Ej. 63

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE CIENCIAS

T E S I S

DE LICENCIATURA DE

F I S I C A

EL CONCEPTO DE PROCESADORES CONTROLADOS POR SUS DATOS Y SU APLICACION A LA FISICA DE ALTAS ENERGIAS

ARTURO ZENTELLA DEHESA

1983

Universidad Nacional Autónoma de México
UNAM
Biblioteca Central
Dirección General de Bibliotecas de la UNAM

UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

TESIS CON FALLA DE ORIGEN

I N D I C E

INTRODUCCION 1

CAPITULO I 4

CAPITULO II 10

CAPITULO III 18

CAPITULO IV 33

BIBLIOGRAFIA 50

[illegible]

Una computadora es un [illegible] de realizar operaciones aritméticas [illegible] decisiones a de interés [illegible] más acción el programa. De [illegible] procesarse [illegible] (tras computadores) [illegible] [illegible] cada un [illegible]

Las operaciones con [illegible] de valores [illegible] [illegible] dora [illegible] [illegible] todas las operaciones [illegible] de ser procesadas [illegible] [illegible] [illegible]

La interpretación [illegible] y que las [illegible] [illegible] en una tarea coherente [illegible] [illegible] [illegible] de un procesador [illegible] [illegible] [illegible] [illegible] que los controla [illegible].

El registro llamado contador de programa [illegible] de una instrucción. Después [illegible] [illegible] [illegible] la unidad de control [illegible] [illegible] [illegible] direcciones que [illegible] en los [illegible] [illegible] [illegible] [illegible]

tos necesarios para [illegible] un procesador en este [illegible] de procesadores de las computadoras [illegible] cuarto de siglo [illegible] años. Actualmente existen [illegible] de [illegible] de gadgets en el mercado.

Nunca en la primera [illegible] estuvo el [illegible] miento de las componentes [illegible] nario para su producción [illegible] usado más extensamente el silicio [illegible] al germanio son también en la [illegible] ciles de producción basadas en el silicio son [illegible] sado un solo cristal [illegible] de diferentes [illegible]

Esta tecnología [illegible] la inserción de [illegible] primero un cristal de silicio puro [illegible] de impurezas homogéneamente [illegible] impurezas posee su propiedad [illegible] del cristal de silicio. Esto es [illegible] temente dependiente de la temperatura. [illegible] pocos importantes: la de difusión [illegible] te de impurezas y la de difusión [illegible] de impurezas.

Mediante [illegible] la compensación y con ello [illegible] das complejas. En la práctica [illegible] pos de zonas situadas en la [illegible] debido a sobrestructuras en la concentración de [illegible] cuerpo en la superficie [illegible]

Finalmente la creación de capas [illegible] silicio (SiO2) y ventajas a [illegible] mite la posibilidad de [illegible] tensiones en el cristal entre [illegible] diversos componentes del circuito [illegible] lante que se aprovecha para la [illegible] ser o como capa protectora sobre el circuito [illegible]

En resumen, la técnica consiste en [illegible] control de crecer, difusión, resinación, [illegible] y metalización (soldadura de alambres con las [illegible] correspondientes dentro del chip para [illegible] nicación con el exterior [illegible]

Mediante la técnica [illegible] un cristal de silicio de [illegible] componentes electrónicos [illegible] integración de las [illegible] diante esta técnica se [illegible] hoy en día la integración [illegible]

CAPITULO II

Las Memorias

La memoria de la computadora [illegible] lo que está almacenado de manera [illegible] general. Los programas son tanto [illegible] usuarios como los [illegible] inherente a la máquina provistos por el fabricante [illegible] herramientas del sistema [illegible] ya dicho con anterioridad [illegible] ido datos son tanto las entradas [illegible] cantidades intermediarias [illegible] de algún problema [illegible] las rutinas de mantenimiento del sistema [illegible] den ser cantidades [illegible] suma parcial o la cuenta de [illegible] determinada fuente.

El concepto de diversificación de las necesidades con respecto a la memoria [illegible] una extensa variedad de [illegible] en el mercado.

Memorias Rápidas y Lentas

La experiencia ha demostrado que no siempre es deseable que la totalidad de la memoria contenida en la autómata esté basada en elementos de alta velocidad [illegible] secuentemente de alto costo. [illegible] de los procesadores en su relación con la velocidad de [illegible] queda de instrucciones en una gran cantidad de sitios [illegible] nes aproximadamente un 90% de la [illegible] en cualquier momento durante [illegible] contenidas en unos cuantos kilobytes (1024 localidades de memoria de longitud [illegible] 1 byte). Con esto queda [illegible] realizar la totalidad de la memoria [illegible] de acceso a ella no necesariamente [illegible]

La clase de memoria de mayor [illegible] cada al almacenamiento masivo de información [illegible] un archivo general. A esta clase pertenecen memorias [illegible] cánicas como cintas magnéticas o unidades [illegible] cintas almacenan la información en forma secuencial [illegible] que el disco funciona según el mismo [illegible] das de lectura y escritura pueden estar [illegible] do una búsqueda en dos dimensiones y [illegible] te el tiempo de acceso [illegible]

[illegible]

[illegible] de Memorias de [illegible]

[illegible] lectura de contenidos [illegible] o enfoques de los [illegible] el contrario de la [illegible] lidad ideas es [illegible] ras han producido en base [illegible] tención a las capacidades de las memorias. [illegible] una nueva clasificación de las memorias [illegible] con respecto hacia [illegible] tenidos en ellas, [illegible]

Las memorias de [illegible] dades y también fuera [illegible] ahora un procesador de los [illegible] con memorias que [illegible] discos o cintas [illegible] a esta clase [illegible] das tecnológicas actuales una RAM con capacidad [illegible] el orden de los [illegible] Bytes [illegible] poder resolver problemas [illegible] berlo memorias de varios [illegible] ocasiones insuficientes [illegible]

Una posible solución [illegible] ria RAM para datos activos [illegible] coria se completa con unidades [illegible] llamas virtual porque el procesador [illegible] cuella de la que en realidad [illegible] técnica se vale de un necesario [illegible] moria y de una secuenciación de la [illegible] de la computadora en unidades [illegible] zana generalmente hasta 512 bytes [illegible] desp. Las direcciones correspondientes [illegible] rios de 8 bytes. Por otro lado [illegible] procesador directamente [illegible] El contribuyen de memoria [illegible] una localidad [illegible] tivo de información. [illegible] desde de memoria [illegible] cuidadosamente [illegible]

[illegible]

[illegible] El [illegible] consiste en [illegible] el resultado [illegible] de [illegible] al [illegible] [illegible] de [illegible] [illegible] en [illegible] [illegible] un vector con [illegible] [illegible] [illegible] el [illegible] vector de [illegible].

