

P.- 44.951

IBM

Docket

PO 9-69-029

380987

SECCION TECNICA
CLASIFICACION I.P.C.
CLASE G.06
SUBCLASE F.

Memoria descriptiva

17



380987

para solicitar PATENTE DE INVENCION por 20 años

a nombre de INTERNATIONAL BUSINESS MACHINES CORPORATION

entidad / ~~nacionalidad~~ norteamericana

con domicilio en Armonk, N.Y., Estados Unidos de América

por: "UN METODO DE BUSCAR UN ARGUMENTO DE BUSQUEDA EN UN  
"INDICE DE VARIOS NIVELES"

(Clase Internacional G11b)



La presente invención se refiere en general a la recuperación de información, y en particular a un nuevo método electrónicamente controlado para investigar en índices de varios niveles, que pueden leerse a máquina.

5

Con frecuencia cada vez mayor se está generando información de todas clases. Y se hace ver ahora más que nunca que existe a menudo un motivo de atasco o "embotellamiento" en no poderse recuperar o encontrar rápidamente un artículo de información de entre la gran masa de información en la que se halla sepultado. Si bien se ha trabajado mucho en la recuperación de información, no se ha hallado hasta ahora una solución de tipo general, aunque han llegado a concebirse muchos métodos complicados de recuperación de información, para tener acceso a una información que implique gran número de documentos o de registros.

10

15

Dentro del ambiente de la recuperación de información, la presente invención se refiere a un utensilio o medio útil de controlar una máquina para localizar información clasificada por claves. Las claves alfanuméricas, de cualquier tipo que sean, dispuestas en secuencia clasificada, pueden convertirse en claves comprimidas, y puede investigarse o buscarse en ellas por medio de la presente invención. Cada clave comprimida representa un límite o "lindero" (sea alto o bajo) para la clave sin comprimir a la que representa. Cada clave comprimida puede llevar asociados datos, o bien el lugar de situación de uno o más artículos de información a los que representa. La información de localización puede ser una

20

25

30

dirección de acceso, una indicación de señalamiento o "pointer", o bien puede ser derivable de la propia clave por medios que no forman parte del presente invento.



5 La presente invención incluye un algoritmo propio de la misma para investigar en un índice comprimido de varios niveles, mejorando grandemente la velocidad de búsqueda en niveles múltiples, en comparación con la búsqueda del índice sin compresión.

10 Se conocen ya, y se han revelado en el pasado, métodos y medios de investigar en un índice de niveles múltiples no comprimido. La búsqueda en índices no comprimidos se está efectuando con sistemas de calculadoras, usando métodos de acceso, medios de control y métodos electrónicos de catalogación especiales. Las patentes  
15 de EE.UU. núm. 3.408.631 de J.R. Evans, 3.315.233 de R. De Camp y col., y 3.366.928 de R. Raice y col.; así como las de núm. 3.242.470 de Hagelbarger y col., y 3.030.609 de Albrecht, son ilustrativas del estado de la técnica del ramo.

20 La recuperación de información de las calculadoras está actualmente limitada de varias maneras, entre las cuales cabe indicar la que impone la enorme cantidad o capacidad de almacenaje necesaria. El formato de claves en forma de índice de varios niveles sin comprimir da lugar a tenerse que explorar gran número de bytes (o baterías de bits) en todas y cada una de las entradas o asientos de claves, mientras se examina un argumento de búsqueda. Es ésta una laboriosa y costosa operación cuando se busca en un índice grande, o cuando se  
25 busca repetidamente en un índice más pequeño. Y esta es  
30



precisamente el área de ataque del presente invento, que reduce grandemente el número de bytes explorados por cada asiento o introducción de clave en un índice investigado. El resultado obtenido es el de reducir las necesidades de almacenaje o memoria de búsqueda y acelerar la investigación, por necesitarse detectar menos bytes a máquina. Se obtiene como consecuencia un apreciable aumento de la velocidad de búsqueda, sin modificar la velocidad de trabajo de un sistema de calculadora.

En los actuales métodos de búsqueda en calculadoras electrónicas, tales como los de las patentes arriba mencionadas, se tienen claves sin comprimir que acompañan a registros o grabaciones efectuados sobre un disco o tambor para clasificar u ordenar las materias o asuntos contenidos en un registro asociado. La búsqueda del registro de grabación asociado puede hacerse sea mediante la clave, sea mediante la dirección de acceso del registro. Por ejemplo, en las patentes de EE.UU. núms. 3.408.631; 3.350.693; 3.343.134; 3.344.402; 3.344.403 y 3.344.405, puede orientarse u ordenarse una clave sin comprimir en un disco magnéticamente registrado o grabado. Una clave, en un ambiente de varios niveles, puede explorarse electrónicamente por medio de un argumento de búsqueda, en busca de la condición de igualdad de comparación. Lograda la condición de igualdad de comparación, se obtiene una dirección de acceso de indicación asociada a la clave sin comprimir respectiva, y se usa dicha dirección para recuperar el registro o grabación a un nivel inferior, representado por la clave que puede estar en cualquier otro lugar del mismo dispositivo o en un dis



positivo diferente. Esta indicación o "pointer", por ejemplo, puede incluir el lugar de situación del dispositivo de disco, o de otro dispositivo, en donde esté registrado o grabado el registro del nivel inmediato inferior. El nivel de índice más bajo localiza el registro de datos que se está buscando, pudiendo entonces recuperarse el registro y usarse para cualquier fin requerido.

Esta invención concierne a la búsqueda o investigación en un índice comprimido de varios niveles. La compresión elimina un tipo de superfluidad o redundancia atribuible a la naturaleza clasificatoria del índice, es decir, elimina la redundancia de tipo inducido por la clasificación u ordenación, y sólo conserva la información mínima necesaria para la búsqueda. La investigación correcta en un índice comprimido de varios niveles implica sutilezas y condiciones críticas que no intervienen en la investigación de índices de varios niveles sin comprimir. El reconocimiento de estas características nada obvias es esencial para poder ir a buscar correctamente un registro requerido en el nivel inmediato inferior del índice, antes de poder ir a buscar el registro de datos correcto.

Por todo ello, es objeto de esta invención un método y un sistema, nuevos en su clase, de poder investigar en un índice de varios niveles, comprimido por eliminación de la redundancia de clasificación u ordenación, sin dejar por eso de ser capaz de ir a buscar el registro de índice adecuado en el nivel inmediato inferior.

Otro objeto de esta invención reside en un



método y un sistema, nuevos en su género, de poder investigar en un índice comprimido de varios niveles; reduciendo el número de bytes que se necesita explorar a máquina durante una búsqueda, en comparación con una búsqueda similar realizada en un índice de varios niveles correspondiente sin comprimir. Con esto se aumenta grandemente la velocidad de búsqueda de la máquina en relación con la velocidad de búsqueda en el índice de fuente clasificado sin comprimir, para una misma velocidad de bytes de la máquina.

Otro objeto de la invención es el de buscar en un índice comprimido en el que el tamaño de las entradas o asientos de claves a niveles múltiples es en gran parte independiente de la longitud de las correspondientes claves sin comprimir. Por ejemplo, una indicación que señale a un índice de nivel inferior viene acompañada de un par de claves comprimidas, que poseen sólo unos pocos bytes representativos de una clave no comprimida que podría tener centenares o millares de bytes. La magnitud de compresión del índice depende principalmente de la "rigidez" del índice, es decir, de la magnitud de la variación en las relaciones de clasificación u ordenación entre las claves sin comprimir del índice.

Otros objetos más concretos y específicos de este invento son:

A) El de buscar en un índice de nivel alto dotado de un formato de bloques comprimidos que permite la investigación con cualquier argumento de búsqueda sin comprimir.

B) Buscar en todos los niveles de índice



con un argumento de búsqueda que no esté en el índice original de claves no comprimidas (UK) a partir del cual se haya construido el índice comprimido, argumento de búsqueda que caería entre claves contiguas no comprimidas representadas: (1) dentro de un solo bloque de índice comprimido; o (2) en dos bloques de índice comprimidos.

5 C) Buscar en cada bloque de índice comprimido de varios niveles independiente de cualquier otro bloque comprimido. Esto evita tener que llevar cualquier información de búsqueda desde un bloque cualquiera de nivel alto a la operación de búsqueda de claves de un bloque de nivel inferior. La única información previa o anterior necesaria es la dirección de acceso de cualquier bloque en el que se vaya a investigar.

15 D) Buscar en un índice de varios niveles entrando en un bloque de índice cualquiera con un contador de igualdad de búsqueda puesto a cero.

E) Buscar en un bloque cualquiera de nivel alto con un formato de CK,CK,R para cada entrada o asiento, siendo R la indicación de señalamiento o "pointer", y cada CK una clave comprimida. La búsqueda puede incluir un nivel de índice mínimo, o del orden más bajo, con una sola CK por indicación.

25 F) Buscar desde la cima de un índice comprimido de varios niveles hasta hallar un bloque de datos, en que:

(1) se tiene acceso a un solo bloque comprimido por cada nivel de índice, y

(2) se encuentra el bloque de datos adecuado si estaba en el índice primitivo del cual se derivó el



índice comprimido, o bien

(3) el argumento de búsqueda no está en el índice, y la búsqueda indica un lugar del índice contiguo a aquel en el que se habría colocado el argumento de búsqueda, de haber estado en el índice original o primitivo.

G) Dar una entrada alternativa de búsqueda en el índice comprimido, al principio de un nivel cualquiera inferior al de la cima.

H) Poder efectuar una búsqueda completa para un argumento de búsqueda, mediante asiento o introducción del índice al principio de un nivel cualquiera y recorriendo en serie todo ese nivel hasta llegar a encontrar la clave alta adecuada o correcta, después de lo cual sólo puede tenerse acceso a un único bloque por cada nivel.

La invención investiga cada bloque de los que tienen un par de claves comprimidas por indicación, a niveles de índice por encima del nivel bajo. El par de claves comprimidas por indicación fué generado partiendo del par de claves no comprimidas (UKs) de lados opuestos del lindero representado entre bloques comprimidos contiguos, en el nivel de índice más bajo.

En esta invención los términos "bloque" y "registro" (de grabación) quieren dar a entender la misma cosa. Los bloques, en las formas de realización, pueden estar físicamente separados, o bien pueden ser diferentes bloques lógicos dentro del mismo bloque físico.

Este invento hace distinción entre la búsqueda hecha en el nivel más bajo de un índice de varios



niveles, y la investigación de los niveles de dicho índice superiores al más bajo. Con el término "nivel bajo" se hará referencia en lo que sigue al más bajo nivel del índice de varios niveles, y con el de "nivel alto" se hará referencia a un nivel cualquiera que está por encima del "nivel bajo".

