

La Técnica del Mapeo

*M. en C. G. Abraham Mas Levario
M. en C. M. Aurora Segura Corona
Ing. Héctor Cortés León
Profesores del Centro de Investigación
en Computación, IPN.*

El presente artículo plantea la técnica del mapeo como una de las soluciones para direccionar memoria ubicada en una tarjeta de expansión que se encuentra conectada en una de las ranuras de expansión de las computadoras personales, la cual es la anfitriona y responsable del mapeo.

CONCEPTUALIZACIÓN DEL MAPEO

Aquí se muestra la idea general de lo que es el mapeo de memoria, así como ciertas definiciones que permiten desarrollar los métodos matemáticos que cualifican la idea del mapeo.⁴

1.1 DESCRIPCIÓN GENERAL DEL MAPEO

La apariencia y el postulado del mapeo se muestran en la **figura 1-1**. La idea básica es hacer que cualquier acceso al segmento, esté reflejado por medio del mapeador en la memoria mapeada. La memoria mapeada podría estar constituida por ocho dispositivos 6164; la capacidad de estos dispositivos es de 8 Kbytes, que dan un total de 64 Kbytes. Para ver un poco más de cerca el mapeo, supóngase que desde la PC se está direccio-

nando el offset 0000 del segmento D, o sea en la dirección D000: 0000 del modo de operación real de la PC, y también supóngase que la PC no tiene memoria disponible físicamente en el segmento D, entonces, cuando desde un programa de la PC se escriba o se lea un dato en esa dirección (12), la memoria mapeada es direccionada por el mapeador para recibir o dar el dato; y cuando se da esta circunstancia es cuando existe el mapeo.

1.2 AXIOMAS

Se presenta una relación de ciertas observaciones que evidencian con más detalle el fenómeno del mapeo como cualquier hecho que puede ser percibido. Las observaciones son esencialmente los siguientes axiomas:

1. Cualquier idea o conjuntos de ideas pueden ser representadas con matemáticas con la finalidad de poder cualificar y cuantificar el objeto mental al que se refiere o refieren dichas ideas.
2. Cualquier objeto o conjunto de objetos computacionales pueden ser organizados con la cualifica-

ción y cuantificación del objeto mental al que se refieren tales ideas.

Corolario 1. En vista de lo anterior a cualquier objeto computacional se le puede atribuir cierto concepto cuando uno puede percibirlo.

3. A cualquier objeto computacional se le puede asignar un nombre según un criterio que tenga relación con la zona donde se aplique, por lo tanto a éste lo llamamos mapeador.
4. La primera acción creada en un mapeo es la comunicación entre el entorno de la PC y el mapeador.

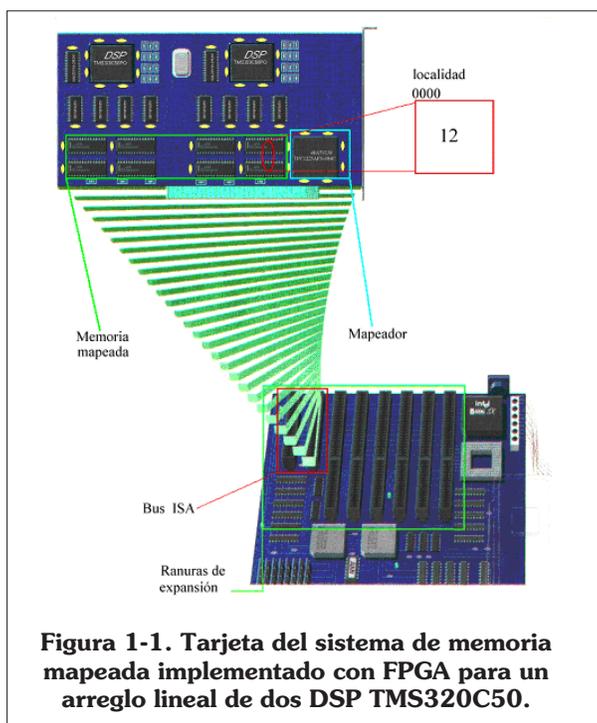


Figura 1-1. Tarjeta del sistema de memoria mapeada implementado con FPGA para un arreglo lineal de dos DSP TMS320C50.