Para [illegible] [illegible] de [illegible] [illegible] [illegible]

CAPÍTULO III

Procesadores de Periferia

La mayoría de los problemas que se presentan los usuarios de la línea de cómputo en la actualidad pueden ser resueltos por las computadoras de propósito general. Sin embargo, existen un gran número de problemas para los cuales estas resultan totalmente inadecuadas. Con este objetivo se han desarrollado sistemas especializados diseñados específicamente para ciertos problemas.

Las necesidades tan diferentes tienen la cualidad en común de ser muy específicas y demasiado difíciles de estructurado. Esta clase de problemas requieren el uso de procesadores de uso específico. Muchos de estos están ya en los mercados, en forma de subsistemas o sistemas multiprocesadores. Desde el punto de vista de la arquitectura de computadoras, el problema de la adecuación de procesamiento digital puede ser visto en la una combinación de dos maneras: el flujo desde datos hacia el flujo de instrucciones (flujo de datos) y el flujo de instrucciones (flujo de instrucciones).

Ambos flujos pueden ser de un elemento o de un conjunto de ellos. Se tiene así la siguiente tabla:

	Flujo de un dato	Flujo múltiple de datos
Flujo de una instrucción a la vez	Procesadores convencionales	Procesadores en paralelo y asociativos
Flujo múltiple de instrucciones	Procesadores "pipeline" (1)	Procesadores múltiples

En este contexto se tomen del inglés las siglas SIMILD. Con S se designa un flujo de un solo elemento a la vez, acompañado con el flujo instrucciones como D con datos. Para flujo múltiple se usa la M también con el calificativo adecuado. "I" o "D". Un sistema procesador de von Neumann ese basa la ejecución de las instrucciones

(1) Pipeline: Procesador en que el orden de ejecución de instrucciones se lleva a cabo antes de ejecutar la ejecución de la instrucción anterior.

[illegible] cada superficie a la interconexión [illegible] nal al chip dentro de un intervalo de [illegible]

No obstante, en la actualidad se [illegible] rar esta situación.

Dentro de la categoría de los SIMD [illegible] los procesadores asociativos. Los [illegible] meros están compuestos de células que en general son [illegible] cionalmente complejas y [illegible] tras elementos estructuras. Existe entre ellas [illegible] nida. Entre las células [illegible] trol, lo cual controla a los elementos procesadores una dicotomía entre el flujo de [illegible] gramas y los datos. No [illegible] cuerdo los microprocesadores que se [illegible] monos a fin de [illegible] dad de control central. Las [illegible] son similares unas a otras [illegible] almacenamiento, [illegible] las células. Finalmente las [illegible] tema se a través de la unidad de control [illegible] a un elemento de procesamiento. Después de que el elemento Procesador ha recibido una instrucción del control, ésta puede ser ejecutada o bien [illegible] el elemento procesador no puede realizar una actividad distinta. Por lo anterior es posible caracterizar [illegible] do momento a cualquier célula mediante los [illegible] números de célula, activa o inactiva. El procesador tema del siguiente capítulo tiene entre otras características la de optimizar el [illegible] la relación entre células activas e inactivas.

Los Procesadores asociativos son SIMD en los [illegible] les el medio de almacenamiento primario es una memoria asociativa. Generalmente las células de un procesador asociativo tienen escasa relación temática (trabajan casi independientemente) y son funcionalmente muy simples. El tipo de procesador que aquí diseñado generalmente utiliza de una memoria asociativa. Esto quiere decir que en un (Procesador asociativo) a diferencia de un PP (procesador en paralelo) las celdas que son alineadas o procesadas

(2) La electrónica que se emplea para desarrollar el sistema SIMD de alta velocidad deriva una [illegible] dad de calor. Esto impone un límite a la tecnología LSI.

Hardware Processor

Hardware processor es un procesador en modelo controlado por los datos, es el procesador que K lo considera como puesto a la tarea en [illegible].

Una computadora convencional [illegible] que [illegible] mas almacenados realiza la tarea de la [illegible] una secuencia ordenada de operaciones las cuales son ejecutadas sobre un conjunto de datos. Un procesador controlado por los datos (DDP = Data Driven Processor) está basado en el principio de que cualquier operación puede llevarse a cabo si están disponibles los operandos y si el operador receptor está en condiciones de aceptarlos.

Una realización de este modelo resulta sencilla si se procede en forma modular. Se debe construir un conjunto de operadores útiles que sean interconectables de manera versátil. La especialización de los módulos en estructura útiles computacionalmente aumenta dificultad para de problemas específicos. En vez de [illegible] no se requiere de un control central encargado por la rotación [illegible] sincronía sino en [illegible] distribuidas dedicadas al mantenimiento. Los datos [illegible] están en bloques [illegible] etiquetados por los operadores mismos. De ser [illegible] ser construidas estructuras muy complejas involucrando [illegible] de iteración decisiones o análisis exhaustivos [illegible].

Para que los datos controlen su propia transferencia entre los módulos resulta conveniente que la palabra que fluye por el procesador tenga contenidos en su estructura dos campos (fields): el campo de datos propiamente dicho y el de control. En el presente caso el campo de datos (X) consta de 16 bits y el de control se compone de 8. Estos últimos bits se encuentran a su vez estructurados de la siguiente manera: Como distintivo para un grupo de datos son usados 4 bits y se les llama campo N (name). Los restantes bits son C (complete), V (valid), H (hold) y B (block reset). V indica si el contenido del campo X es un dato útil. C marca su último miembro de un conjunto de datos definiendo así un bloque. H y B viajan en sentido contrario al flujo del campo X y sirven para optimizar la relación de módulos productivos contra los inactivos, maximizando la velocidad del flujo de datos sin que se pierda información ni tiempo. H se genera cuando un módulo no ha terminado de procesar la palabra en cuestión siendo [illegible] de [illegible]

Por primera ahora, una [illegible] compuesta de cerrojo se [illegible] no se refiere a [illegible] sino a la conducta de cerrojo [illegible] con la [illegible] registro de salida durante un ciclo más. Por ello se establece el /dolchón/ de un ciclo. Es decir [illegible] entre los registros de salida y de entrada.