Con esta invención, los bloques investigados de nivel alto del índice tienen un formato distinto del de los bloques de índice buscados de nivel bajo. El formato de nivel alto lleva asociadas dos claves comprimidas (CKs) formando par, con una sola indicación, que tiene acceso a un bloque de índice de nivel inmediato inferior; en tanto que el formato de nivel bajo asocia a cada indicación o "pointer" una sola CK, para tener acceso a un bloque de nivel de datos. En el formato de nivel alto, la primera CK de cada par indica el cambio de índice dentro del bloque designado por la indicación asociada, y la segunda CK de la pareja indica el cambio de índice, desde la última clave del bloque designado por la indicación asociada, hasta la primera clave del siguiente bloque de la secuencia del índice. Ambas claves comprimidas de la pareja necesitan obtener una búsqueda correcta con cualquier argumento de búsqueda posible que haya de recuperar la indicación asociada. Así, la primera CK de la pareja es la primera CK alta cuando el argumento de búsqueda cae dentro de la gama de índice del bloque de índice señalado o seleccionado por la indicación; y la segunda CK de la pareja es la primera CK alta cuando el argumento de búsqueda está a un nivel bajo en la última clave del bloque, o cae entre este bloque del índice, y

380987



el bloque siguiente. Por tanto, un argumento de búsqueda que caiga entre dos bloques de índice de nivel bajo está relacionado con el bloque a cuyo final iría ligado en cualquier actualización futura.

5                   Cada bloque de nivel alto o de nivel bajo del índice puede investigarse independientemente de cualquier otro bloque. Esta independencia de los bloques del índice se define como la aptitud de buscar en un bloque cualquiera del índice, al que se haya tenido acceso, sin  
10                   confiar en el resultado de la búsqueda efectuada en cualquier otro bloque del índice, excepto en la dirección de acceso de indicación o "pointer" que pudiera resultar de una búsqueda anterior. La independencia de la búsqueda de bloques exige que el contador de igualdad de comparación de bytes y argumento de búsqueda, usado en la operación de búsqueda, se vuelva a la posición o condición  
15                   inicial antes de investigar en un bloque de índice dado cualquiera. Esta reiniciación del contador de igualdad destruye cualquier información que pudiera contener, procedente de una búsqueda anterior.  
20

                  Esta independencia de búsqueda permite también elegir arbitrariamente cualquier bloque de índice que tenga capacidad o pueda ser investigado. La independencia de búsqueda o investigación en los bloques proporciona con esta invención las características siguientes:  
25

                  - La independencia permite que todo bloque, del que se sospeche que tiene una condición de error, sea él solo seleccionado y comprobado, investigándolo o examinándolo con unos argumentos de búsqueda prefijados.  
30



- La independencia permite seleccionar un solo bloque cualquiera y actualizarlo, sin que ello afecte a ningún otro bloque del índice.

5 una completa investigación del índice en un nivel cualquiera de éste, saltándose así todos los niveles superiores. Este nivel, en el que se haya entrado inicialmente, puede tener uno solo o varios bloques. El número de bloques por nivel aumenta exponencialmente a medida que disminuye el nivel del índice; únicamente el nivel de cima o más alto puede tener un solo bloque, pero el nivel de cima puede proyectarse con varios bloques.

15 - La característica de independencia últimamente descrita permite también acrecentar la fiabilidad de la búsqueda, al permitir que se salten uno o más cualesquiera o todos los niveles superiores, empezando la búsqueda por un nivel cualquiera a voluntad.

20 - La independencia permite que la búsqueda en un nivel cualquiera deseado proceda en serie recorriendo toda su sucesión o secuencia de bloques hasta hallar la indicación adecuada que señala al bloque de nivel inmediato inferior. Si el nivel investigado es un nivel alto, es posible iniciar una búsqueda a varios niveles con la indicación adecuada hasta recuperar el bloque de datos requerido.

+ La independencia da una flexibilidad operativa con un índice que tenga un número fijo de niveles, número que puede ser cualquiera, aunque sea uno solo.

30 - La independencia permite una búsqueda binaria en los bloques de un nivel cualquiera, partiendo de



una tabla de indicaciones (direcciones de acceso de blo-  
que) para ese nivel. En la tabla, las indicaciones van en  
secuencia por el orden de clasificación u ordenación de  
los bloques de ese nivel. Por ejemplo, puede usarse una  
5 tabla como esa en lugar del nivel de cima de cualquier ín-  
dice comprimido. (Una distinción aplicable en la práctica  
es la de que una búsqueda binaria exige indicaciones acce-  
sibles de manera aleatoria, en tanto que un solo bloque  
comprimido cualquiera exige ser tomado o leído en serie.)  
10 La búsqueda binaria recupera primero la indicación o poin-  
ter de enmedio de la tabla para seleccionar el bloque de  
enmedio de ese nivel de índice. Si la primera clave del  
bloque de enmedio es superior al argumento de búsqueda,  
se almacena su indicación; y luego se recupera el bloque  
15 de enmedio de la primera mitad del índice, usando la in-  
dicación de enmedio de la primera mitad de la tabla de  
indicaciones. En cambio, si la que es alta es la última  
clave del bloque de enmedio, se almacena su indicación,  
y se selecciona el bloque de enmedio de la segunda mitad  
20 señalado por la indicación de la segunda mitad de la ta-  
bla de indicaciones. Se prosigue así, de modo que a cada  
paso se estrecha la zona de búsqueda en  $1/2^n$  bloques al  
nivel uno, siendo  $n$  el número de bloques hasta aquí in-  
vestigados, hasta encontrarse la indicación correcta. Es-  
25 ta indicación correcta puede encontrarse dentro de un blo-  
que cualquiera investigado, y viene señalada por el hecho  
de que la indicación almacenada del bloque no es su pri-  
mera ni su última indicación. Si una indicación almacena-  
da es la primera o la última de un bloque cualquiera de  
30 índice, su posible idoneidad viene indicada si la búsqe-



da binaria almacena la indicación siguiente del extremo opuesto del bloque sucesivo; en este caso se guardan o al macenan ambas indicaciones, determinándose por verificación cuál de ellas es la correcta o adecuada. La búsqueda binaria termina también cuando no hay más indicaciones o "pointers" que seleccionar en la tabla de ellas, y en este caso la última tomada por lectura (sea primera o última) es la adecuada.

Los precedentes y otros objetos, rasgos característicos y ventajas de la invención se irán desprendiendo de la siguiente descripción pormenorizada de unas formas preferidas de realización del invento, ilustradas en los dibujos adjuntos, en los cuales:

- la figura 1A ilustra un índice sin comprimir, de nivel alto;
- la figura 1B ilustra el índice comprimido de nivel alto, derivado de aquél;
- las figuras 2A y 2B ilustran unos circuitos de registro regulador y de entrada/salida (I/O) usados para guardar un índice sin comprimir de nivel alto y el índice comprimido resultante, respectivamente;
- la figura 3 muestra una disposición de regulación o marcación de tiempos y control de modos;
- la figura 4A ilustra la marcación de tiempos del modo de generación, y
- la figura 4B ilustra la marcación de tiempos del modo de búsqueda, para el circuito de las figuras 9A y 9B;
- la figura 5A ilustra un formato para un bloque de índice comprimido de nivel bajo, en tanto que



- la figura 5B ilustra un formato para un bloque de índice comprimido de nivel alto;

5 - la figura 6 representa una estructura de bloque de índice comprimido de varios niveles, generada conforme a esta invención;

- la figura 7A ilustra, en vista general resumida, un sistema de calculadora que contiene la presente invención, y

10 - las figuras 7B y 7C ilustran unas tablas que hay en dichas formas de realización;

- la figura 8 ilustra los controles de enlace y el descodificador de órdenes de mando para una forma de realización de sistema de control de busca en índice de varios niveles;

15 - las figuras 9A y 9B ilustran unos controles de reloj de modos usados en una versión del modo de búsqueda del presente invento;

20 - las figuras 10 y 11 muestran unos controles usados para generar e investigar en un índice comprimido;

- las figuras 12 y 13 representan unos circuitos usados en la búsqueda o investigación de un índice comprimido;

25 - las figuras 14A a 14G inclusive ilustran un método de búsqueda realizado con arreglo a una forma de ejecución de este invento; y

- las figuras 15 a 19 inclusive ilustran una forma de realización de un sistema de control de búsqueda de índice de varios niveles.

30



### Descripción general

Un índice de varios niveles que puede investigarse por medio de esta invención es el que puede ser -  
generado según la exposición de la solicitud de patente -  
5 española nº. 380.988, presentada el mismo día que la pre-  
sente y cedida al mismo cesionario.

Un índice de varios niveles investigado con  
forme al presente invento es el representado en la figura  
6 por los niveles de índice comprimido L1 a L4 inclusive.  
10 Una búsqueda o investigación recupera información del ni-  
vel de datos (L0). El índice de varios niveles incluye un  
índice de nivel bajo comprimido L1 y unos índices compri-  
midos de nivel alto L2, L3 y L4. Un quinto nivel no es --  
comprimido, y puede ser una entrada o asiento en un catá-  
15 logo de sistema de calculadora usual; esta entrada com-  
prende el nombre de la base de datos L0 y una dirección  
de acceso (indicación)  $R_{4-1}$  que sitúa o localiza el blo-  
que 4-1 del índice comprimido, del nivel de cima L4.

El nivel de datos L0 comprende una gran plu-  
20 ralidad de bloques de datos, clasificados o señalados ca-  
da uno por su clave no comprimida (UK), que incluye desde  
un primer bloque de información, de clave UK(A<sub>1</sub>) hasta un  
último bloque que tiene por clave UK(@<sub>n</sub>). La selección de  
la clave para cada bloque no forma parte de esta inven-  
25 ción, y puede recurrirse a la práctica usual de tomar un  
campo cualquiera de un bloque, que se usa para orientar  
el bloque. Por ejemplo, la clave puede ser un campo del  
bloque que represente un artículo de inventario, un núme-  
ro personal, un número de departamento, un número de re-  
30 serva, un número de matrícula de automóvil, etc., mien-



1  
tras otras partes del bloque representan información cla-  
sificada por la clave. Los bloques del nivel de datos L0  
pueden localizarse al azar, en cualquier lugar donde haya  
espacio en un dispositivo de memoria de acceso aleatorio  
5 (por ejemplo, en un mando de disco magnético, un tambor  
magnético o un dispositivo de archivo en tiras). No hay  
necesidad de que los bloques tengan en los niveles L0 a  
L5 ninguna relación rígida de tipo posicional, secuencial  
o de otra clase. Cada uno puede estar situado en cualquier  
10 lugar donde se disponga de espacio en el dispositivo, con  
tal que se den como entrada a esta invención las direccio-  
nes de acceso de los bloques en el espacio disponible. Es  
requisito primario para una rápida recuperación el de que  
el dispositivo pueda tener acceso rápidamente a un bloque  
15 cualquiera, cuando se le dé su dirección respectiva.

