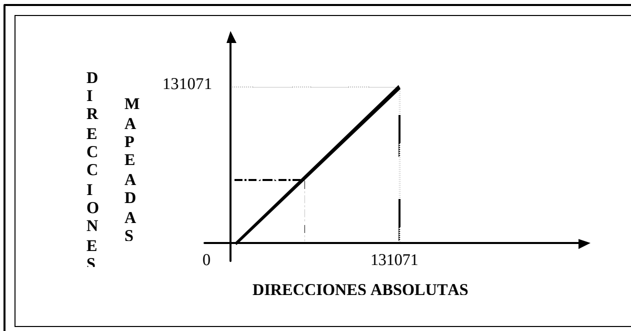

Figura 3. Plano cartesiano para el mapeo de memoria.

Si *a* es el lado y *b* la diagonal de un cuadrado, entonces según el teorema de Pitágoras:

$b^2=a^2+a^2=2a^2$,

y por tanto

$$\left(\frac{b}{a}\right)^2 = 2$$

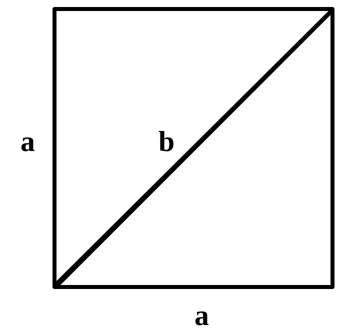

Figura 4. Cuadrado del Mapeo.

Se dice que no existe ninguna fracción tal que su cuadrado sea igual a 2. Bien, demostremos por contradicción, y sean *p* y *q* números enteros para los cuales

$$\left(\frac{p}{q}\right)^2 = 2$$

... donde se puede suponer que *p* y *q* no tienen factores comunes, porque en otro caso podríamos simplificar la fracción. Pero si $(p/q)^2=2$, entonces $p^2=2q^2$, y por tanto p^2 es divisible por 2. En este caso p^2 es también divisible por 4, puesto que es el cuadrado de un número par. Así $p^2=4q_1$; esto es, $2q^2=4q_1$, y $q^2=2q_1$. De esto se sigue que *q* debe también ser divisible por 2. Pero esto contradice la suposición de que *p* y *q* no tienen factores comunes. Esta contradicción demuestra que el cociente *b/a* no puede ser expresado mediante un número racional, esto es: *"La diagonal y el lado de un cuadrado son inconmensurables"*.

Esta demostración nos lleva a una etapa más de abstracción cuando se conecte con el Teorema de Cantor, pero antes cabe una conclusión parcial. Se concluye que la función del mapeo queda indeterminada por la exclusividad de las direcciones absolutas, ya que la ausencia de las direcciones mapeadas en los razonamientos matemáticos elimina la correspondencia biunívoca, y ésta en sí es la esencia del mapeo.

Del cuadrado de la **figura 4** de lado ***a*** y diagonal ***b***, parece evidente, considerando toda su área, que hay muchos más puntos tomando en cuenta los lados verticales y el horizontal superior que en el lado horizontal inferior. O también se podría decir que, como la magnitud de la diagonal ***b*** es mayor que la magnitud del lado ***a***, entonces hay más puntos en la diagonal que en cualquiera de los lados ***a***. Lo anterior nos lleva a un axioma heurístico: *"los puntos en la geometría euclídea son como las direcciones absolutas del direccionamiento de un procesador"*. Sin ellos los demás conceptos no tendrían existencia.

Para aclarar esto se va a demostrar que el lado inferior horizontal ***a*** tiene más puntos que la diagonal ***b*** del cuadrado. Básicamente se establece un procedimiento que haga corresponder un punto dado de la diagonal ***b*** con un punto del lado ***a***. Siempre y cuando a cada punto de la diagonal ***b*** corresponda uno y nada más que uno del lado ***a*** y nunca uno del lado ***a*** corresponda a más de uno de la diagonal ***b***. Es decir, se establece entre ambos conjuntos de puntos la correspondencia biunívoca.