5. La segunda acción creada en un mapeo es la comunicación entre el segmento D del entorno de la PC y la memoria mapeada.
6. El primer factor creado en una comunicación es el fenómeno de localización de espacios, llamado en esta tesis direccionamiento.

1.3 TEOREMAS QUE FUNDAMENTAN EL DISEÑO TEÓRICO DEL MAPEADOR

Teorema 1. Las líneas más significativas del direccionamiento del bus ISA son los factores que contribuyen a que el mapeador establezca la correspondencia unívoca entre el segmento D de memoria de la PC y la memoria mapeada.

Teorema 2. Las líneas menos significativas del direccionamiento del bus ISA son parte de la correspondencia unívoca entre el segmento D de memoria de la PC y la memoria mapeada.

1.4 PRIMER ACERCAMIENTO MATEMÁTICO AL MAPEO

El fenómeno del mapeo es esencialmente una circunstancia de localización por reflejación desde el punto de vista de la tarjeta coprocesadora, y además es una circunstancia de localización por correspondencia desde el punto de vista del mapeador quien genera la comunicación entre dos entidades completamente diferentes.

Bien, cuando tenemos el mapeo como una circunstancia de localización por correspondencia se está usando el método matemático llamado "correspondencia" y en base a su definición se puede iniciar el estudio del mapeo.

La definición de correspondencia en el campo de estudio de la teoría de

conjuntos, es: la asociación correspondiente de los elementos de un conjunto con los elementos de otro.

Entonces, de acuerdo con lo anterior supóngase que en el mapeo se distinguen dos conjuntos, uno es el conjunto llamado "segmento D" de la memoria en el modo real de la PC y el segundo es el conjunto llamado "memoria mapeada". Ambos conjuntos contienen 64 Kbytes de localidades. En donde, las localidades del segmento D son expresadas mediante los símbolos que van desde D0000:0000 hasta D0000:FFFF, y las localidades de la memoria mapeada desde 0000 hasta FFFF. La representación esquemática y correspondencia postulada de los dos conjuntos está en la **figura 1-2**.

La correspondencia que se da en el mapeo existe sólo para un par de elementos asociados en un determinado tiempo t_n , donde n es cualquier número mayor que cero. Además este tipo de correspondencia se llama "unívoca", porque todo elemento del

segmento D se relaciona con uno, y sólo uno del conjunto memoria mapeada, y de este modo la asociación es continua desde D000:0000→0000 hasta D000:FFFF→FFFF.

Con los resultados anteriores y con el concepto de "relación", se representa el mapeo en una forma matemática más compacta y manipulable para su estudio. Una relación se define como un conjunto de pares ordenados obtenidos por una correspondencia previamente establecida. Al observar la **figura 1-2** vemos que la correspondencia entre la localidad D000:0000 y la localidad 0000, representada por una flecha dirigida hacia la derecha, asocia un par de localidades indicadas como (D000:0000, 0000), que es un par ordenado. Esto se cumple con cada una de las asociaciones anteriores, de modo que la relación de los conjuntos "segmento D" y "memoria mapeada" se muestra en el **listado 1**.