Dos Tipos de Módulos

Como ya se mencionó anteriormente se desea [illegible] módulos según los encargados de efectuar las operaciones que se requieran. Y la [illegible] que no se inmediata [illegible] es decidir cuáles son las operaciones que se requieren. Originalmente deben ser las operaciones básicas a partir de las cuales debe de poderse construir una estructura adecuada para la mayoría de las tareas computacionales. Las operaciones de interés son aritméticas, lógicas y operaciones con memoria. Aunado a [illegible] de que los módulos sean de tipo general, estamos también en la libertad de crear módulos distintos que realicen actividades similares o incluso uno que sustituya un arreglo de una colección de otros, si con ello la eficiencia del proceso de esa actividad.

Hasta ahora se han construido 14 módulos que se consideran útiles para resolver problemas de reconocimiento de trayectorias (problema típico de la física de partículas). Los módulos están todos en la tarjeta [illegible] y trabajan con un reloj de una frecuencia de 40MHz.

Se han previsto dos tipos de módulos: del tipo bits en paralelo, palabras en serie y de bits en serie - palabras en paralelo. El primer tipo de módulos tiene una longitud de palabra fija (32 bits) y transfieren una palabra en un ciclo. Este tipo de módulos está diseñado específicamente para realizar operaciones individuales sobre muchos datos. Son empleados por tanto para la solución del problema de reconocimiento de patrones.

El segundo tipo de módulos tiene la particularidad de que la longitud de palabra es variable y puede procesar ocho palabras en paralelo (semiconcatenadas). Si la longitud de palabra es 1 entonces se requiere de 1 ciclo de reloj para la transferencia completa de las palabras. Estos módulos son empleados para realizar operaciones

módulos que N se sumia a un valor [illegible] se escribe en la memoria y se [illegible] un índice. [illegible] cursar los datos de la lista al módulo selector un [illegible] se contando elementos de un conjunto de datos por [illegible] el puerto de lectura se [illegible] aquellos datos [illegible] dice de lectura ya [illegible] con un puerto [illegible]

El conjunto de salida es un subconjunto de los datos entraron antes por el registro de escritura. Y [illegible] de a una selección hecha desde una selección a que [illegible] son sometidos los elementos del grupo original. Se pone que todos los elementos que pasan través del [illegible] to de escritura del módulo son examinados por una [illegible] y son retroalimentados posteriormente a la Lista/[illegible] vía el puerto de lectura. Durante el proceso de [illegible] el número es cambiado en aquellos datos que pasan [illegible] Ba.

3) Buffer. Este módulo es un FIF[illegible] Está pensado para resolver las necesidades de alm[illegible] miento temporal entre los datos que se difunden en [illegible] más de sufrir procesos independientemente son que [illegible] reunir en un mismo camino. Las palabras pueden [illegible] tos a leídas durante el mismo ciclo de reloj (25MHz) [illegible] fre hay datos disponibles a la salida de Buffer mient[illegible] éste no se encuentre vacío. Los M no pueden ser escr[illegible] si el ritmo promedio de entrada no excede el de salid[illegible] si la memoria dinámica de 256 localidades no ha sido [illegible] turada.

4) Mapa. Este módulo es una [illegible] ría asociativa. Se le ha asignado a cada posible valor que puede entrar por el puerto de escritura una célula de memoria. Todas las celdas se inicializan con cero. El proceso de escritura consiste en recuperar por dirección nuevos valores vecinos. Una forma particular del módulo Mapa consiste en una recuperación de 16 datos vecinos. A cambio del mayor número de datos recuperados esta opción requiere de dos ciclos de lectura. La parte truncada del valor para lectura es usada al puerto de salida y conocemos el valor de los índices. Los datos son escritos como una secuencia ordenada y se usa una [illegible] en el puerto de lectura para evitar dos lecturas hasta que un valor de escritura exceda al valor en el puerto de lectura. En el caso de secuencias no ordenadas, la M debe mantenerse hasta que sea transferida una M al puerto de escritura. Cuando ambos C han sido recibidos en la

etc). En la construcción de este módulo se utilizó un ALU en un chip de la serie MECL 10000 (Motorola). Un parche permite conectar convenientemente los bits de los canales XrN a las dos entradas de la ALU como función del control del ACU. El línea [illegible] de dos de las líneas de entrada capacita la operación del dato de salida. A la vez se permite el intercambio de nuevos datos a los registros de entrada. Todo se hace mediante el control de la H de entrada y de las H internas. Datos también alineados en las entradas cuando una C de salida.

9) Normalizador: Si un multiplicador es usado para aplicar una función lineal sobre un valor de entrada de X. Las memorias internas como las de 2147 directamente pueden ser previamente cargadas con valores de 12 bits de precisión. Un parche permite que los bits de los 20 bits pertenecientes a los canales XrN puedan ser conectados según las necesidades a las direcciones de 8 bits de las dos tablas de valores. Los contenidos de las localidades direccionadas son sumados a un valor con una precisión de 12 bits. Si se desea normalizar un dato de 12 bits de longitud lo conveniente es conectar los 8 bits más significativos a las líneas de direcciones del bloque superior de memorias y los 8 menos significativos al bloque inferior. Para valores menos largos pueden ser utilizados bits del canal de H sea ser conectados en paralelo a las dos tablas. Como resultado se tiene normalizadas de subcanales de X. Multiplicadas en producto constituye función del tipo F(X1) . G(X2), siendo X1 y X2 dos subcanales de X. Las H pasan a través del normalizador el producto H.