Los bloques de la figura 6, en el nivel L0,  
se representan por el orden de la secuencia de clasifica-  
ción de sus claves no comprimidas,  $UK(A_1)$  a  $UK(@_n)$ . Esta  
representación ordenada se incluye en la organización de  
20 la estructura de búsqueda o clasificación a nivel múlti-  
ple. Ahora bien, esta relación ordenada no tiene relación  
alguna de posición con los lugares de los bloques de da-  
tos o de índice del o de los dispositivos de acceso alea-  
torio en los cuales se guarden los bloques. Una consecuen-  
25 cia conveniente de esta organización de orientación o bús-  
queda en posiciones aleatorias es la de que deja de ser  
necesario mudar un bloque no modificado siempre que se  
agreguen nuevos bloques en cualquier lugar de su secuen-  
cia de clasificación.

Es preferible, aunque no obligatorio, que



el nivel más alto no tenga más que un solo bloque.

La búsqueda de cualquier bloque de L0, usando esta estructura de clasificación, sólo exige esa toma de acceso de un bloque por cada nivel de clasificación o de índice a velocidad de calculadora, sea cual fuere el número de bloques de un nivel cualquiera. Por tanto, en la figura 6, cualquier bloque de L0 requerido puede ser directamente recuperado al sexto acceso de bloque, tras cinco accesos de bloque de índice, a partir del nivel L5 hacia abajo, pasando por los niveles L4, L3, L2, L1 y L0. Ninguno de los seis accesos es afectado por el número de bloques que haya en cualquiera de estos niveles, incluido el nivel de datos L0.

El comienzo de cada bloque de índice está situado en una dirección de acceso, llamada indicación de señalamiento R, que tiene dos números como subíndice. El primer subíndice representa el nivel del bloque seleccionado al que se ha tenido acceso, y el segundo representa la posición de clasificación del bloque seleccionado, en su nivel particular. Las indicaciones  $R_{3-1}$  a  $R_{3-3}$  inclusive de dentro del nivel L4 sitúan los respectivos bloques 3-1 a 3-3 inclusive del nivel L3. De igual modo, cada una de las indicaciones  $R_{2-1}$  a  $R_{2-9}$  del nivel L3 sitúa un respectivo bloque 2-1 a 2-9 inclusive en L2. Igualmente, las respectivas indicaciones  $R_{1-1}$  a  $R_{1-27}$  de L2 sitúan en L1 los respectivos bloques 1-1 a 1-27 inclusive. Finalmente, cada indicación  $R_{A1}$  a  $R_{An}$  sitúa un bloque respectivo en el nivel de datos L0.

En el nivel L1, cada clave comprimida (CK) lleva como apéndice una indicación: por ejemplo, la pri-



1  
 mera  $CK(A_1)$  lleva la indicación  $R_{A_1}$  como apéndice para si  
 tuar el primer bloque de  $L_0$ ; y cada bloque del nivel  $L_1$   
 se genera por medio de un método de índice comprimido.

5 Una base de datos  $L_0$  muy grande puede ser  
 manipulada por la estructura de orientación o clasifica-  
 ción de la figura 6. Por consiguiente, este índice puede  
 manipular un elevadísimo número de claves para búsqueda  
 entre un número correspondiente de bloques del nivel  $L_0$ .  
 Por ejemplo, las tablas B y C que siguen representan un  
 10 índice comprimido que dará acomodo a 27.000 bloques de  
 datos por separado dentro del nivel  $L_0$ , si cada bloque de  
 $L_1$  incluye 1.000 claves comprimidas (CKs), que es un nú-  
 mero admisible en la práctica. La tabla A representa el  
 índice no comprimido correspondiente al índice comprimido  
 15 de las tablas B y C. En otro ejemplo, si se supone que to-  
 dos y cada uno de los bloques de índice de los niveles  $L_1$   
 a  $L_4$  inclusive de la figura 6 tienen 35 indicaciones -  
 ("pointers") por bloque, los cuatro niveles de índice ad-  
 mitirán para clasificación hasta 1.500.625 bloques de da-  
 20 tos al nivel  $L_0$ . Por tanto, resulta posible recuperar al  
 azar cualquiera de los 1.500.625 bloques de datos con cin-  
 co accesos de máquina, que pueden hacerse en menos de un  
 segundo usando siete dispositivos diferentes de acceso di-  
 recto (DASD), cada uno con un tiempo medio de acceso de  
 25 menos de 200 milisegundos, lo que puede lograrse con la  
 actual tecnología de los dispositivos de acceso directo.

30 En el caso especial en que todo bloque de  
 índice tenga un número  $C$  de claves y se use un número  $j$   
 de niveles de índice, el número máximo de bloques de  $L_0$   
 acomodados es de  $C^j$ .



He aquí algunos ejemplos de uso de cuatro niveles de índice (j=4):

5 (1) Usando 100 indicaciones por bloque: con 1.010.101 bloques de índice en los cuatro niveles se puede alojar un máximo de 100.000.000 de bloques de datos en el nivel L0.

10 (2) Usando 1.000 indicaciones por bloque, con 1.001.001.001 bloques de índice en los cuatro niveles se puede alojar un máximo de 1.000.000.000.000 (un billón) de bloques de datos en el nivel L0.

15 En ambos ejemplos (1) y (2) se necesitan cinco accesos de bloque para ir a buscar un bloque cualquiera de datos de L0, iniciando la búsqueda por el bloque de nivel más alto. Si en lugar de UKs se usan CKs en cada bloque de índice, el número de bloques de índice se reduce usando bloques de la misma longitud de byte, o bien se reduce la longitud de byte de los bloques de índice  
20 usando el mismo número de bloques de índice. Así, para una compresión de 1/10 usando CKs, en el ejemplo (1) se podría (a) reducir en una décima parte el número de bloques de índice, con la misma longitud de byte, para un total de 101.011 bloques de índice, o bien (b) reducir en  
25 una décima parte la longitud de byte para cada uno de los 1.010.101 bloques. Una compresión semejante en el ejemplo (2) podría conducir (a) a usar la misma longitud de byte reduciendo el número total de bloques de índice a --  
30 100.100.101, o bien (b) reducir en una décima parte la longitud de byte de cada uno de los 1.001.001.001 bloques



1/ J

de índice.

La tabla A siguiente ilustra un "índice de  
varios niveles sin comprimir", que tiene cuatro niveles  
de índice L1 a L4 inclusive de bloques, y a partir del  
5 cual se genera el "Índice comprimido de varios niveles"  
de las siguientes tablas B y C:

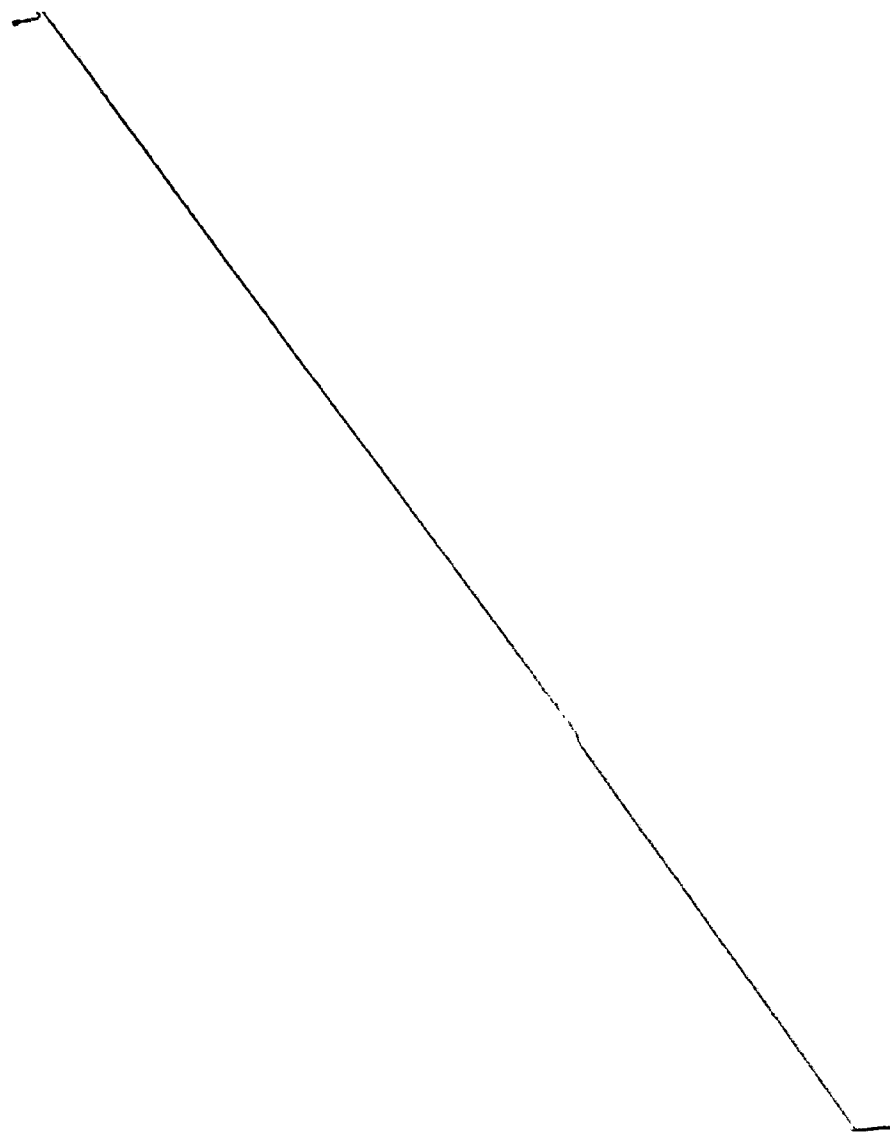




TABLE A

INDICE NO COMPRIMIDO DE VARIOS NIVELES

<u>L1</u>			<u>L2</u>			<u>L3</u>			<u>L4</u>		
<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>
1-1	A <sub>1</sub>	R <sub>A1</sub>	2-1	A <sub>n</sub>	R <sub>1-1</sub>	3-1	C <sub>n</sub>	R <sub>2-1</sub>	4-1	I <sub>n</sub>	R <sub>3-1</sub>
	'	'		B <sub>1</sub>			D <sub>1</sub>			J <sub>1</sub>	
	<u>A<sub>n</sub></u>	<u>R<sub>An</sub></u>									
1-2	B <sub>1</sub>	R <sub>B1</sub>		B <sub>n</sub>	R <sub>1-2</sub>		F <sub>n</sub>	R <sub>2-2</sub>		R <sub>n</sub>	R <sub>3-2</sub>
	'	'		C <sub>1</sub>			G <sub>1</sub>			S <sub>1</sub>	
	<u>B<sub>n</sub></u>	<u>R<sub>Bn</sub></u>									
1-3	C <sub>1</sub>	R <sub>C1</sub>		C <sub>n</sub>	R <sub>1-3</sub>		I <sub>n</sub>	R <sub>2-3</sub>		@ <sub>n</sub>	R <sub>3-3</sub>
	'	'		D <sub>1</sub>			J <sub>1</sub>			FIN	
	<u>C<sub>n</sub></u>	<u>R<sub>Cn</sub></u>		<u>-----</u>			<u>-----</u>			<u>-----</u>	
										Final de	
										índice	
										en L4	
1-4	D <sub>1</sub>	R <sub>D1</sub>	2-2	D <sub>n</sub>	R <sub>1-4</sub>	3-2	L <sub>n</sub>	R <sub>2-4</sub>			
	'	'		E <sub>1</sub>			M <sub>1</sub>				
	<u>D<sub>n</sub></u>										
1-5	E <sub>1</sub>	R <sub>E1</sub>		E <sub>n</sub>	R <sub>1-5</sub>		O <sub>N</sub>	R <sub>2-5</sub>			
	'	'		F <sub>1</sub>			P <sub>1</sub>				
	<u>E<sub>n</sub></u>	<u>R<sub>En</sub></u>									

380987



TABLA A (Cont.)