Así, ante todo se necesita fijar una manera de localizar inequívocamente un punto cualquiera en el plano del cuadrado y el lado inferior ***a***. Por tanto, por el método cartesiano un punto ***P*** de la diagonal (el plano) se distingue de todos los demás por sus distancias a los lados ***a*** (para el método cartesiano los lados que se consideran son el vertical izquierdo y el horizontal inferior) del cuadrado. El lado vertical ***a*** se superpone al eje positivo de las ordenadas y el lado horizontal ***a*** se superpone al eje de las abscisas. Para simplificar las cosas se considera que los lados del cuadrado tienen como medida la unidad. De esta forma, todas las distancias (***x***, ***y***) medidas en el cuadrado serán inferiores o cuando más iguales a uno, y se expresarán como fracciones decimales.

A un punto del plano del cuadrado se le hace corresponder un punto del eje horizontal, el cual se obtiene mediante el cálculo del área del cuadrado delimitado por las coordenadas del punto del plano. Con este razonamiento se involucra a los lados que permiten la correspondencia biunívoca entre las ordenadas y las abscisas. De este modo, el punto se localizará mediante las siguientes expresiones:

$$y \; ? \; 0.0$$

$$x \; ? \; [0.a]^2$$

Para establecer una correspondencia entre un punto del plano, como (0.3, 0.3), con un punto en el eje de las abscisas se superpone el producto del área del cuadrado con el lado inferior horizontal ***a***, de tal modo que el producto 0 del área coincida en el origen del plano cartesiano y su crecimiento siga por el eje de las abscisas. Así, el punto (0.3, 0.3) se hace corresponder con el punto donde ***y*** = 0.0, y ***x*** = 0.09.

La **figura 5** muestra la correspondencia entre el punto (0.3, 0.3) del plano y el punto (0.09, 0.0).

La razón de emplear el área del cuadrado es poder involucrar intuitivamente dos conceptos matemáticos, a saber: la *correspondencia biunívoca* entre los ejes del plano cartesiano, y la *potencia*. En el anterior procedimiento la potencia existe en el momento del cálculo de área del cuadrado.

En forma general se puede decir que, para todo punto ***P*** de coordenadas (0.***a***, 0.***a***), existe un punto ***Q*** en el segmento de coordenadas ($[0.a]^2$, 0.0). Este punto está en el segmento, puesto que su ***x*** es mayor que 0 y menor que 1. Este procedimiento se

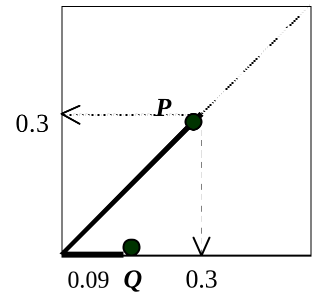

Figura 5. Correspondencia entre un punto de la diagonal y un punto del lado horizontal inferior del cuadrado.

conoce con el nombre de *Correspondencia de Cantor*, y el *Teorema*, que es también de Cantor, se enuncia como sigue:

"Los puntos del plano tienen la potencia del continuo"

Desde la perspectiva del direccionamiento de memoria lo anterior tiene el siguiente significado: de un modo u otro con las direcciones del mapa de memoria de un procesador se podría mapear tanta memoria como un factor conocido lo permitiese.

En el caso del diseño digital de un procesador el mapa de puertos es el factor multiplicativo más empleado. El método más usual es el usar comandos a través de un puerto o puertos. Haciendo una analogía con la Correspondencia de Cantor, se podría considerar que el comando del puerto tendría la misma función de las ordenadas: *compactar*. Con este método se compacta de un modo artificial el direccionamiento; así, un procesador con un mapa de 65536 podría llegar a mapear el orden de Gigas. Si el comando es de 16 bits de datos, entonces un puerto podría proponer 2^{16} combinaciones posibles que permitirían establecer 2^{16} mapas de memoria de 65536 direcciones, es decir, se podría tener un direccionamiento de hasta 4 Gigas.