El conjunto indicado es la simbolización del mapeo para todos los

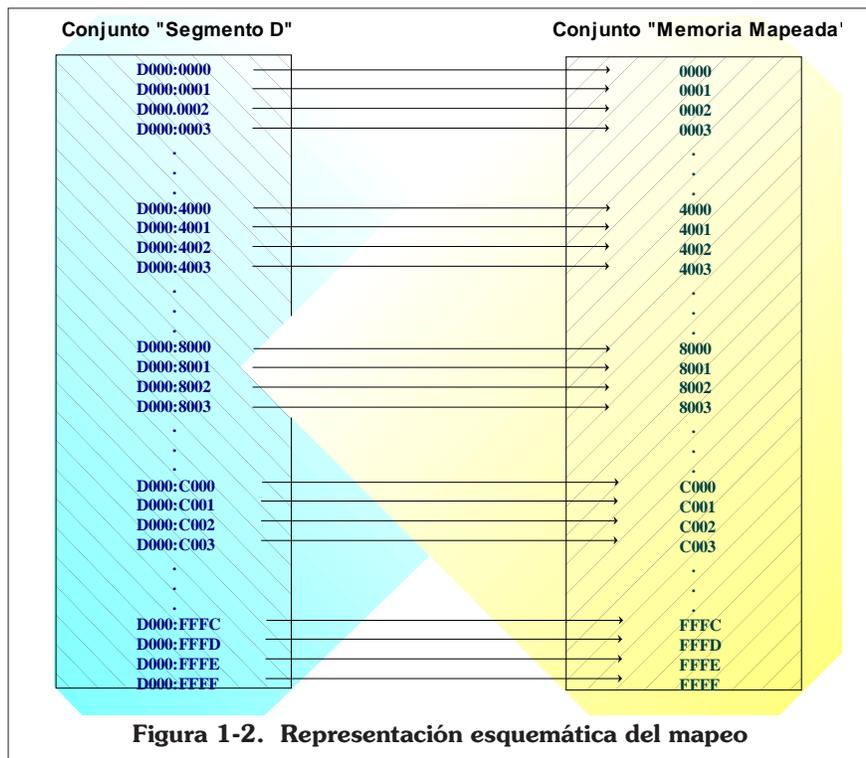


Figura 1-2. Representación esquemática del mapeo

{ (D000:0000, 0000), (D000:0001, 0001), (D000:0002, 0002), (D000:0003, 0003), ...
 (D000:4000, 4000), (D000:4001, 4001), (D000:4002, 4002), (D000:4003, 4003), ...
 (D000:8000, 8000), (D000:8001, 8001), (D000:8002, 8002), (D000:8003, 8003), ...
 (D000:C000, C000), (D000:C001, C001), (D000:C002, C002), (D000:C003, C003), ...
 (D000:FFFC, FFFC), (D000:FFFD, FFFD), (D000:FFFE, FFFE), (D000:FFFF, FFFF)
 }

Listado 1. Relación de los conjuntos “segmento D” y “memoria

instantes en donde él exista. Hasta este punto se ve que cada offset del segmento D corresponde exactamente a una localidad cuyo número es el mismo que el offset del segmento D.

Finalmente, tomando en consideración que el mapeo es un conjunto de pares ordenados de números, tales que ninguna pareja de ellos tiene el mismo primer número, y que los segundos números de las parejas pueden ser obtenidos mediante una igualdad, que se puede simbolizar a su vez como la ecuación de la función que se establecerá más adelante.

Antes de establecer la función es necesario representar las direcciones del segmento D como DNNNN en lugar de D000:NNNN, donde cada “N” es cualquiera de los siguientes números en el sistema hexadecimal: “0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F”. El motivo de lo anterior es que en el diseño digital es más común y práctico representar una dirección del bus de direcciones del modo real de las PC con cinco números hexadecimales para una posible manipulación algebraica.

De acuerdo a la observación de que la magnitud del offset del segmento D es igual a su correspondiente en la memoria mapeada, se puede generalizar a todos los pares ordenados del conjunto del mapeo como (DNNNN, NNNN) para deducir que el segundo número del par es el offset que se direcciona en el segmento D,

por lo tanto la ecuación de la función es:

$$f(\text{DNNNN}) = \text{DNNNN} - \text{D0000} \quad (1-1)$$

En donde NNNN es cualquier valor entre 0000h y FFFFh y el resultado de $f(\text{DNNNN})$ es el segundo número hexadecimal de cierto par ordenado de un mapeo. Cada una de las N de la función es un número hexadecimal y como tal puede ser representado con una notación binaria de cuatro dígitos mediante los respectivos valores de cuatro líneas de direcciones del bus, además supóngase que NNNN se puede representar como $N_{12}N_8N_4N_0$, en donde los subíndices de las N indican el subíndice de la línea de direcciones menos significativa al formar el número hexadecimal. Así que N_0 que es la que está más a la derecha se refiere al valor hexadecimal que forman el grupo de cada una de los magnitudes de las primeras cuatro líneas físicas del bus de direcciones, o sea, $N_0=(A_3A_2A_1A_0)$. Siguiendo lo anterior, el valor de la N_4 se obtiene con la siguiente igualdad: $N_4=(A_7A_6A_5A_4)$, por lo tanto N_8 y N_{12} se obtienen con las siguientes igualdades:

$$N_8=(A_{11}A_{10}A_9A_8),$$

$$N_{12}=(A_{15}A_{14}A_{13}A_{12}),$$

respectivamente.