10) Tabla binaria (X): Este módulo sirve para generar una función arbitraria F(X1,X2) de los canales de entrada. Existe un parche para cada una de las entradas o mediante ellos son seleccionadas 3 líneas para direcciones en cada caso. Los bits de las restantes también intervienen en el parche. El valor de salida consta de 16 bits. Bajo el control de las H los datos se alimentan a las entradas. Si se conectan erróneamente los bits de la función al nombre de salida, el operador tiene que ser desplazado como una máscara sobre los datos.

11) Tabla unitaria: Es análoga a la tabla binaria excepto que ésta desarrolla una sola entrada.

D) Relaciones:

Los módulos siguientes están disponibles para

CAPITULO IV

Programa de Simulación

El procesador en paralelo descrito anteriormente requiere de un mecanismo para controlar el funcionamiento de cada uno de los módulos tipo por la cual fluyen. Este funcionamiento se refiere tanto a las condiciones deseables como a las condiciones imprevistas que pueden surgir con un conjunto de datos reales. Un mecanismo que lleve a cabo esta labor es mediante la adaptación a cada módulo de un FIFO circular (ring buffer). Inicialmente con ceros valores contenidos dentro de los FIFOs. Las entradas del módulo son conectadas a la salida de los FIFOs respectivamente. La salida del módulo se conecta a la entrada de otro FIFO. Si ahora se desea correr el programa los datos en los FIFOs de entrada, entrarán al módulo con procesados y los resultados son escritos en el FIFO de salida. Intercalando los valores en este último FIFO al final de la corrida y comparándolos con los valores esperados, es posible decir si falla en el funcionamiento deseado del módulo. La forma de correr los módulos se hace aquí a nivel de "hardware".

Una elección alternativa es correr los módulos a un nivel de "software". En este caso la actividad llama simulación, se realiza con la ayuda de una computadora convencional mediante un programa escrito en lenguaje de alto nivel. Este método es el que se discute en la presente tesis. Para la simulación que a cual se presenta se utiliza una computadora DIGITAL del tipo VAX 11-780 y el lenguaje es FORTRAN 77.

La simulación del procesador está organizada en rutinas que representan a cada uno de los módulos en forma independiente a un programa principal que simula la forma en que estos módulos están interconectados en la realidad. Lo anterior se hace a través de llamar las rutinas en el orden apropiado. El programa principal así descrito está encargado de pasar los resultados de salida de cada módulo a las entradas de los módulos correspondientes según la arquitectura de módulos que se está simulando. También está encargado de mantener una lista actualizada de los estados de cada uno de los registros de todos los módulos participantes en la simulación. Además esta lista incluye el número de ciclos que ha sido realizados.

[illegible] una terminal de pantalla y de impresora. Mediante un [illegible] serie adecuada de respuestas a las preguntas que va [illegible] dando Tmsp se puede decidir el modo en que [illegible] a realizar todas las [illegible] de inicialización del [illegible] (configuración interna del hardware, [illegible] (valores de los cables antes de cada [illegible]). Los valores [illegible] valores antes de cada ciclo en el caso de los cables [illegible] lectura o escritura [illegible] de los cables [illegible] el caso del cable de [illegible] cada [illegible] las señales [illegible]. Tmsp también permite [illegible] que del microprocesador final de cada [illegible]. Esto le puede ser de distintas maneras según la velocidad [illegible] deseada. Puede ser escrito en la pantalla [illegible] de salida definida por el usuario Tmsp. En el caso [illegible] terminales de impresión puede resultar un [illegible] incluso en un dispositivo de [illegible] para todas las [illegible] o preguntas. Esto último es más que conocido [illegible] uso cotidiano de Tmsp. El programa de diagnóstico [illegible] contemplada esta necesidad de ahorrar memoria [illegible] sible instruirla para que suprima algunas o todas las [illegible] puntos y remite el análisis de resultados necesarios.

A continuación se presenta un diagrama de [illegible] que corresponde a Tmsp.

Subrutinas Dentro de Tmsp

La estructura interna de Tmsp está organizada [illegible] secciones que han sido elaboradas a la manera de subrutinas, las cuales son Inc, Rdb, Inc, Exit, Outhold [illegible].

Inc tiene secciones inicializadoras individuales para cada módulo independientemente. Estas secciones son seleccionables mediante una instrucción GOTO con [illegible] de dirección (número) GOTO [illegible]. Es mediante la subrutina Inc que se establecen los valores clave para los nombres (name keys) y el número de los cables que serán usados para entradas y salidas (read-write-outputable). Normalmente Inc presenta en la pantalla los nombres que están reconocidos por el módulo durante la simulación como nombres claves (read, write, increment, key name).

Inc define, según la respuesta del operador, los valores iniciales para M y N, los cables de lectura y escritura. Posee también la posibilidad de inicializar los cables de entrada en forma automática (auto-kick).

elementos del circuito CABLE. Ello se debe a la posibilidad de una simulación que se lleva a cabo mediante una computadora convencional, la cual posee un procesador central que es del tipo de von Neumann. Por otra parte lo que se está simulando es un procesador en paralelo y que ello se está realizando al procesamiento simultáneo en dos o más lugares diferentes de uno o varios datos, que posteriormente deberán interactuar. En un procesador de von Neumann la ejecución es en serie. Una manera de utilizarlo para simular el procesador en paralelo es resolviendo la simulación por secciones, las cuales permitan ser simuladas con un procesador secuencial.

La técnica se desarrolla de la siguiente manera. Se obtiene el resultado de la simulación de la primera sección. Este resultado es derivado en una localidad, llamada L1. A continuación se toma una segunda sección del procesador en paralelo, la cual en la realidad trabaja simultáneamente con la primera. El resultado resultado es puesto en la localidad L2. En caso de que se deba dividir artificialmente el procesador en paralelo en N secciones para poder ser simulado, al final se tendrá ocupada la localidad LN con el último resultado. Ahora se requiere combinar L1,...,LN mediante una operación representativa de su contraparte real, el resultado se escribe en L0. L1,...,LN son resultados de procesos concurrentes y es ello que el cálculo de L0 debe de esperar hasta que se dispongan de todos los anteriores. Sin embargo, el la operación sobre las L1 es asociativa, es posible ahorrar espacio de memoria y proceder precisamente en forma acumulativa con ayuda de una sola localidad: L0.