La figura 6 muestra el diseño del nodo de procesamiento, en donde el procesador cuenta con un par de buses de 16 bits tanto para el área de datos y de direcciones; además tiene un puerto de 16 bits, y el banco de memoria lo constituyen 6 dispositivos SRAM de 64k x16. La entidad digital, por así decirlo, encargada de comprobar la Correspondencia de Cantor y de involucrar el factor para compactar el direccionamiento es un circuito programable: CPLD.

El listado 1 presenta el método codificado en VHDL que deberá implementarse para lograr la "compactación" del mapa de memoria original del procesador.

De este modo se puede decir que se ha comprobado un teorema computacional, el cual se enuncia diciendo:

"el método de comandos por puertos es un procedimiento adecuado de compactación del mapa de memoria de direcciones absolutas, y un eficiente mecanismo de expansión del mapa de memoria de direcciones mapeadas"

Con este Teorema se explica la razón de la compactación del direccionamiento de un procesador al momento del mapeo de grandes cantidades de memoria mapeada. En la actualidad existen otros métodos de compactación más complejos, tal y como lo es el *método de paginación* **[3] similar al usado por el kernel[3] de los sistemas operativos multiusuarios: UNIX, WINDOWS NT, etc.**

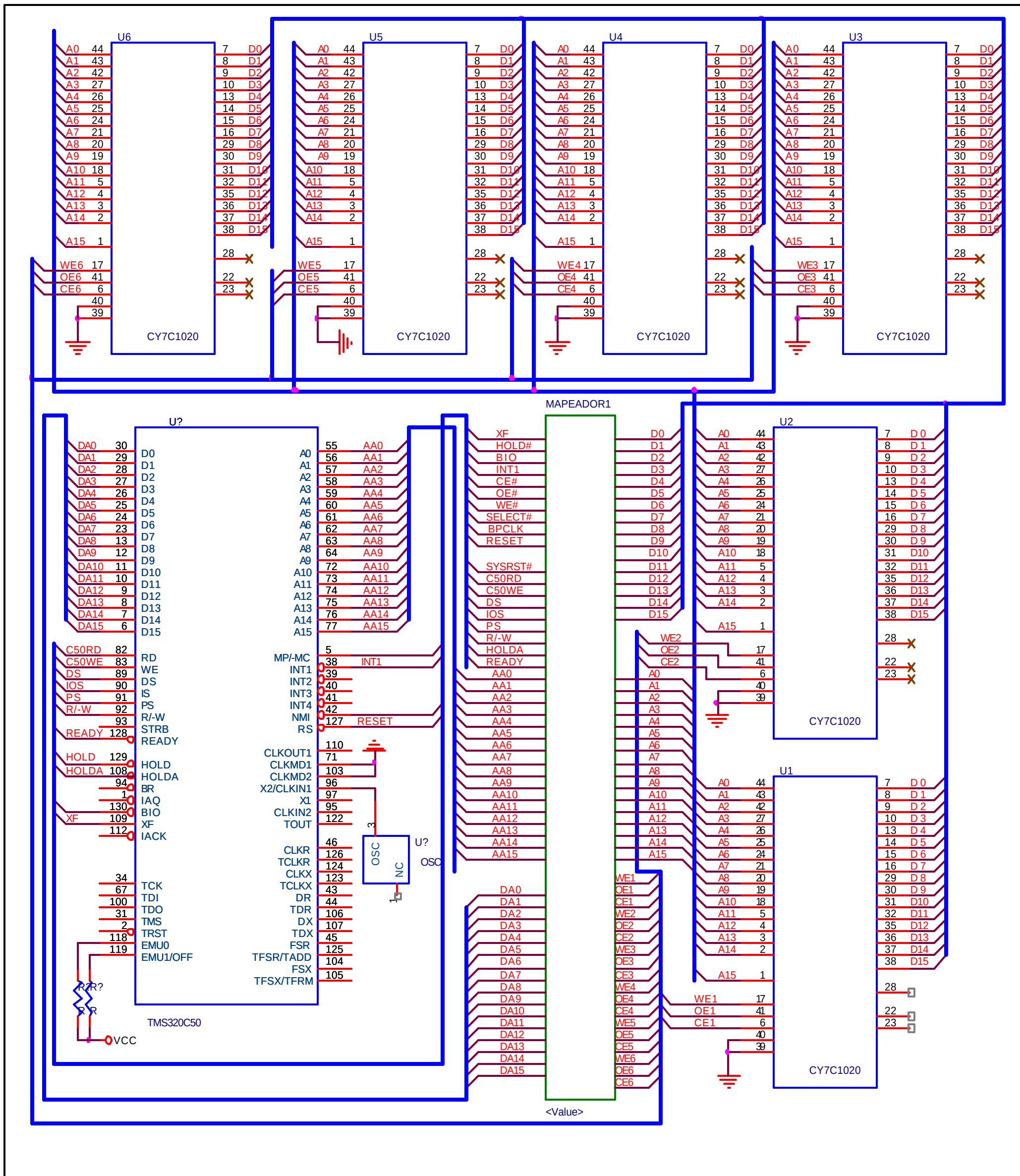

Figura 6. Esquemático del nodo de procesamiento con 6 dispositivos de memoria.

RESULTADOS

La investigación realizada en este trabajo proporciona un principio filosófico aplicado a partir de muchos, que permite entender la base con que se construyen los circuitos de hardware que proporcionan los diversos modos de direccionamiento en los bloques de *memoria virtual* **[3] de los sistemas operativos.**

CONCLUSIONES

Cuando se localiza el elemento básico de un hecho real, como lo es una dirección de un direccionamiento, en esa medida se debería poder asociar ese hecho como uno de los significados del Teorema de Cantor. De hacerlo así se tendría

```
—Progama en VHDL para el plantear el comportamiento del Mapeador.
—En la figura 6, se muestra el esquemático para el direccionamien-to
—del Mapeo.
library ieee;
use ieee.std_logic_1164.all;
use work.numeric_std.all;
entity mapeador is
port(
  —buses del procesador.
  AA:in std_logic_vector(15 downto 0);
  DA:inout std_logic_vector(15 downto 0);
  —control del procesador.
  C50RD: in std_logic;
  C50WE: in std_logic;
  —selección de área de datos(ds)