INDICE NO COMPRIMIDO DE VARIOS NIVELES

<u>L1</u>			<u>L2</u>			<u>L3</u>			<u>L4</u>		
<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>
1-6	F <sub>1</sub>	R <sub>F1</sub>		F <sub>n</sub>	R <sub>1-6</sub>		R <sub>n</sub>	R <sub>2-6</sub>			
	'	'									
	'	'		G <sub>1</sub>			S <sub>1</sub>				
	<u>F<sub>n</sub></u>	<u>R<sub>Fn</sub></u>		---	---		---	---			
1-7	G <sub>1</sub>	R <sub>G1</sub>	2-3	G <sub>n</sub>	R <sub>1-7</sub>	3-3	U <sub>n</sub>	R <sub>2-7</sub>			
	'	'									
				H <sub>1</sub>			V <sub>1</sub>				
	<u>G<sub>n</sub></u>	<u>R<sub>Gn</sub></u>									
1-8	H <sub>1</sub>	R <sub>H1</sub>		H <sub>n</sub>	R <sub>1-8</sub>		X <sub>n</sub>	R <sub>2-8</sub>			
	'	'									
				I <sub>1</sub>			Y <sub>1</sub>				
	<u>H<sub>n</sub></u>	<u>R<sub>Hn</sub></u>									
1-9	I <sub>1</sub>	R <sub>I1</sub>		I <sub>n</sub>	R <sub>1-9</sub>		@ <sub>n</sub>	R <sub>2-9</sub>			
	'	'									
				J <sub>1</sub>			FIN				
	<u>I<sub>n</sub></u>	<u>R<sub>In</sub></u>		---	---		---	---			

Final de  
índice  
en L3

SIMBOLOS

- BL. = bloque
- UK = clave sin comprimir
- PTR = indicación o "pointer"

380927

TABLA A (Cont.)



INDICE NO COMPRIMIDO DE VARIOS NIVELES

<u>L1</u>			<u>L2</u>			<u>L3</u>			<u>L4</u>		
<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>
1-10	$J_1$	$R_{K1}$		$K_n$	$R_{1-11}$						
	'	'		$L_1$							
	$K_n$	$R_{Kn}$									
1-12	$L_1$	$R_{L1}$		$L_n$	$R_{1-12}$						
	'	'		$M_1$							
	$L_n$	$R_{Ln}$									
1-13	$M_1$	$R_{M1}$	2-5	$M_n$	$R_{1-13}$						
	'	'		$N_1$							
	$M_n$	$R_{Mn}$									
1-14	$N_1$	$R_{N1}$		$N_n$	$R_{1-14}$						
	'	'		$O_1$							
	$N_n$	$R_{Nn}$									
1-15	$O_1$	$R_{O1}$		$O_n$	$R_{1-15}$						
	'	'		$P_1$							
	$O_n$	$R_{On}$									

380987



TABLA A (Cont.)

INDICE NO COMPRIMIDO DE VARIOS NIVELES

<u>L1</u>			<u>L2</u>			<u>L3</u>			<u>L4</u>		
<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>
1-16	$P_1$	$R_{P1}$	2-6	$P_n$	$R_{1-16}$						
	'	'		$Q_1$							
	$P_n$	$R_{Pn}$									
1-17	$Q_1$	$R_{Q1}$		$Q_n$	$R_{1-17}$						
	'	'		$R_1$							
	$Q_n$	$R_{Qn}$									
1-18	$R_1$	$R_{R1}$		$R_n$	$R_{1-18}$						
	'	'		$S_1$							
	$R_n$	$R_{Rn}$									
1-19	$S_1$	$R_{S1}$	2-7	$S_n$	$R_{1-19}$						
	'	'		$T_1$							
	$S_n$	$R_{Sn}$									
1-20	$T_1$	$R_{T1}$		$T_n$	$R_{1-20}$						
	'	'		$U_1$							
	$T_n$	$R_{Tn}$									

38 0987

TABLA A (Cont.)

INDICE NO COMPRIMIDO DE VARIOS NIVELES



<u>L1</u>			<u>L2</u>			<u>L3</u>			<u>L4</u>		
<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>
1-21	$U_1$	$R_{U1}$		$U_n$	$R_{1-21}$						
	'	'		$V_1$							
	$U_n$	$R_{Un}$		---	---						
1-22	$V_1$	$R_{V1}$	2-8	$V_n$	$R_{1-22}$						
	'	'		$W_1$							
	$V_n$	$R_{Vn}$									
1-23	$W_1$	$R_{W1}$		$W_n$	$R_{1-23}$						
	'	'		$X_1$							
	$W_n$	$R_{Wn}$									
1-24	$X_1$	$R_{X1}$		$X_n$	$R_{1-24}$						
	'	'		$Y_1$							
	$X_n$	$R_{Xn}$		---	---						
1-25	$Y_1$	$R_{Y1}$	2-9	$Y_n$	$R_{1-25}$						
	'	'		$Z_1$							
	$Y_n$	$R_{Yn}$									

38 0987

TABLA A (Cont.)



INDICE NO COMPRIMIDO DE VARIOS NIVELES

<u>L1</u>			<u>L2</u>			<u>L3</u>			<u>L4</u>		
<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>UKs</u>	<u>PTRs</u>
1-26	Z <sub>1</sub>	R <sub>1</sub> Z <sub>1</sub>	Z <sub>n</sub>	R <sub>1-26</sub>							
	'	'	@ <sub>1</sub>								
	<u>Z<sub>n</sub></u>	<u>R<sub>Zn</sub></u>									
1-27	@ <sub>1</sub>	R <sub>1</sub> @ <sub>1</sub>	@ <sub>n</sub>	R <sub>1-27</sub>							
	'	'	FIN								
	<u>@<sub>n</sub></u>	<u>R<sub>@n</sub></u>	-----/-----								
	Final de		Final de								
	índice		índice								
	en L1		en L2								

38 0987

TABLA B



INDICE COMPRIMIDO DE VARIOS NIVELES

<u>L1</u>			<u>L2</u>		
<u>BL.</u>	<u>CKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>CKs</u>	<u>PTRs</u>
1-1	CK(A <sub>1</sub> ), 	R <sub>A1</sub>	2-1	CK(A <sub>n</sub> ), CK(B <sub>1</sub> ), R <sub>1-1</sub>	
	'	'		CK(B <sub>n</sub> ), CK(C <sub>1</sub> ), R <sub>1-2</sub>	
	'	'			
	OO ,	R <sub>An</sub>		CK(C <sub>n</sub> ), OO / , R <sub>1-3</sub>	
	-----			-----	
1-2	CK(B <sub>1</sub> ), 	R <sub>B1</sub>		'	
	'	'		'	
	'	'		'	
	OO ,	R <sub>Bn</sub>		'	
	-----			-----	
1-3	CK(C <sub>1</sub> ), 	R <sub>C1</sub>		'	
	'	'		'	
	'	'		'	
	OO ,	R <sub>Cn</sub>		'	
	-----			-----	
	'			'	
	'			'	
	'			'	
	'			'	
	'			'	
	'			'	
	'			'	
	'			'	
	-----			-----	

SIMBOLOS

BL. = bloque  
 CK = clave comprimida  
 PTR = Indicación

38 0987

TABLA B (Cont.)



INDICE COMPRIMIDO DE VARIOS NIVELES

<u>L1</u>			<u>L2</u>		
<u>BL.</u>	<u>CKs</u>	<u>PTRs</u>	<u>BL.</u>	<u>CKs</u>	<u>PTRs</u>
1-25	$CK(Y_1),$	$R_{Y1}$	2-9	$CK(Y_n), CK(Z_1),$	$R_{1-25}$
	'	'		$CK(Z_n), CK(@_1),$	$R_{1-26}$
	'	'			
	$OO ,$	$R_{Yn}$		$CK(@_n), OO / ,$	$R_{1-27}$
	-----			-----	
1-26	$CK(Z_1),$	$R_{Z1}$			
	'	'			
	$OO ,$	$R_{Zn}$			
	-----				
1-27	$CK(@_1),$	$R_{@1}$			
	$OO ,$	$R_{@n}$			
	-----				

**38 0987**





En la tabla A, columna L1, se ilustran los bloques de nivel de índice L1 más bajo, de claves no comprimidas (UKs), obtenidos partiendo de los campos de claves de los bloques de información del nivel de datos L0.

5 Los bloques de información del nivel L0 no necesitan situarse en ningún orden en particular, y se supone que su localización es de tipo aleatorio. Las claves se toman de cualquier campo de los bloques de información de L0 que se quieren clasificar u ordenar. Después de obtenidas las  
10 claves de bloques de L0, se clasifican y organizan en bloques (se bloquean) para generar la secuencia de bloques de UK de L1, por ejemplo, como en la columna L1 de la tabla, por medios de programación o de equipo físico ya conocidos en la técnica del ramo y que no forman parte  
15 de esta invención. Por tanto, las UKs y sus bloques se hallan en la secuencia de clasificación en la columna L1, y se guardan en forma que pueda dar o servir de entrada al modo de generar, de esta invención.