Regresando a la presente simulación, los resultados L1 son las H y B del centro de control de la palabra el procesador. El camino de la información y el sentido del flujo de los datos siempre se significa a menos se dan otra vez las líneas de flujo. Dicho de otra manera dos cables de salida nunca entran a un mismo cable de entrada (lectura o escritura), aunque no está afirmado que no se puedan reunir en un mismo módulo.

Una cosa contraria sucede si se toma como punto de vista el de las H y B. Ahora constantemente se están tomando caminos de información. Esta es precisamente una característica de los procesadores en paralelo y para la simulación con un procesador secuencial es conveniente usar la técnica antes descrita. En este caso particular

la escuela de Tuebn.

Lista/indexadores es, quizás, lista como en su cierto entre es una memoria de índices. Tiene 256 localidades de memoria, un registro de escritura (input register), por el que entran los datos que deben ser almacenados, un registro de lectura (index register), por el cual entran los índices con los que se indica la localidad de la memoria que se desea, y un registro de carrera (latch) abierto después del registro de salida (output register) que el objeto de no perder información en el caso de aparecer una R en el registro.

Lista posee adicionalmente conectores internos (switches) para seleccionar ciertas opciones del hardware del módulo. Mediante estas conexiones se escogen los nombres clave. Si un dato está etiquetado con un nombre clave al llegar al registro de escritura será escrito en la memoria si V=1 (si tiene una V). En el caso de que un dato que llegue al registro de lectura porte un nombre clave de lectura (una V (o en su defecto una C si es el último dato) el campo X será asumido como un miembro del conjunto de índices que desean leer la memoria). El nombre clave es el medio por el cual se evitan conflictos, ya que son tomados en cuenta sólo los datos de un mismo conjunto sin dejando fluyen por el mismo canal diversas familias de datos.

Otro conector selector de hardware es para escoger la prioridad. En la prioridad de escritura se tiene que al llegar dos datos válidos al módulo, uno al registro de escritura y otro al registro de lectura al principio del mismo ciclo, sólo es aceptado el dato de escritura y además aparece una R en el registro de lectura. Esta R permanece a la entrada del registro de lectura y es enviada constantemente a la salida del módulo anterior a través del cable de lectura (cable de salida del módulo anterior).

Finalmente, se tienen los conectores llamados parches. Un parche (patch) es el lugar donde puede ser cambiado el orden de los bits dentro de los campos X y N. La clave de los nombres, en la práctica consta de sólo los tres bits menos significativos de N; los conectores internos que separan las direcciones para las memorias pueden ser parchados con un bit del campo N. El bit que frecuentemente se utiliza es el bit más significativo del

índices de valor entero [illegible] los nombres con los siguientes valores: [illegible] = 2000 (base10) (base16) o bien N3 N2 N1N0 = 1010 (base2) y 8 (base10).

La combinación de los dos [illegible] de entrada del multiplexor son a su vez enviados en un demultiplexor, el cual en la práctica es el valor conducido por clave de nombre (name key). Como ilustración a este [illegible] resulta: si la clave de nombres de escritura vale 5 (base10) = 0(10) (base2) entonces los nombres seleccionados para la escritura son N = 0 (base16), N = 2 (base16), el igual que N = 5 (base16), N = A (base16).

[illegible]

[illegible] el nombre de la subrutina [illegible]
[illegible] : MODULO

: RUN MODULO

[illegible] presenta inmediatamente dos sumarios:

DATE IS 08-AUG-84 TIME IS 15:08:41
WHAT IS SUBROUTINE NAME FOR THE MODULE BEING TRANSFER

[illegible] desde [illegible] el módulo [illegible]
la cual [illegible] para muestrear [illegible] :

: LIST

[illegible] se desea [illegible] el modo QUICK, en lugar [illegible] el módulo
[illegible] directamente [illegible] frecuentemente usadas (default).

QUICK DEFAUL

[illegible] prueba 17 respondemos afirmativamente :

: YES

[illegible] se sitúa automáticamente el módulo llamado a la subrutina en [illegible]
[illegible] la variable STATE = -1. Para LIST se ha escrito [illegible] de los mensajes
[illegible] presenta un cuadro que muestra su configuración :

	XF	XE	XD	XC	XB	XA	X9	X8	X7	X6	X5	X4	X3	X2	X1	X0	N3	N2	N1	N0
[illegible] Predictor Patch:	1F	1E	1D	1C	1B	1A	19	18	17	16	15	14	13	12	11	10	03	02	01	00
[illegible] Cable Patch:											1X	1W	1V	1U	1T	1S	1R	1Q		
Multi-Writer Address Patch:	17	16									14	13								

[illegible]	PEX	PEN	PEredU	PEwellU	PEd
1	1	1	1	1	1

List of Recognized Name	[illegible]	F	E	D	C	B	A	9	8
Values in Table Key Names	[illegible]	7	6	5	4	3	2	1	0

[illegible] Key Name	1
[illegible] Table KeyName	1
[illegible] Table Key Name	1

```
T          XW-   FILE-COMP        CALE      OPS-WRT    OPT-OPT
1           0        0             1          1          0

                X   B   C W M S

    File        0   0
    Filter      0   0
    Outputs     0   0
    Filter      0   0
    Filter      0   0
    File        0   0
    File        0   0

Memory Locations

Locations saved

El puntero anterior señala que las entradas están convertidas en
líneas. También indica así cuántas serán transmitidas a COMP.
Cuando vienen B = 1 significa otra vez de operación (B = 0 significa de lectura)

CYCLE  1 WRITE  CABLE ------- INPUT OPTIONS

Para la entrada 1 se debe tener la dirección en Cuam opciones así que
también las primeras opciones son de WRITE CABLE.

INPUT OPTIONS

1001 WA= 1-C/2-HP= 0 1 0                    500114

T          XW-   FILE-COMP        CALE      OPS-WRT    OPT-MISC
1           0        0             1          1          0

                X   B   C W M S

    File        0   0
    Filter      0   0
    Outputs     0   1
    Filter      1   1
    Filter     4001 1      1
    File       4001 1      1
    File       4001 1      1
```

M00=5001 M01= M02= M03= M04= M05= M06= M07=

OUTPUT CABLE X16= 0 N4= 1 O=V= 0 0

CYCLE 2 WRITE CABLE - - - - INPUT CONTROL

A esta nueva pregunta responderemos con una VV ya que el valor en el campo de la X se ha introducido en un valor determinado en memoria.