La resta de la **ecuación 1-1** es la operación algebraica que establece el mapeo en sí. Esto quiere decir que el valor en el cual la PC está direccionando en el “segmento D” no se utiliza en la correspondencia unívoca, más bien, es el factor que crea la correspondencia unívoca y que es la condición para que exista el mapeo. Se concluye por la anterior observación que para hacer un direccionamiento en la memoria mapeada sólo se usan las direcciones desde la A_0 a la A_{15} del bus de direcciones del bus ISA. Ahora, se sabe que para direccionar memoria en el modo real desde el bus ISA se usan veinte líneas de direcciones que van desde la A_0 a la A_{19} y hasta este punto sólo se ha justificado de un modo claro el empleo de dieciséis líneas ($A_0, A_1, \dots, A_{14}, A_{15}$) en el mapeo. Entonces se postula que la resta de la **ecuación 1-1** es una aproximación y un modo implícito de decir que con las líneas A_{16}, A_{17}, A_{18} , y A_{19} se crea de algún modo el factor que contribuye al mapeo.

En las siguientes secciones se tratarán métodos complementarios a los anteriores para finalizar el estudio del mapeo.

1.5 EL MAPEADOR COMO PARTE DE UNA RED DE INTERCONEXIÓN

Se puede considerar a todo el conjunto de mapeadores de las posibles tarjetas de expansión que interactúan con el bus ISA como una red de interconexión dinámica. Una red de interconexión dinámica, desde el punto de vista de esta tesis, es una estructura física de direccionamiento por etapas; cada una de las etapas posee componentes básicos que se dedican a direccionar las memorias mapeadas de la tarjeta de expansión en el ambiente de las ranuras del bus ISA. Estos componentes son enlaces y conmutadores programables.

Si se considera que la PC es la única fuente de la red y que las memorias mapeadas son los destinos, entonces se puede ver que la red

está estructurada de tal manera que permite hacer una sola conexión de una fuente a un destino a la vez. Y si se supone que hay memoria mapeada en cada uno de los dieciséis segmentos de memoria de la PC, entonces la red tiene dieciséis posibles destinos. Como únicamente se cuenta con las veinte líneas ($A_0, A_1, \dots, A_{18}, A_{19}$) del bus de direcciones del bus ISA para establecer una conexión de mapeo, por lo tanto se puede deducir que las líneas A_{19}, A_{18}, A_{17} y A_{16} forman parte del mecanismo de direccionamiento de las etapas de la red debido a que tales líneas indican uno de los dieciséis segmentos, o sea, uno de los dieciséis destinos.

Para establecer las características de la red se postulan una serie de suposiciones que se enumeran abajo.

1. Se usa una sola línea de las cuatro anteriores ($A_{19}, A_{18}, A_{17}, A_{16}$) en cada etapa para direccionar gradualmente un destino.
2. Como solo entra una línea en cada etapa, entonces los conmutadores de cada etapa deberán tener una sola entrada de programación, y además cada una de estas entradas de programación deberán estar conectadas conjuntamente con la línea que llega a esa etapa.
3. El conmutador solo puede conmutar hacia uno de dos estados posibles (estado 0 o estado 1) debido a la naturaleza binaria de la línea que se conecta a su entrada de programación. Por lo tanto, el conmutador programable hace una unión de la entrada al estado 0 si la línea trae un valor de 0 o una unión de la entrada al estado 1 si la línea tiene un valor de 1.
4. Como la red direcciona gradualmente un destino y además tiene únicamente una entrada, se concluye que los conmutadores programables de cada etapa tienen únicamente una entrada.

Ahora, si se considera que se tiene una relación de números binarios, a partir de los valores que se puedan leer con las cuatro líneas ($A_{19}, A_{18}, A_{17}, A_{16}$) de un direccionamiento, en cada uno de los segmentos. El bit más significativo de ese cuarteto es A_{19} , en la relación mostrada en la **tabla 1-1**, se percibe que, en esta línea, los primeros ocho valores son cero y en los últimos ocho son uno. El número de A_{18} cuando A_{19} es cero, cambia de cero en los primeros cuatro valores a uno en los siguientes cuatro valores, y cuando A_{19} es uno, A_{18} cambia de cero nuevamente en los siguientes cuatro valores hasta que finalmente es uno en los últimos cuatro valores; de este modo también se puede distinguir cómo va variando A_{17} con respecto a A_{18} , y cómo va variando A_{16} en relación con A_{17} .