Por ejemplo, pueden guardarse o almacenarse  
20 en un dispositivo de cinta de I/O (entrada/salida) de manera secuencial (por ejemplo, en los veintisiete bloques secuenciales 1-1 a 1-27 inclusive de la tabla A, columna L1). Estos bloques de UKs se usan respectivamente, por esta invención, para generar los bloques de claves no comprimidas 2-1 a 2-9 inclusive indicados en la columna L2  
25 de la tabla A. Los bloques de UKs de la columna L2 se usan luego para generar los bloques de UKs de la columna L3, y así sucesivamente hasta que se genera el nivel más alto L4, que comprende un solo bloque de UK.

30 Por consiguiente, en la solicitud de paten-



te española nº. 380.988 antes citada, cada nivel actual  
(de un momento dado) de bloques de UKs se usa para gene-  
rar el nivel inmediato superior de bloques de UKs. Es  
más, mientras se genera el nivel de bloques de UKs inme-  
diato superior, la forma de realización que aquí se deta-  
5 lla comprime también las claves del citado nivel actual  
de UK.

La longitud de los bloques de UKs en un ni-  
vel dado cualquiera viene determinada por el tamaño nece-  
sario para los bloques de ese nivel. El límite o lindero  
10 del final de cada bloque de la tabla A, en la columna L1,  
está representado por líneas de trazo interrumpido  
(-----) y algunas de estas líneas de trazos interrumpidos  
tienen además una o más barras o líneas inclinadas  
15 que las cortan (/) para indicar la significación del lin-  
dero a niveles superiores. Todos los linderos de bloques  
del nivel L2 en la tabla A están identificados, pues, por  
el símbolo ----/---- ; todos los linderos de bloques de  
L3 lo están con el símbolo ----//---- ; y todos los lin-  
20 deros de bloques de L4 lo están con el símbolo ----///---- .  
El uso de estos linderos de nivel superior como linderos  
de L1 indica su nivel de significación.

Las UKs situadas a lados opuestos de cada  
lindero de final son significativas en la generación de  
25 las claves comprimidas de nivel superior, y se llaman  
"UKs de lindero". Por tanto, cada lindero de final de  
bloque está representado por un par de "UKs de lindero".

La secuencia de UKs de segundo nivel (L2)  
representada en la columna L2 de la tabla A comprende  
30 todas las "UKs de lindero" de la secuencia de bloques de



L1.

La secuencia de UKs de tercer nivel (L3) re-  
presentada en la columna L3 de la tabla A comprende el úl-  
timo par de UKs de cada bloque de UKs de la secuencia de  
5 nivel L2. El último nivel (L4) del ejemplo de la tabla A  
comprende el último par de UKs de cada bloque de UKs de  
la secuencia de nivel L3.

Ciertas "UKs de lindero" de L1 forman el úl-  
timo par de UKs del final de cada bloque del nivel supe-  
rior. Así, en el nivel L1, uno de cada tres linderos iden-  
10 tifica un par de "UKs de lindero" usadas para finalizar  
cada bloque del nivel L2; un lindero de L1 de cada nueve  
define las "UKs de lindero" usadas para finalizar cada  
bloque del nivel L3; y el último (el 27º) lindero de L1  
15 define las "UKs de lindero" usadas para finalizar el blo-  
que de nivel más alto, al nivel L4. Así, las "UKs de lin-  
dero" que finalizan el bloque de nivel alto dan fin asi-  
mismo al último bloque de cada "nivel alto" inferior (por  
encima de L1), y representan también las últimas "UKs de  
20 lindero" del nivel bajo L1.

El número de UKs de cada nivel alto (L2 y su-  
periores) se supone de seis en el ejemplo de la tabla A.  
Cada par de UKs de nivel alto y una indicación o "pointer"  
genera dos CKs correspondientes, con la misma indicación,  
25 como se halla en las tablas B y C.

En la práctica, en cualquier bloque puede  
disponerse un gran número de indicaciones o "pointers",  
cada una con un par de CKs. El tamaño del bloque, en la  
práctica, viene determinado por el usuario de la inven-  
30 ción, y dependerá del tipo de almacenaje o memoria de que



se disponga para el índice de varios niveles, y de la velocidad de búsqueda requerida.

5 El tamaño de un bloque comprimido está en relación directa con la velocidad de búsqueda, ya que todo bloque individual es investigado secuencialmente desde su principio. Por tanto, cuanto más corto sea el bloque, menor será el tiempo de búsqueda invertido al recorrerlo. Rara vez es necesario buscar hasta el final en un bloque dado cualquiera, ya que la búsqueda o investigación termina en cuanto el argumento de búsqueda queda por bajo respecto de cualquier clave comprimida de un bloque. Una buena regla empírica para determinar el tiempo medio de búsqueda por bloque es la de tomar el tiempo necesario para explorar la mitad de un bloque.

15 El número de bloques explorados secuencialmente por un argumento de búsqueda es, en general, igual al número de niveles del índice de varios niveles. Así, la velocidad de búsqueda es independiente del número de bloques que haya en un nivel dado cualquiera. Otros factores que intervienen en la determinación del tamaño práctico de los bloques de varios niveles son el rendimiento de utilización del espacio de almacenaje en los dispositivos particulares de entrada/salida (I/O) en los que pueda haber bloques almacenados, y el tiempo de acceso de los mismos.

25 Aun cuando en la tabla A se indican bloques de igual tamaño para todos los niveles altos, es éste un caso especial. El tamaño de los bloques, expresado en número de claves comprimidas por bloque, puede estar representado por  $C_1, C_2, \dots, C_j$ , en los respectivos niveles



1, 2, ... j, siendo j el nivel más alto. C/2 representa el número de indicaciones de señalamiento de un bloque de índice de nivel alto, siendo este nivel alto el L2 o uno superior. También es C/2 el número de bloques de nivel inmediato inferior orientados o clasificados por este mismo bloque. Con C<sub>1</sub> se representa el número de indicaciones que hay en un bloque de L1.

K<sub>1</sub>, K<sub>2</sub>, ... K<sub>j</sub> representan el número de bloques que hay en los niveles indicados por los subíndices respectivos. El número K de bloques decrece exponencialmente según aumenta el número del nivel. Por tanto, el número total de bloques de un índice es K<sub>1</sub> + K<sub>2</sub> + ... + K<sub>j</sub>. Así, pues, este número disminuye yendo de K<sub>1</sub> a K<sub>j</sub>. En el nivel más bajo L1 se usa solo una CK por indicación, y K<sub>0</sub> = K<sub>1</sub> · C<sub>1</sub>.

En el caso especial en que el número de indicaciones o "pointers" (R) por bloque sea el mismo para todos los niveles del índice, y K<sub>j</sub> = 1, se tiene C = K<sub>0</sub>/K<sub>1</sub> = K<sub>1</sub>/K<sub>2</sub> = ... = K<sub>j-1</sub>. Este caso especial es el representado en la tabla A. El número total de bloques de datos de LO manipulados por este caso especial es de R<sup>j</sup>.

Las tablas B y C muestran los cuatro niveles del "Índice comprimido de varios niveles" derivado del "Índice no comprimido de varios niveles" indicado en la tabla A. Las tablas B y C tienen el mismo número de bloques que la tabla A, pero cada bloque de las tablas B y C es mucho más pequeño, debido a la singular compresión del índice. Por consiguiente, existe una relación de uno por uno entre los respectivos bloques de los índices comprimido y sin comprimir.

380987



La figura 7A da un resumen del ambiente pa-

ra una forma de realización del presente invento, cuyas -  
 etapas se ejecutan en gran parte por medio de unos contro-  
 les de índice 516. Incluye una unidad de canal y/o central  
 de tratamiento (CEU) 511, que conecta una memoria 510,  
 por medio de unas líneas de transmisión y control 511A,  
 a unos controles de enlaces 512 y a unos controles 530 de  
 I/O. Los controles de I/O 530 establecen conexión con una  
 pluralidad de dispositivos de I/O (entrada/salida) 530C-1,  
 530C-2 y 530C-3. Los dispositivos 530c llevan guardado  
 el índice comprimido de varios niveles. Se necesita más  
 de un dispositivo de I/O solamente cuando el índice com-  
 primido requiera una capacidad de almacenaje mayor de la  
 que puede obtenerse con un solo dispositivo. En este sen-  
 tido, el número de referencia 530c es representativo de  
 cualquier número necesario de dispositivos en los cuales  
 se almacene el índice. Cada dispositivo 530c tiene de  
 preferencia la posibilidad de un rápido acceso aleatorio  
 a base de bloque por bloque, y puede consistir en uno o  
 más discos magnéticos, tambores magnéticos, o archivos  
 de tira magnética.

Por tanto, antes de la iniciación de cual-  
 quier método en la figura 14D, se necesita que el o los  
 dispositivos de I/O 530c contengan el índice de varios ni-  
 veles, de bloques comprimidos; y que la memoria 510 esté  
 cargada con la necesaria tabla de órdenes, que contenga  
 unas órdenes descodificables por medio del descodificador  
 de órdenes 513 de la figura 7A.

Antes de iniciar una búsqueda en un deter-  
 minado nivel, es necesario asimismo cargar o introducir

en la memoria 510 las tablas de indicaciones de la figura 7C.



Búsqueda en varios niveles (a nivel múltiple)

5 Un argumento de búsqueda (S.A.) es una partida, artículo o elemento sin comprimir que forma parte - de una lista de argumentos de búsqueda contenida en la memoria 510 de la figura 7A. Hay que buscarlo en el índice de varios niveles representado en la figura 6, que puede estar guardado en uno o más de los dispositivos 530c de I/O. El argumento de búsqueda puede ser cualquier artículo o grupo alfanumérico que incluya caracteres especiales de cualquier clase; y puede incluir asimismo una clave cualquiera sin comprimir (UK) usada en la generación del índice de varios niveles, la que puede denominarse UK primitiva u original. Si el artículo es una UK original, se encontrará por medio de una búsqueda en varios niveles. En cambio, si el artículo o partida no es una UK primitiva u original, no está representada en el índice de varios niveles y, por tanto, no será hallada; pero el bloque particular se encuentra en el lugar en que dicha partida se colocaría en el índice comprimido, caso de que más tarde se fuese a actualizar el índice para incluirla.

25 En la figura 6, la búsqueda empieza por hallar el lugar de colocación del bloque comprimido de cima 4-1, usando la indicación  $R_{4-1}$  almacenada en el quinto nivel. Se trata de la indicación almacenada al final de la generación del índice de varios niveles en la solicitud de patente española antes citada. Esta indicación, por lo tanto, sitúa el lugar en que el argumento de búsqueda debe empezar a buscar o investigar en el índice de varios



niveles.