VV

X16= 5001 N4= 1 O=V=H= 0 1 0 5001 1 1

D	Xwa	Pr1E=CONM	Cnt5	OP1=WRT	OP2=Wlst
1	1	0	1	1	0

X N O E V H S

Output Cable	0	1	
Output Register	0	1	
List Mem Outputs	2	1	
Index Register	1	1	
Input Register	5002	1	1
Index Cable	5001	1	
Input Cable	5002	1	1

List Memory Locations:

M00=5001 M01=5002 M02= M03= M04= M05= M06= M07=

OUTPUT CABLE X16= 0 N4= 1 O=V= 0 0 10

Este procedimiento se lleva a cabo repetidas veces hasta alcanzar el valor 1 en la posición M00 hasta M05. Y cada vez pertenecerá a VV adicionarlas como indicada al lista de escritura. Tmar esc.1921.4 variables similares a las anteriores.
El último pregunta será el siguiente:

X16= 1010 N4= 1 O=V=H= 0 1 0 5001 1

D	Xwa	Pr1E=CONM	Cnt5	OP1=WRT	OP2=Wlst
1	4	0	1	1	0

```
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

[illegible] Location:

[illegible] M01=5002  M02=5003  M03=5004  M04=5005  M05=5006  M06=  M07=

OUTPUT CABLE=X16=  0.Na= 1 Dev= 3.0

        Ahora llevaremos a cabo la lectura en forma [illegible] Primero [illegible]
un dato con una C. Ello pues es el último dato del conjunto. [illegible] [illegible]
una M, la entrada de operador [illegible] hasta que tome
[illegible] llegue una C al puerto de lectura.
        Primero [illegible] el último dato ( C ) [illegible] en el [illegible] X
para [illegible] el programador. El puerto [illegible] con un
valor M [illegible] de lectura [illegible] 4001 (hexadecimal). [illegible] dato
[illegible] al [illegible] el contenido de la memoria M01 [illegible]
[illegible] se [illegible].
        Se [illegible] en la fase de preguntas correspondiente al cable 01 [illegible] 50.
[illegible] índice a Tres que debe de preguntar por cada uno de los cables. A
[illegible] se indica con S ( SET ) [illegible] los números
X [illegible] M [illegible]. En la fase de lectura [illegible] pregunta por
[illegible] DEVO CABLE INPUT OPTION= lo cual se establece con 10. En seguida se
[illegible] 3 para [illegible] el cable de lectura X=4001, M=(no modificado),
[illegible] M 1, M=0. La secuencia de respuestas a preguntas sería:

CABLE= 1  WRITE CABLE ------ INPUT OPTION?
        SC
WRITE INPUT OPTION?
        3
M1  IN HEXADECIMAL= 5001------ NEW VALUE?
        5007
M1  IN HEXADECIMAL= 1------ NEW VALUE?
        1

[illegible] VALUES 0 OR 1
```

```
[illegible] 5007 N4= 1 C,V,H= 1 0 0                    5007[illegible]
[illegible] READ CABLE------ INPUT OPTIONS?
      10
[illegible] OUTPUT OPTIONS?
      0
[illegible] IN HEXADECIMAL= 5000------ NEW VALUE?
      1000
[illegible] IN HEXADECIMAL= 1------ NEW VALUE?
      1
[illegible] VALUES 0 OR 1
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]
[illegible]

[illegible] 5000 N4= 1 C,V,H= 0 1 0                    5000[illegible]

      B          Xwe     Fill COMP          Gate      OPS-WAIT     OPS-USED
      1           4          0              0           2            1

                    X   N    C V H E

[illegible] Cable          0   1
[illegible] Doublet        0   1
[illegible] Via Outputs    5001   1      1
[illegible] Doublet        0   1      1
[illegible] Doublet        5007   1    1   1
[illegible] Cable          5000   1      1
[illegible] Cable          5007   1    1   1

[illegible] Minors Locations:

  M00=5001   M01=5000   M02=5001   M03=5001   M04=5000   M05=5004   M06=5007   M07=

[illegible] MIRE MIRE MIRE  WRITE CABLE [illegible]=  0   [illegible]
OUTPUT CABLE X16=     0 N4= 1 C,V= 0 0                    10

      El siguiente valor de X es el cable de lectura de el 5001 la cual es solicita
[illegible] TW.
```

```
CYCLE  9 READ  CABLE------ INPUT OPTIONS
       WW
X16= 4001 MA= 1 C,W,H= 0 1 0                    600114

        0          X=0     CYCLE-COUNT        CHAE      OPTIONS     OPTIONVAL
        1            6            0             0           0            1

                     X  W   C V H E

Output Cable        5001  1     1
Output Resistor     5001  1     1
Last New Output     5002  1     1
Index Resistor         1  1     1
Input Resistor      5007  1   1 1
Index Cable         4001  1     1
Input Cable         5002  1   1 1

Will Modify Equations?

  M00=5001   M01=5002   M02=5003   M03=5004   M04=5005   M05=5006   M06=5007   M07=

000000000 MORE MORE MORE  WRITE CABLE NUMBER=  0    000000 000000000
OUTPUT CABLE X16=  5001 MA= 1 C,W= 0 0                    10

        Respondiendo A todas las opciones WW a la Respuesta de Instrucciones
terminamos de entrar todos atributos del convertidor de datos que funcionaremos
anteriormente. El cuarto resultado es el siguiente:

CYCLE 10 READ  CABLE------ INPUT OPTIONS
       WW
X16= 4002 MA= 1 C,W,H= 0 1 0                    600114

        0          X=0     CYCLE-COUNT        CHAE      OPTIONS     OPTIONVAL
        1            6            0             0           0            1

                     X  W   C V H E

Output Cable        5003  1     1
Output Resistor     5005  1     1
Last New Output     5006  1     1
Index Resistor         5  1     1
Input Resistor      5007  1   1 1
Index Cable         4002  1     1
Input Cable         5007  1   1 1
```

[illegible]

M01-M02 M01=5002 M02-5003 M03-5004 M04-[illegible] M05-[illegible] M06 M07

OUTPUT CABLE X16= 5005 N4= 1 C,W= 0 1 [illegible]
[illegible] WIRE WIRE WIRE WRITE CABLE-EITHER [illegible]

[illegible] Isolane 6L [illegible] [illegible] Se [illegible] el cobre [illegible]
[illegible] 0.