  — o de puertos(is) del procesador.
  DS: in std_logic;
  IOS: in std_logic;
  —señales activas en bajo.
  —señales de control del mapeador.
  —Chip Select.
  CE: out std_logic_vector(5 downto 0);
  —Output Enable.
  OE: out std_logic_vector(5 downto 0);
  —Write Enable.
  WE: out std_logic_vector(5 downto 0);
  —buses del mapeador.
  A:out std_logic_vector(15 downto 0);
  D:inout std_logic_vector(15 downto 0));
end mapeador;
architecture comportamiento of mapeador is
signal pto: integer range 0 to 6;
begin
p1:process(IOS)
begin
  case DA(2 downto 0) is
  —comandos para identificar la activación
  —de cada una de las memorias mapeadas.
  when «000» => pto <= 0;
    when «001» => pto <= 1;
    when «010» => pto <= 2;
    when «011» => pto <= 3;
    when «100» => pto <= 4;
    when «101» => pto <= 5;
    when others => pto <= 6;
  end case;
end process p1;
p2:process(DS)
begin
case pto is
  when 0 =>  CE(0)<='0';
    OE(0)<='0';
    WE(0)<='0';
    A(15 downto 0)<=AA(15 downto 0);
    if (C50WE='0') then
      D(15 downto 0)<=DA(15 downto 0);
    else
      DA(15 downto 0)<=D(15 downto 0);
    end if;
  when 1 =>  CE(1)<='0';
    OE(1)<='0';
    WE(1)<='0';
    A(15 downto 0)<=AA(15 downto 0);

    if (C50WE='0') then
      D(15 downto 0)<=DA(15 downto 0);
    else
      DA(15 downto 0)<=D(15 downto 0);
    end if;
  when 2 =>  CE(2)<='0';
    OE(2)<='0';
    WE(2)<='0';
    A(15 downto 0)<=AA(15 downto 0);
    if (C50WE='0') then
      D(15 downto 0)<=DA(15 downto 0);
    else
      DA(15 downto 0)<=D(15 downto 0);
    end if;
  when 3 =>  CE(3)<='0';
    OE(3)<='0';
    WE(3)<='0';
    A(15 downto 0)<=AA(15 downto 0);
    if (C50WE='0') then
      D(15 downto 0)<=DA(15 downto 0);
    else
      DA(15 downto 0)<=D(15 downto 0);
    end if;
  when 4 =>  CE(4)<='0';
    OE(4)<='0';
    WE(4)<='0';
    A(15 downto 0)<=AA(15 downto 0);
    if (C50WE='0') then
      D(15 downto 0)<=DA(15 downto 0);
    else
      DA(15 downto 0)<=D(15 downto 0);
    end if;
when 5 =>  CE(5)<='0';
    OE(5)<='0';
    WE(5)<='0';
    A(15 downto 0)<= AA(15 downto 0);
    if (C50WE='0') then
      D(15 downto 0)<=DA(15 downto 0);
    else
      DA(15 downto 0)<=D(15 downto 0);
    end if;
  when others =>
    A(15 downto 0)<=»ZZZZZZZZZZZZZZZZ»;
    D(15 downto 0)<=»ZZZZZZZZZZZZZZZZ»;
    CE(5 downto 0)<=»111111";
    OE(5 downto 0)<=»111111";
    WE(5 downto 0)<=»111111";
end case;
end process p2;
end comportamiento;
```