A continuación se lee el bloque 4-1, tras haberse tenido acceso al mismo con la indicación  $R_{4-1}$ . Al leerse en serie las claves comprimidas, tomándolas del  
5 bloque 4-1, se van comparando respectivamente contra el argumento de búsqueda, usando para ello un método de búsqueda en bloques uno por uno. Al ir comparándose las CKs respectivamente con el argumento de búsqueda, una de las CKs del bloque de cima dará un nivel alto de comparación,  
10 y su indicación asociada  $R_{3-x}$  se almacenará transitoriamente.

La indicación almacenada  $R_{3-x}$  procedente del bloque de cima se usa luego para seleccionar el bloque adecuado en el nivel inmediato inferior (L3). Este  
15 bloque se lee o toma de igual manera, y sus CKs se comparan con el argumento de búsqueda hasta encontrar la primera CK alta, y su indicación asociada  $R_{2-x}$  se almacena transitoriamente para recuperar el bloque adecuado en el nivel inmediato inferior (L2). A continuación se recupera  
20 el bloque con  $R_{2-x}$  en el nivel inmediato inferior, y se toman sus CKs leyéndolos contra el argumento de búsqueda hasta percibirse su primera CK alta y almacenarse su indicación  $R_{1-x}$  asociada. Este procedimiento de recuperar un bloque en cada nivel inmediato inferior se prolonga  
25 hasta que la última indicación o "pointer" recuperada selecciona uno de los bloques del nivel de datos (L0). Cada bloque de datos tiene uno o más campos que deben definir su clave UK primitiva, usada en la construcción del índice. Este campo es recuperado del bloque de datos y comparado  
30 con el argumento de búsqueda, para verificar si el



1  
bloque de datos recuperado es el requerido. Es el bloque  
de datos adecuado si su campo prefijado da igualdad de -  
comparación con el argumento de búsqueda. Si en la compa  
ración no dan igualdad, el argumento de búsqueda no está  
5 en el índice comprimido. Ahora bien, los bloques de va-  
rios niveles seleccionados son los únicos bloques del ín-  
dice que necesitarían ser actualizados, si más adelante  
se necesitase incluir este argumento de búsqueda en el ín-  
dice.

10 La búsqueda a niveles múltiples descrita en  
el párrafo inmediatamente precedente se obtiene por el  
método ilustrado en las figuras 14A a 14F inclusive. EL  
acceso al bloque adecuado se hace, en la búsqueda, por el  
método de las figuras 14D a 14F inclusive. La búsqueda  
15 dentro de un bloque cualquiera seleccionado para acceso  
se efectúa por el método de las figuras 14A a 14C inclu-  
sive. Las figuras 14A a 14C inclusive son esencialmente  
idénticas a las figuras 14A, 14B y 14C de la solicitud de  
patente americana nº. 788.876; en ella se describe y rei-  
20 vindica un método de investigar o buscar dentro de cual-  
quier bloque. Por consiguiente, las figuras 14A a 14C in-  
clusive de la presente solicitud no se describen con de-  
talle, ya que se confía la descripción a la de dicha so-  
licitud nº. 788.876, tomada como referencia, salvo en al-  
25 gunos cambios o modificaciones en sus etapas. Estos cam-  
bios son: en la figura 14A, las entradas por D10 y D12,  
que sustituyen a la etapa de iniciación; en la figura 14B,  
se han añadido las etapas 884, 866 y 867; y en la figura  
14C, la etiqueta de salida C10 sustituye a la etapa de fi-  
30 nal (END) indicada en este dibujo en la citada solicitud.

380987



anterior.

El equipo físico ilustrado en las figuras 3 y 9A a 13 inclusive ejecuta el método de las figuras 14A, B y C de la presente solicitud, de igual manera que en la mencionada solicitud americana nº. 788.876, con un solo cambio en la figura 13, que consisten la prolongación de la salida del circuito de coincidencia 332 que tiene la designación de "bloque actual alto en P = 0".

El método de búsqueda en varios niveles se introduce en la figura 14D en la etapa inicial 910, que representa una operación de iniciación inducida manual o automáticamente, para tener acceso a la etapa siguiente 911 que comprende la acción de ir a buscar una orden de "inscribir argumento de búsqueda y primera indicación" tomándola de la tabla de órdenes de la memoria 510 (figura 7A), y transmitiéndola por la barra colectora 511A desde la CPU al descodificador de órdenes 513, que descodifica la configuración única y singular de bitios de su código de operación, activando la línea de salida 513I de la figura 8, que va a las figuras 17 y 19. Esta orden es ejecutada en la CPU por su transmisión del argumento de búsqueda y de una indicación (inicialmente, la que señala al bloque de cima) a los controles de enlaces 512.

En la etapa 912 se entra al mismo tiempo cuando, en la figura 19, se activa la línea 513I que va a los controles 667 de S.A. y acceso a indicaciones, los cuales activan entonces su línea de salida 667A poniendo el contador 11a de direcciones de memoria baja (figura 15) a la dirección de acceso prefijada del registro de argumentos de búsqueda contenido en la memoria baja 10 de la



figura 2B. El registro de indicaciones o "pointer" de la memoria baja 10 se halla contiguo a su registro de S.A.

La puerta electrónica 621 de la figura 17 es activada por la misma línea 513I, de modo que los bytes de argumento de búsqueda recibidos de la CPU por la línea 511B de salida de barra colectora pasan por la puerta 621 a la llegada de barra colectora de datos de memoria baja (621A), que es recibida por el circuito disyuntivo 522a de la figura 15, el cual hace pasar el argumento de búsqueda al registro necesario de la memoria baja 10. Durante la recepción del argumento de búsqueda, una línea 666A de campo de S.A. habilita la puerta 621 por medio del circuito disyuntivo 622 de la figura 17. La línea 666A viene del descodificador 666 de la figura 19, al tiempo que reconoce las direcciones de acceso de byte precedentes del contador 11a, dentro del campo de registro de S.A. de la memoria baja 10. A continuación, el descodificador 666 activa la línea 666C cuando el último byte de S.A. (de relleno, si es necesario) activa el contador de direcciones 11a poniéndolo en el lugar de registro de indicaciones en la memoria baja 10, continuándose la transferencia de bytes de indicación hasta que se termina, para que la indicación sitúe el bloque de cima del índice comprimido en el registro de indicaciones de la memoria baja 10. La etapa 912 se termina al introducirse en la memoria baja 10 el registro de S.A. y el hallado para la indicación.

En la etapa 913 se tiene la señal de C.E. y D.E. procedente del circuito de báscula 624 de la figura 17, después de que el circuito de coincidencia 623 activa dicho circuito de báscula 624 en respuesta a haberse acti-



vado la línea 512C (de terminación de transferencia de la CPU) que viene de los controles de enlaces 512 de la figura 8. A continuación, la CPU transmite una señal de aceptación de estado desde el control de enlaces 512, por la línea 512D, que repone el circuito de báscula 624 dando fin a su señal de C.E. y D.E.

Una búsqueda a niveles múltiples exige que la etapa 920 se ponga en acción en respuesta a la etapa 913, para que la CPU vaya a buscar una orden de "buscar" como orden sucesiva, de la tabla de órdenes de la memoria 510. (Para la búsqueda en bloques múltiples en un solo nivel se usa la etapa alternativa 940, que se describe más adelante.)

La orden de "buscar" es transmitida al decodificador de órdenes 513, que activa su línea de salida 513J que va a las figuras 17 y 18. Al mismo tiempo se activan las etapas 921 y 922.

Con la etapa 921, la señal presente en la línea 513J activa el circuito monoestable 625 (figura 17), activando a su vez el circuito de báscula 626 que da una señal de control de selección de I/O y lectura a los controles 530h de selección de I/O de la figura 17, los cuales forman parte de los controles 530 de I/O de la figura 16. Esto condiciona la barrera 627 y los controles 530g.

También se lleva la línea 513J de órdenes de búsqueda, a los controles 667 de la figura 19, para activar la línea 667B de acceso al registro de indicaciones, que hace que la memoria baja 10 haga salir su campo de registro de indicaciones poniéndolo en la salida 14 de barra colectora de la memoria baja, campo que se hace pa-

17



sar por medio de la puerta condicionada 527 (figura 17) a los controles condicionados 530g de selección de I/O. Cada dispositivo 530c de I/O es seleccionado con la indicación o "pointer", y es el mismo dispositivo en que se inscribió; el índice de varios niveles, por el método de generación de la solicitud de patente española nº. 330.988 anteriormente citada. Al bloque de cima se tiene acceso luego en el dispositivo 530c de I/O, usándose esta indicación para completar la ejecución de la etapa 921.

La etapa 922 se ejecuta cuando la línea 513J de órdenes de búsqueda activa un circuito monoestable 625 (figura 17), que aplica impulsos por la línea 625B que va a la figura 13, donde repone el contador 301 de igualdad de argumento de búsqueda. Esto hace que la búsqueda a efectuar en el bloque siguiente sea independiente de los resultados de la búsqueda en cualquier bloque anterior, excepto por lo que se refiere a la dirección de acceso del bloque actual, o en curso en un momento dado.

Entonces se entra en la etapa 923, donde empieza la lectura o toma del bloque al que se ha obtenido acceso, saliéndose por D10 a la figura 14A para comparar el argumento de búsqueda con los bytes contenidos en este bloque, de la manera descrita con detalle en la solicitud americana anteriormente citada nº. 788.876. Al llegarse a la etapa 844, en la figura 14A, el byte pasado al registro de niveles 268 (figura 12) activa su línea de salida 268A de nivel alto que va a la figura 18.

Cuando se está ejecutando el método de la figura 14B, se entra en la etapa 866 para tomar por lectura el formato de bloque de nivel alto, en el que cada in-



dicación lleva asociada una segunda CK. La etapa 866 opera, respecto al circuito binario de báscula 211 de la figura 9B, indicando que en una operación de nivel alto se activa el circuito de coincidencia 209, y el circuito de báscula 213 de "sigue P" ha de ponerse en acción para ejecutar la etapa 867 en preparación para tomar por lectura la segunda CK del par de nivel alto. Sigue la etapa 884 para controlar la lectura de la segunda CK de un par.

Durante la búsqueda del bloque de cima (o de cualquier otro bloque) se halla una indicación, y se guarda en el registro de indicaciones de la memoria baja 10 mediante las etapas 886-891 de la figura 14C. Esta indicación se usa más tarde para tener acceso al bloque adecuado del nivel inmediato inferior del índice de varios niveles. La etapa 892 (figura 14C) indica eventualmente (por la activación del circuito de báscula 328) que se ha encontrado una clave alta en relación con el argumento de búsqueda, o bien (por la activación del circuito de báscula 333) que se ha leído la indicación o "pointer" de la última clave que tenía  $P=0$ , ya que ninguna CK del bloque daba alta de comparación con el argumento de búsqueda. La etapa 894 indica luego que se ha activado o "puesto" ("set") el circuito de báscula 331 de "búsqueda terminada" (figura 13), y el circuito de coincidencia 336 ejecuta la etapa 895 al dar una salida de reposición general que hace que de la figura 14C se salga por C10 a la etapa 930 de la figura 14D. La etapa 930 determina si la orden en curso es una orden de simplemente "buscar", o bien de "buscar en nivel único"; que son las dos opciones de que se dispone en este lugar del método.