X16 4002 N4= 1 C,V,M= 1 0 0 [illegible]

1	Xwc	Pr1E=COMP	Cnt1E	OP1=WR1	OP2=WL1
1	4	1	0	1	0

	X	N	C	V	M	B
[illegible] Cable	5004	1	1			
[illegible] Resistor	5006	1	1			
[illegible] Output	0	1	1			
[illegible] Divider	0	1	1			
[illegible] Resistor	5007	1	1			
[illegible] Cable	4002	1	1			
[illegible] Cable	5007	1	1			

[illegible] Mixer Localized:

M00 5001 M01=5002 M02=5003 M03=5004 M04=5005 M05=5006 M06 5007 M07=

OUTPUT CABLE X16= 5006 N4= 1 C,W= 0 1 [illegible]

Y con un ciclo más se obtiene una C de salida:

X16 4002 N4= 1 C,V,M= 0 0 0 [illegible]

0	Xwc	Pr1E=COMP	Cnt1E	OP1=WR1	OP2=WL1
1	0	0	0	1	0

	X	N	C	V	M	B
Output Cable	0	1	1			
Output Resistor	0	1	1			
[illegible] Mix Output	0	1				
[illegible] Divider	0	1				

[illegible] 5002 1
[illegible] 5003 1
[illegible] 5007 1

[illegible] Localizaciones

[illegible] M01=5002 M02=5003 M03=5004 M04=5005 M05=5006 M06=5007 M07=

INPUT CABLE X16= 9 M4= 1 C.V. 1 0 18

El programa se corta pues se ha concluido lo grabado. Es cable con XY.

Para la rutina de escritura/lectura en bloques no requiere utilizar los ranchos. Al bit más significativo de los contadores sener dover de transmisores se los ranchos el bit más significativo del número correspondiente. Es decir, en este caso, se responde: QUICK SETUP? NO, OUTPUT TO WHAT RENDIMIENTO VAR? ya que no es necesario que los resultados queden almacenados en los archivos OUTPN.TI y OUTW.TI, que son en los que tras guardar sus resultados. Por lo que tan solo sirve los valores que son incrementales y de incremental, saltando el modo de QUICK se puede responder a la [illegible] de [illegible] siguiente:

WITH NON STANDARD PATCHINGS
YES
COMPLETELY NON STANDARD?
NO

La segunda respuesta es NO porque se debe aprovechar el resto de los ranchos en su forma estandarizada (DEFAULT). Ahorra una [illegible] de los [illegible] disco resultantes.

ANY CHANGES IN INPUT REGISTER STANDARD PATCH?
N
ANY CHANGES IN INDEX CABLE STANDARD PATCH?
N
ANY CHANGES IN READ ADDRESS STANDARD PATCH?
Y
X7=03
N
ANY CHANGES IN WRITE ADDRESS STANDARD PATCH?
Y
X7=03
N
WRITE PRIORITY?
Y

Tres [illegible] LIST [illegible]
[illegible] al [illegible] [illegible]

	XF	XE	XD	XC	XB	XA	X9	X8	X7	X6	X5	X4	X3	X2	X1	X0	N5	N4	N1	N0
Input Register Patch	1F	1E	1D	1C	1B	1A	19	18	17	16	15	14	13	12	11	10	0X	0D	0X	0C
Index Cable Patch									17	16	15	14	13	12	11	10				
Quad Vector Address Patch	37	36							07	06										

PFX	PFXA	PFN	PFnxhi	PFnxlo	PF3
1	1	1	1	1	1

List of Recognized Name [illegible] F E D C B A 9 8
Values by This Key Names [illegible] 7 6 5 4 3 2 1 0

Increment Key Name	1
Input Cable Key Name	1
Index Cable Key Name	1

S	XMD	PrSE	OSMP	UbSE	UPSrWRD	UPSrVIEL
1	0	0	0	0	0	0

	X	N	C	U	H	I
Output Cable	0	0				
Output Register	0	0				
Input Outputs	0	0				
Index Register	0	0				
Input Registers	0	0				
Input Cable	0	0				
Input Cable	0	0				

[illegible] Memory Locations:

[illegible] Memory locations empty:

Los datos de escritura se mandan con dos nombres alternadamente: N[illegible]N[illegible]. [illegible] lectura debe mantenerse el mismo orden alternando. Al final de la serie [illegible] el modulo LIST con sus registros llenados con las siguientes cantidades:

	X	H	C	V	H	Z
Output Cable	5004	1		1		
Output Resistor	5004	1		1		
Line Mix Output	M000	0		1		
Input Resistor	06	0		1		
Input Resistor	5005	1	1		1	
Input Cable	5006	0		1		
Input Cable	5006	1	1		1	

Field Numbers/Locations:

V10 5000	M01=	M02=5002	M03=	M04=5004	M05=	M06=5006	M07=
V20=	M01=5001	M02=	M03=5003	M04=	M05=5005	M06=	M07=

A less preamplifier OUTPUT CABLE---> INPUT SWITCH on the input panel
H 1 Turn the preamplifier 3, 4, 5 & 6, for resulting output preamplifier input channels
and circuitry.