TABLA 1-1
Tabla de números binarios

A_{19}	A_{18}	A_{17}	A_{16}
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

Se puede considerar que con la anterior regla se está indicando, que el número de salidas de cada etapa es según el número de grupos de ceros o unos que originen las anteriores líneas de direcciones. Así que como A_{19} origina dos grupos (un grupo de ocho ceros y un grupo de ocho unos),

entonces la etapa que controla esta línea solo tiene dos salidas, y de ahí que esta etapa sólo tenga un conmutador programable. Se puede entonces suponer que la PC se conecta directamente a dicha etapa. La línea A_{18} origina cuatro grupos (dos grupos de cuatro ceros y dos grupos de cuatro unos), de modo que la etapa controlada por A_{18} tiene cuatro salidas, y para tener cuatro salidas se requiere dos conmutadores, por lo tanto la etapa controlada por A_{18} se conecta a las dos salidas de la etapa controlada por A_{19} . Finalmente, la etapa controlada por A_{17} se conecta a las cuatro salidas de la etapa controlada por A_{18} , y la etapa controlada por A_{16} se conecta a las ocho salidas de la etapa controlada por A_{17} . En la **figura 1-3** se muestra la estructura de la red de interconexión dinámica para el mapeo en cada uno de los segmentos.

Generalizando, la red consiste de $n = N / A$ etapas, donde N es el número de salidas y A es el número de líneas usadas para establecer una conexión entre la PC y la memoria mapeada. Como $N = 16$ y $A = 4$, entonces $n = 4$ etapas. Cada etapa en la red consiste de un cierto número de enlaces y de conmutadores, en donde el número de enlaces y el número de conmutadores es el mismo de acuerdo a la posición de la etapa en la red. La etapa 1 que está controlada por el bit más significativo (A_{19}) de la dirección del destino tiene la posición cero, la etapa 2 que está controlada por A_{18} tiene la posición uno, la que controla A_{17} es la etapa 3 y tiene la posición dos, y la que controla A_{16} es la etapa 4 y tiene la posición tres.

Cada etapa consiste de $m = 2^p$ enlaces y conmutadores, en donde p es el número de la posición de la etapa, en la red. Según lo visto anteriormente, la etapa 1 consta de un enlace y de un conmutador ($m = 2^0 = 1$), la etapa 2 consta de dos enlaces y

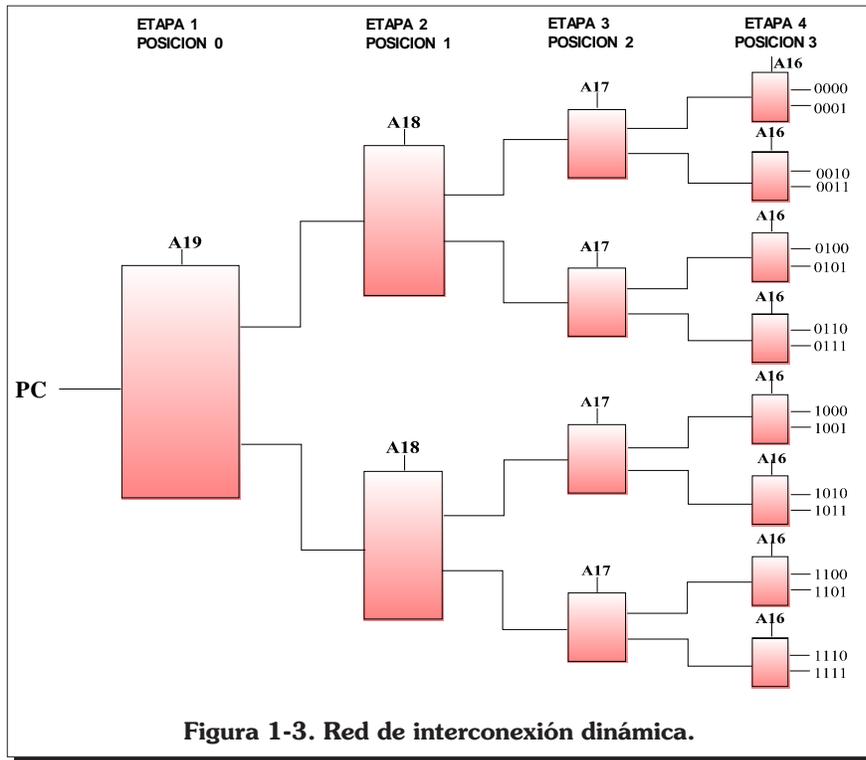


Figura 1-3. Red de interconexión dinámica.