07 JUL 1970



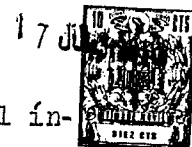
Con una orden de "buscar" se sale de la etapa 930 a la 931, que determina el nivel en curso de acuerdo con la señal presente en la línea de nivel alto 268A o en la línea de nivel bajo 268B, que van de la figura 12 al circuito de coincidencia 642 o al 644, respectivamente, -  
 5 de la figura 18. Como inicialmente existe un nivel alto, se activa primero la línea 268A.

A continuación se entra en la etapa 921, activándose el circuito 642 para disparar un monoestable  
 10 643, que pone en acción el circuito de báscula 626 de la figura 17. La etapa 921 se ejecuta luego de la manera anteriormente explicada para esta etapa. La salida del circuito de báscula 626 obtiene acceso y toma por lectura el bloque señalado por la indicación en curso existente en  
 15 el registro de indicaciones o "pointers" de la memoria baja 10, colocado allí por los elementos del método indicados en la figura 14C.

Se ejecuta a continuación la etapa 922, de la manera ya explicada para esta etapa, y se sale por D10  
 20 a la figura 14A, repitiéndose las etapas de las figuras 14A, 14B y 14C como antes se ha explicado, en busca del bloque en curso para la indicación que señala al bloque adecuado del nivel inmediato inferior, indicación que se superpone en el registro de indicaciones de la memoria  
 25 baja 10, de la manera anteriormente descrita. A continuación se toma la salida por C10 (figura 14C) a la figura 14D, etapa 930. Como la existente es la orden de "buscar", se vuelve a entrar en la etapa 931.

La repetición periódica del método en las  
 30 figuras 14D, 14A, 14B y 14C continúa, localizándose con

380987



5 cada ciclo el bloque del nivel inmediato inferior del índice de varios niveles. En la etapa 931 se examina siempre el estado del registro de niveles 268 de la figura 12, acusado por el byte denotativo de nivel del último bloque  
10 leído, que no representa necesariamente el nivel del bloque al que se ha de dirigir la indicación o "pointer" en curso en el registro de indicaciones de la memoria de nivel bajo 10. Así, solo después de tomado o leído un bloque de nivel bajo (L1) es cuando el registro de niveles  
15 268 indica un nivel bajo para la etapa 931, para poder señalar que el bloque del nivel de datos 10 es el representado por la indicación en curso en el registro de indicaciones de la memoria 10.

15 Por consiguiente, el método de búsqueda a nivel múltiple pasa cíclicamente de la etapa 931 a la etapa 921 mientras se esté buscando o investigando en el índice de varios niveles. La búsqueda a niveles múltiples se termina esencialmente al hallarse la indicación de L0 y colocarse en el registro de indicaciones de la memoria  
20 baja 10, lo que es señalado cuando la etapa 931 indica un nivel bajo. Se sale luego a la etapa 932, que toma esta indicación hallada, por lectura de la memoria 10, llevándola a la CPU. En la etapa 932, la indicación hallada para el bloque de datos de L0 se hace pasar a la CPU de modo  
25 que la CPU pueda recuperar el bloque de datos requerido, que puede estar almacenado en un dispositivo de I/O no accesible desde los controles 530 de I/O. Por ejemplo, un bloque de datos como ese pudiera estar en otro control del mismo cuadro de datos, o bien en otro cuadro disponible  
30 para la misma CPU; o bien pudiera estar disponible para otra CPU con la que esta CPU pudiera comunicar en un sis-

380987



tema múltiple de tratamiento. La etapa 932 se activa por medio de la línea de nivel bajo 268B, que activó el circuito de coincidencia 644 (figura 18), el cual envía señales por la línea 644A para activar los controles 667 -  
5 de la figura 19 delimitando o deslindando la dirección - de acceso del registro de indicaciones, y activa a su vez el circuito de báscula 645 de la figura 18 habilitando la puerta 646 para que deje pasar los bytes de indicación hallados en la salida de barra colectora 14 de la memoria  
10 baja (figura 15), llevándolos a través del circuito disyuntivo 647 de la figura 18 a la llegada de barra colectora 647A y a los controles de enlaces 512 de la figura 8, que transmiten la indicación a la CPU. El descodificador 666 de la figura 19 indica, por medio de una salida puesta  
15 en la línea 666C, cuándo la transferencia ha llegado al final del campo de registro de indicaciones. La línea 666C activa entonces el circuito de coincidencia 648 de la figura 18, señalando C.E. y D.E., lo que indica el final de la operación de búsqueda.

20 Cuando la etapa 933 (figura 14D) da la señal de C.E. y D.E. procedente de la línea 649A, la CPU contesta con una señal en la línea 512D de aceptación de estado, por medio de los controles de enlaces, 512, (figura 8). Esto da fin a la operación de búsqueda. Queda  
25 entonces a cargo de la CPU la recuperación del bloque de datos y la verificación de si este bloque es el que tiene en su clave el argumento de búsqueda, o bien si el argumento de búsqueda no está en el índice.

Búsquedas en un solo nivel (a nivel único)

30 Una búsqueda puede hacerse: (1) secuencial-



mente en la totalidad de cualquier nivel, por medio de los bloques del nivel individual; o (2) como búsqueda binaria en el nivel; o bien (3) como una combinación de los casos (1) y (2). Para cualquiera de los casos (1), (2) o (3), se usa una orden de "buscar en un solo nivel", pero se dan diferentes indicaciones para la etapa 911. Para una búsqueda secuencial, la primera indicación de la etapa 911 se dirige al primer bloque del principio del nivel a investigar, que puede ser un nivel cualquiera de un índice de varios niveles, tal como el representado en la figura 6. El nivel en que se ha entrado se investiga en serie de bloque a bloque hasta encontrarse la clave correcta, que es transferida a la CPU 511. Queda a cargo de la CPU 511 almacenar esta indicación, tal como en CPU Reg-1 en la memoria 510 de la figura 7A.

En el caso (2) de la búsqueda binaria, la indicación para el bloque de enmedio del nivel es usada primeramente por la etapa 911. En el caso (3) de una búsqueda binaria parcial seguida de una búsqueda secuencial de terminación, el último bloque de búsqueda en la binaria puede usarse para determinar la indicación de principio y el número de bloques que queda por investigar en la sección o tramo pequeño del nivel en cuestión, separado por la búsqueda binaria para la sucesiva búsqueda secuencial.

Estos tres casos de investigación de un solo nivel por medio del equipo físico expuesto exigen distinto apoyo de programación de búsqueda en tablas por parte de la CPU. Estos programas de búsqueda en tablas son fundamentalmente ya conocidos y, por consiguiente, no se



estudian aquí con detalle. En el caso de la investigación  
secuencial de un solo nivel, las indicaciones de la tabla  
de indicaciones son recogidas secuencialmente por el pro-  
grama de búsqueda en tablas, para dar salida a la indica-  
5 ción que corresponde al siguiente bloque del nivel en  
cuestión. En el caso de la búsqueda binaria parcial segui-  
da de una búsqueda secuencial parcial del nivel en cues-  
tión, se invocan ambos programas de investigación de ta-  
blas, y el lugar de indicación resultante de la búsqueda  
10 binaria se hace pasar al principio de la búsqueda secuen-  
cial parcial. Este paso de parámetros de un programa a  
otro es ya un método conocido en la técnica de las calcu-  
ladoras hasta ahora.

La iniciación del tipo de búsqueda en un so  
15 lo nivel, o a nivel único, se hace iniciando el tipo ne-  
cesario de programa de búsqueda en tablas para obtener la  
indicación correcta y llamar al control de equipo físico  
de búsqueda 512, 513, 516 y 530 indicado en la figura 7A,  
que efectúa entonces la búsqueda real y efectiva en el  
20 índice, con los dispositivos 530c de I/O.

Una búsqueda a nivel único puede hacerse  
dentro de cualquier nivel de un índice de varios niveles,  
que tenga una pluralidad de bloques, sea en o por debajo  
del nivel de cima. La búsqueda a nivel único se termina  
25 hallando la indicación adecuada dentro del nivel en cues-  
tión.

La búsqueda a nivel único permite investi-  
gar en un nivel de cima que tenga más de un bloque, lo  
cual es importante. El nivel de cima puede generarse en-  
30 tonces con más de un solo bloque, cuando se necesite un



número fijo de niveles en un índice de varios niveles.

Labúsqueda a nivel único acrecienta asimismo la fiabilidad de la investigación, permitiendo soslayar un fallo en cualquier nivel alto. En este caso puede usarse la búsqueda en un nivel para entrar en el índice de varios niveles por cualquiera de estos que se halle debajo del nivel que ha fallado. No se necesita entonces información de índice procedente de niveles superiores.

Al terminarse la búsqueda a nivel único en un nivel alto cualquiera, esto es, por encima del nivel bajo (L1), puede iniciarse una búsqueda a nivel múltiple con la indicación hallada por la búsqueda a nivel único, para localizar un bloque del nivel de datos (LO) requerido.

En cualquier tipo de búsqueda a nivel único, secuencial o binaria, la búsqueda se termina en cuanto se lee o toma una indicación entre las indicaciones primera y última de un bloque cualquiera en la búsqueda en curso, lo que ocurre la primera vez que, en comparación con el S.A., una clave comprimida (CK) del bloque, comprendida entre su primera CK y su última CK (P=0), da un nivel alto de comparación.

Más complicadas son las cosas si la indicación primera o la última se toman del bloque que se está investigando en ese momento.

Cada bloque del índice se genera de modo que su última CK (de P=0, sin bytes, y asociada a la última indicación o "pointer" del bloque) debe dar alta en comparación con cualquier argumento de búsqueda (S.A.). Durante una búsqueda cualquiera, la última indicación del

1 JUL



bloque investigado se lee y se lleva al registro de indicaciones de la memoria baja 10, de no haberse tomado ninguna otra indicación del bloque.

5 En una búsqueda secuencial, la última lectura de indicación indica que es preciso continuar la búsqueda en el bloque siguiente para determinar: (1) si esta última indicación puede orientar correctamente al S.A.; y (2) si se necesita seguir investigando para hallar la indicación correcta.