P	Xwg	Pr1E=COMP	Cn1E	Ch1=WNT	Ch1=Wlet
1	7	0	1	1	0

	X	H	C	V	H	Z
Input Cable	0	1		1		
Input Resistor	0	1				
Line Mix Output	0	1				
Input Resistor	0	1				
Input Resistor	5003	1		1		
Input Cable	5000	1				
Input Cable	5003	1		1		

Field Numbers/Locations:

V10=5000	M01=5001	M02=5002	M03=5003	M04=	M05=	M06=	M07=

Y [illegible] 3 [illegible] 1

Y	Xwg	Pr1E=COMP	Cn1E	Ch1=WNT	Ch1=Wlet
1	7	0	0	1	0

```
                    X   N   C V H E

Output Cable        5000  1      1 1
Output Pulser       5001  1      1 1
Max Output          5002  1      1
Pulser                 2  1      1 1
Pulser              5006  1    1   1
Cable               6002  1      1 1
Cable               5006  1        1

Memory Locations:

M00=5000   M01=5001   M02=5002   M03=5003   M04=5004   M05=5005   M06=5006   M07=5007

   El diagrama de las conexiones que se tienen actualmente
   Tras en todo sesión que no use QUICK SETUP.  Lo llama as ésta

Use conventions for write and read cables

The possible answers to the first question are:
RETURN,S,U,D,W,X,M,E,CR,RD,SCR
Use the following meanings:
RETURN: data field(X16) and name field(N4) unchanged; C,V,H,E and C
S: X16=X16,N4=N4,V=H=N=0

                    Page 77
```

```
IF=1,X16=X16,N4=N4,C=H=B=C
X16 read 10, N1=N1,V=1,C=H=B=0
V=1,X16=X16+
Change X16 to comment for "VV" variable
F=1,X16=X16,N4=N4,C=V=H=0
Read the other questions
Read the other questions
X16,N4,C,V,H,B will be read individually
V=1,X16=X16,N4=N4,C=V=B=0
read this description
Read conventions for variable table

Variable answers to the first question are:
VERTICAL,D,HB,SP,RP,R
Variable the following meanings:
VERTICAL: R=B=0
V=1,R=0
F=1,H=0
H=0=1
SUSPEND PRINTING(I.E. NO STATE=1 CALL)
RESUME PRINTING
read this description
```

Con lo anterior se concluye la referente a las pruebas del módulo LISTA/INDEXADORA. La índole de las pruebas que no se presentan explícitamente aquí, es la misma que la de las que ya se probaron.

En la práctica las pruebas 1) a la 8) son más posible selección de la gran cantidad de ellas que se dan también. La elección de una prueba en particular dependerá de las características específicas que se desean examinar del módulo en turno. Pero de cualquier forma las pruebas aquí sugeridas se proponen como un conjunto de manera global. De tal manera que incluyan muchas de las características básicas que conforman el módulo en este caso: LIST.

BIBLIOGRAFIA

En el CAPITULO 1 se muestra diversas [illegible] [illegible] de [illegible] [illegible] (SICOB). A [illegible] de una [illegible] [illegible] [illegible] de las computadoras. Para una [illegible] sobre la configuración de sistemas [illegible] la [illegible] referencia:

[illegible] John P. "Microcomputer Architecture & [illegible]" [illegible] Wiley & Sons, 1981.

[illegible] & Blankenbaker "Principles of Computer [illegible]" McGraw Hill, 1980.

[illegible] & Blankenbaker "Principles of Computer [illegible]" McGraw Hill, 1983.

[illegible] "Superconductivity Computer Sciences" (Vol. [illegible] 25), mar., 1980.

[illegible] "Supercomputers" Sci. Am. (Vol. 246, No. 1, p. 135), ene., 1982.

En lo referente a microprocesadores se recomienda la siguiente bibliografía, además de los manuales correspondientes de cada uno de los microprocesadores ahora mencionados:

[illegible] Hardware Handbook" Digital Equipment Corporation, 1980.

[illegible] "Architecture of a New Microprocessor", [illegible] (Vol. 12, No. 2), feb., 1979. (F3000)

[illegible] & Dunter "A Microprocessor Architecture Changing World" Computer (Vol. 12, No. 2), feb., 1979. (8086)

[illegible] & Ichour "A Microprocessor for the Revolution" The 8089" Byte (Vol. 4, No. 1, 2, 3), ene., feb., mar., 1979.

[illegible] Gilliam & Ravenel "The Intel 8086 Microprocessor: A 16-bit Evolution of the 8080" Computer (Vol. [illegible] No. 6), jun., 1978.

Para una presentación más completa referente a [illegible] en gran escala se encuentran los libros de [illegible] en General a los que se presentan a continuación:

[illegible] Charles "Electronic Principles Physical [illegible] & Circuits" Wiley, 1969.

[illegible] 1 [illegible] "[illegible] Circuits Signals 2 Integrated Mode, [illegible] 1979.

Al interesado en profundizar en sistemas de lectura modular se le recomienda la lectura de las siguientes [illegible]

*Luecke, et al.: "Semiconductor Memory Design and Application", McGraw Hill, 1973.
*"VAX Hardware Handbook", Digital Equipment Corporation, 1980.
*Gibson: "The Cache Concept for Large Scale Computers", Rechenstrukturen, Springer Verlag, 1971.

En [illegible] los CAPITULOS III, IV, la bibliografía se el contenido de cada conferencia o comunicaciones privadas entre los miembros del grupo que trabajan en el proyecto conjunto entre la Universidad de UNAM, la Universidad de Columbia, la Universidad de Massachusetts, el Laboratorio Nacional de Brookhaven y el Laboratorio Nacional de Fermi:

*"A Review of Programmable Systems Applications with Parallel Experiments", Thomas Nash, Presented Talk at the Conference on the Application of Microprocessors to High Energy Physics Experiments, CERN, Geneva, Switzerland, May 4-6, 1981.
*"Conference on Instrumentation for Colliding Beam Physics", Quinta, Heinzel, Kheer & Siegrist, SLAC Stanford CA., Feb. 17-27, 1982.
*Conferencias: C. A. Ilias, Centro Brasileiro de Pesquisas Físicas, Río de Janeiro, Agosto 83.
*"Real Time Processing of Detector Data", Siegrist, Bohnenson, Knapp, IEEE Transactions on Nuclear Science, Vol. NS-27, No. 1, Feb. 80.
*Comunicación Privada: Avilez, Kheer, Abril 83.
*Conferencias: "A Super High Speed Data Driven Processor", Christian, Argonne National Lab., Physics Section, Marzo 83.
*Conferencias: Siegrist, Fermi National Accelerator Laboratory, Primavera 83.