Como la permutación es esencialmente un par ordenado se puede encontrar una regla de igualdad que establezca como determinar el segundo miembro a partir del primer miembro. Esta regla (para los fines de este estudio) puede ser representada como una función Booleana siguiendo la suposición de que si se está hablando de una conexión, implícitamente se está tratando con un espacio direccionado y según con lo ya visto se puede establecer que S denota la representación binaria de una fuente, de modo que se puede establecer que $S = A_{19}A_{18}A_{17}A_{16}$ indica con cual destino se hace la conexión. Ahora, para tener consistencia el destino D también debe denotar una representación binaria con cuatro números binarios. Así que la permutación que indica que existe mapeo en el segmento D ($A_{19}=1, A_{18}=1, A_{17}=0, A_{16}=1$) es:

$$[(1101, 1101)] \quad (1-2)$$

dos conmutadores ($m = 2^1 = 2$), la etapa 3 consta de cuatro enlaces y cuatro conmutadores ($m = 2^2 = 4$) y la etapa 4 consta de ocho enlaces y ocho conmutadores ($m = 2^3 = 8$).

ción en los sistemas digitales básicamente está integrado por expresiones Booleanas, las cuales determinan cómo debe operar el sistema según el estímulo recibido.

Si la función Booleana es $F(S)$ y si además suponemos, en base a la figura 1-3, que cada etapa de la red de una conexión es una función Booleana parcial de $F(S)$, entonces $F(S) = f_{19}f_{18}f_{17}f_{16}$. Estas funciones Booleanas parciales son básicamente funciones de conmutación; entonces, en general, cualquier permutación de una red puede ser representada en términos de un conjunto de funciones de conmutación. En vista de lo anterior, la determinación de $F(S)$ en base a la magnitud de la fuente S, se

Con lo que respecta a los enlaces de cada una de las etapas de la red, se aprecia que estos son buses de direcciones que están formados por las primeras dieciséis líneas de direcciones ($A_0, A_1, A_2, A_3, \dots, A_7, A_8, \dots, A_{12}, A_{13}, A_{14}, A_{15}$) del bus ISA debido a que las últimas cuatro ($A_{16}, A_{17}, A_{18}, A_{19}$) se emplean específicamente para controlar, a través de la red, el direccionamiento de una localidad de memoria mapeada.

A partir de la figura 1-3 y del concepto de permutación para una red de interconexión se obtendrá para el sistema una expresión Booleana. Una permutación se refiere a la conexión de un conjunto de fuentes a un conjunto de destinos tales que cada fuente es conectada a un destino simple; al observar este concepto se ve que es congruente con lo aplicado anteriormente en la correspondencia unívoca. La representación simbólica de una conexión en este caso es la representación de un par ordenado. En vista de lo anterior y recordando que la red solo activa a la vez un mapeador, la permutación en este caso es $[(S,D)]$ lo que significa que únicamente existe un mapeador en la red haciendo una conexión de una fuente "S" con un destino "D".

S				Conecta a	F(S)
A_{19}	A_{18}	A_{17}	A_{16}	\otimes	$f_{19} f_{18} f_{17} f_{16}$
1	1	0	1	\rightarrow	1 1 0 1

Tabla 1-2. Tabla de implementación de la función.

1.6 CONSTRUCCIÓN DEL SISTEMA DE PROGRAMACIÓN DEL MAPEADOR

En base a los estudios cuantitativos y cualitativos anteriores se concluye que el mapeador requiere un sistema para programar a los conmutadores. Un sistema de programa-

encuentra a partir de una tabla de verdad simple, como la siguiente: Cada una de las funciones de conmutación f_{19}, f_{18}, f_{17} , y f_{16} , por lo

tanto pueden ser expresadas como:

$$f_{19} = A_{19},$$

$$f_{18} = A_{18},$$

$$f_{17} = \overline{A_{17}},$$

$$f_{16} = A_{16}.$$

La función Booleana, entonces, es $F(S) = A_{19}A_{18}\overline{A_{17}}A_{16}$, lo que indica que para encontrar el destino se hace $D = F(S)$. Esta función Booleana se cumple cada vez que se desea mapear memoria en el segmento D, o sea, cuando la permutación existente en la red de interconexión de mapeo es [(1101, 1101)], así que el mapeador para el segmento D debe integrar en su sistema de programación la implementación de la función a la cual se le puede nombrar como una función de mapeo de la red de interconexión. En general, se puede decir que la es la función sintética de las dieciséis posibles funciones de la red de interconexión de mapeo. La función $F(S)$ además de indicar que destino se conecta con el fuente, indica a través de los valores de las líneas A_{19} , A_{18} , A_{17} y A_{16} cómo programar los conmutadores de cada etapa para direccionar gradualmente un destino.

1.7 CONSIDERACIONES PRÁCTICAS DEL MAPEADOR

Se puede suponer que el mapeador consiste de buses, dispositivos programables y dispositivos de programación. Los buses, además de estar constituidos por las primeras dieciséis líneas de direcciones del bus ISA, también contienen los datos del mismo bus el cual lleva en sí la información que direcciona las direcciones en la memoria mapeada.

Los dispositivos programables son los buffers que controlan la posibilidad de conexión de los buses a la

memoria mapeada. Estos pueden ser implementados a través de circuitos TTL, como el dispositivo 74244 para la parte de direcciones y 74573 para la parte de datos de los buses, u otra posibilidad es la implementación de buffers en una FPGA.

Se puede decir que los dispositivos de programación son los circuitos que reciben tanto las líneas de control del bus ISA como las líneas de direcciones que generarán posteriormente la función de mapeo, y dependiendo de cómo se implemente la función de mapeo será la selección del dispositivo programable. La implementación puede llevarse a cabo a través de una ecuación Booleana en un dispositivo PLD tal como una GAL, o a través de una función programada en un dispositivo FPGA.

Si se pone de nuevo atención sobre la anatomía de la memoria mapeada (que se dio al inicio de este capítulo), podemos ver que la idealización de contar con un dispositivo físico de memoria con una capacidad de 64 Kbytes lineales realmente se obtuvo de la abstracción del conjunto de ocho dispositivos SRAM 6164 de 20 ns, lo que facilitó el estudio del mapeo. Ya que este dispositivo es de 8k X 8 bits posee 13 líneas de direcciones para direccionar una localidad entre 8192 disponibles, y además se postula usar la línea A_0 del bus de direcciones del bus ISA como un control para direccionar palabras de datos de 16 bits en lugar de palabras de 8 bits, entonces se aplicarán respectivamente las líneas A_1 , A_2 , hasta A_{13} de direcciones del bus ISA a las líneas A_0 hasta A_{12} del bus del mapeador; a su vez, estas últimas líneas se aplican a las entradas A_0 hasta A_{12} respectivamente de los dispositivos 6164.

En vista de lo anterior se tienen realmente cuatro bloques de memoria mapeada en lugar de uno, así que

las líneas A_{14} y A_{15} se añaden a la función de mapeo, por lo tanto la función es esencialmente el mecanismo para decodificar la memoria mapeada, ya que dependiendo de los valores de las líneas A_{14} y A_{15} se mapea uno de los cuatro bloques de memoria.

Posteriormente se presentará la implementación del mapeador con dispositivos TTL y dispositivos GAL, y también se da la alternativa para implementarlo con la tecnología de FPGA.

Hasta este momento la función de mapeo no está totalmente construida, tiene los elementos básicos para contribuir al mapeo, pero por sí mismos no garantizan que la propuesta se logre adecuadamente.

CONCLUSIONES

La formulación de los postulados condicionaron la definición del mapeo, así como el conocimiento de los axiomas de la ingeniería de cómputo, razón por la cual se presentó una descripción cualitativa y cuantitativa del mapeo en la memoria de una computadora personal (PC).

Posteriormente con los postulados se obtendrá la ecuación matemática del mapeo, cuyas raíces algebraicas permitirán establecer el procedimiento para demostrar el mapeo bajo la perspectiva del teorema de Cantor.

Finalmente, el procedimiento se extrapolará a una estructura lógica que se diseñará con los principios básicos del diseño de sistemas digitales. Esto podría contribuir a la formación del programa de Doctorado de Ingeniería de Cómputo que el Centro de Investigación en Computación tiene como uno de sus objetivos.