Listado 1. Programa en VHDL para el CPLD, que compacta el direccionamiento del procesador.

un pensamiento en términos de conceptos simbolizados, un pensamiento altamente especializado y funcional, tal y como se presentó en este trabajo. El único punto débil, si lo hay, es que esto se construye con la intuición y no basándose en razonamientos lógicos. Aquí los razonamientos lógicos se emplearon en una fase posterior, para intentar un cierto grado de acuerdo con el lector.

Además, parece que la *intuición* es el mecanismo de captación de un concepto, que junto con la familiarización y el *know how* de un campo permiten conocer ciertos significados especializados (los conceptos simbolizados), y al alinearlos a los problemas que se intentan solucionar, los solucionan, o por lo menos permiten obtener descripciones que funcionalmente aclaran el problema.

Ahora, de acuerdo con la investigación de este trabajo se podría concebir que el analfabetismo funcional se podría extrapolar patológicamente en las actividades teóricas de un pragmático, y así uno podría comprender el porqué ciertas personas aunque son diestros en manipular el álgebra, el cálculo y la geometría analítica sean incapaces de dar soluciones matemáticas a ciertos problemas de la ingeniería y de las ciencias, y en esa medida presentarían la inhabilidad para generar trabajos originales. Otro aspecto patente es que, ante la carencia de un significado especializado, se entra en una situación de inacción, y sólo se accesa a un gran potencial de acción cuando se logra dicho significado. Quizás este párrafo presente una de las razones de la productividad inconstante al realizar esta investigación.

Bibliografía

[1] Aleksandrov A. D., Kolmogorov A. N., y Laurentiev, M. A., "*La matemática: su contenido, métodos y significado*", Alianza Universidad, Alianza Editorial, Vol. 1, México, 1994, Cap. 1.

[2] Erro Luis Enrique, "*El Pensamiento Matemático Contemporáneo*", Instituto Politécnico Nacional, México, 1986, Cap. 1.

[3] Freedman Alan, "*Diccionario de Computación Bilingüe*", McGraw-Hill Interamericana, S.A., Colombia, 1996.