10 En una búsqueda binaria, ésta debe continuar si se toma por lectura la primera o la última indicación de un bloque, hasta que: (1) se llegue al último nivel de búsqueda binaria, situación en la cual la última indicación es la correcta; o bien (2) se lean dos indicaciones contiguas tomadas de bloques adyacentes de distinta clase, y en este caso es la CPU la que debe decidir cuál es la correcta.

20 La figura 14G representa el caso especial de la situación (2) del párrafo anterior, o de la situación (1) del párrafo que inmediatamente precede al anterior, situaciones que pueden presentarse en cualquier búsqueda a nivel único. Se presenta cuando la primera CK de un bloque siguiente (N+1) dentro del nivel en cuestión es la que da alta en comparación con el S.A. En este caso, la búsqueda a nivel único se termina almacenando la CPU dos indicaciones, que son la última del bloque anterior (el N) y la primera del bloque en curso (el N+1).  
25 Puede entonces usarse cada indicación para iniciar una búsqueda a nivel múltiple en los niveles inferiores y recuperar un bloque respectivo del nivel de datos (IO). Una  
30



comparación de verificación del argumento de búsqueda (S.A.) con el campo de claves de cada bloque de LO recuperado determina cuál es el correcto. Naturalmente, si el primer bloque de LO recuperado es el que da la verificación correcta, se hace innecesaria la búsqueda a nivel múltiple usando la segunda indicación.

En lo que sigue se da una tabla resumen de las condiciones de señalización al final de un bloque de investigación en curso:

10 TABLA RESUMEN

CONDICION DE LEC TURA DE INDICA- CION	SENALES INDICATI VAS DE LA CONDI- CION	CONDICION DE FI- NAL DE BUSQUEDA
(1) Ultima indi- cación	C.E. y D.E.	Leer bloque si- guiente
(2) Indicación intermedia	S.M. y C.E. y D.E.	Indicación en curso correcta
(3) Primera indi- cación	ATTN y C.E. y D.E.	Indicación en curso correcta
20 (4) Dos indica- ciones ambi- guas	S.M. y ATTN y C.E. y D.E.	Indicación en curso o última correcta
(5) Final de ar- chivo (no hay 25 indicación en curso)	U.E. y C.E. y D.E.	Ultima indica- ción correcta

Búsqueda secuencial a un solo nivel

La búsqueda secuencial a nivel único se ini-  
cia por la etapa 910 (figura 14D) de igual modo que se em-  
pieza una búsqueda a nivel múltiple: por ejemplo, accio-



1.  
nando un pulsador de puesta en marcha, o bien ejecutando una instrucción de enlazar. La iniciación de una búsqueda selecciona la unidad de control y activa la línea 512F, reponiendo el circuito de báscula 660a de primer bloque, de la figura 19.

5  
10  
15  
20  
Sigue la etapa 911 de la manera ya descrita, excepto en que la indicación para cada iteración de la etapa 911 durante una búsqueda secuencial a nivel único se obtiene de la tabla de indicaciones de la figura 7C, para el nivel único seleccionado. La primera indicación de la tabla seleccionada tiene acceso al primer bloque del nivel del índice a investigar. El orden de las indicaciones de cada tabla de ellas representa la secuencia de ordenación de los bloques dentro de su nivel respectivo, aunque la localización física de estos bloques pueda estar en cualquier parte (siempre que se tenga acceso a ella por medio de las indicaciones de la tabla). Por lo tanto, la etapa 911 de la figura 14D responde a la etapa 910 con la acción de búsqueda y emisión, por parte de la CPU, de una orden de "Inscribir argumento de búsqueda y primera indicación", para ir a buscar la primera indicación de la tabla de ellas seleccionada en la figura 7C.

25  
A continuación se ejecuta la etapa 912, por transferencia del argumento de búsqueda y de esa indicación desde la CPU al registro de S.A. y al registro de indicaciones de la memoria baja 10.

La etapa 913 genera entonces una señal de C.E. y D.E.

30  
La etapa 940 responde yendo a buscar una or



den de "búsqueda a nivel único", como orden siguiente de la tabla de órdenes de la figura 7A, en una secuencia de órdenes prefijada para una operación de búsqueda a un solo nivel.

5                    Sigue la etapa 941, con acceso del dispositivo 530c de I/O al bloque definido por el contenido en curso del registro de indicaciones de la memoria baja 10 (de la manera ya descrita para la etapa 921). A continuación, en la etapa 941a, se repone el contador de igualdad 301 de la figura 13, en preparación de la búsqueda -  
10                    en el bloque; y se ejecuta la etapa 942 de la misma manera ya descrita para la etapa 922, para leer el bloque en curso al que se ha tenido acceso en la etapa 941.

                  Acto seguido, en una etapa de final de archivo (EOF) 943 se efectúa una verificación o comprobación para ver si el bloque que en ese momento se está leyendo ha señalado EOF. Si no es así, se toma la salida -  
15                    D12 a la etapa 840 (figura 14A) a fin de investigar el bloque en curso que está siendo leído en serie por la etapa 942. Pero, de existir EOF, la etapa 943a señala U. E., y se termina la operación de búsqueda a nivel único. En esta última situación, la indicación correcta es la -  
20                    última de la lectura de bloque anterior pasada a la CPU, para obtener la condición )5) de la Tabla resumen precedente.  
25

                  Al final de la búsqueda en serie en cualquier bloque, en curso, (sin haber EOF), por el método -  
de las figuras 14A, 14B y 14C, se toma la salida por C10 (figura 14C) a la etapa 930 de la figura 14D, para verificar si la orden en curso es o no de "buscar". Si no lo  
30

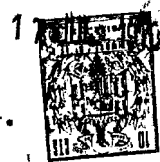


es, se trata de una orden de "buscar a nivel único", que es la otra de las dos opciones ofrecidas por la etapa 930. Para la orden de "buscar a nivel único", se sale por D11 para entrar en la etapa 926 de la figura 14E.

5 El método de la figura 14E busca las condiciones de primera y última CK que pueda haber en el bloque de lectura en curso, incluido el caso especial antes citado en relación con la figura 14G. En la etapa 926 se determina si se toma por lectura la última indicación del  
10 bloque en curso y la etapa 928 determina si se lee la primera indicación del bloque actual. Hay que efectuar dos iteraciones de entrada por D11 (una por cada uno de dos bloques secuenciales) antes de poderse determinar tanto si la primera CK está alta para el bloque en curso, como  
15 si la última CK dio alta para el bloque anterior.

Siempre que la indicación de toma por lectura provenga de una CK alta de dentro del bloque, es decir, no resulte de ser alta la primera o la última CK, se pasa de la etapa 928, por la salida E12, a la etapa 956  
20 de la figura 14F. En la etapa 956 se transfiere a la CPU 511 el contenido del registro de indicaciones o "pointers" de la memoria baja 10, y en la etapa 957 se señala S.I. y C.E. y D.E. para dar fin a la operación de búsqueda a nivel único representada por la condición (2) de la prece  
25 dente Tabla resumen.

Si en la etapa 926 se encuentra que se había tomado por lectura la última indicación del bloque en curso, se toma la salida por E13 a la etapa 950 (figura  
30 14F). La etapa 950 responde transfiriendo a la CPU la última indicación proveniente del registro de indicaciones



de la memoria baja 10, como se indica en la figura 14G.  
A continuación se ejecuta la etapa 951, señal C.E. y D.  
E., que aparece sola para todos y cada uno de los bloques  
que no tengan una CK alta en comparación con el S.A.; es-  
to es en los que solo dé alta la última CK(P=0) artifi-  
cial. Esto satisface la condición (1) de la Tabla resumen  
precedente.

Se entra a continuación en la etapa 952,  
para determinar si la búsqueda a nivel único en curso es  
secuencial o binaria. Si es secuencial, se pasa a la eta-  
pa 953 que da acceso a la siguiente indicación de la ta-  
bla de indicaciones a nivel único, para obtener acceso al  
bloque siguiente. La salida por F10 da lugar a la reite-  
ración de las etapas 911-913 y 940-943 de la figura 14D,  
emitiéndose en la etapa 911 la orden de "inscribir argu-  
mento de búsqueda y primera indicación". La etapa 912 da  
por resultado la transferencia del mismo S.A. (la búsque-  
da no está terminada), y de la nueva indicación a que se  
ha tenido acceso por la etapa 953. En la etapa 940 se  
vuelve a emitir la orden de "buscar a nivel único", que  
da lugar a que se lea el bloque siguiente en la etapa  
942 (figura 14D). A continuación, de no llegarse a un  
final de archivo (EOF), se reitera el método de las fi-  
guras 14D, 14A, 14B y 14C para salir por C10 de la figu-  
ra 14C, volviendo a la etapa 930 de la figura 14E, y se  
toma la salida D11 a la etapa 926 de la figura 14E. Si  
lo que se toma en la lectura del bloque en curso es su  
primera indicación, la etapa 926 responde entonces nega-  
tivamente, y se pasa a la etapa 928. Si lo que se tomó  
del bloque anterior fué su última indicación, ello no



tiene efecto en la etapa 926 durante el bloque en curso.

Así, en la etapa 928 se detecta si lo que se toma del bloque en curso es la primera indicación. Si es así, se toma la salida E14 a la etapa 958 de la figura 14E, para determinar si el bloque en curso es el primero del índice de un solo nivel. Caso afirmativo, la primera indicación es realmente la correcta, y se pasa a la etapa 960 para tomarla del registro de indicaciones de la memoria baja 10 y llevarla a la CPU. Se señalan luego, en la etapa 961, las condiciones de ATTN, y C.E. y D.E., y la etapa 962 indica negativamente la existencia de una búsqueda secuencial. Sigue la etapa 964 para indicar que no se había dado la señal de S.M., de modo que la CPU sabe o conoce ya la existencia de la condición (3) de la Tabla resumen precedente. Se da fin entonces a la búsqueda a nivel único.

En cambio, si de la etapa 958 se sale con una negación, la etapa 959 da la señal de S.M., indicando que el bloque en curso no es el primero del índice. La indicación correcta es entonces o la primera de la lectura en curso, o bien la última del bloque anterior. A continuación, en la etapa 960 se toma la primera indicación o "pointer" de la memoria baja 10 y se lleva a la CPU. La señal de S.M. hace que la CPU guarde la nueva indicación en la memoria 510 (por ejemplo, en CPU Reg-2). La etapa 961 da entonces las señales de ATTN y C.E. y D.E. La CPU tiene entonces información suficiente, por las etapas 962 y 964, para ejecutar la etapa 963, y se satisface entonces la condición (4) de la Tabla resumen. La operación a nivel único se termina en la etapa 964, y la



GPU puede ejecutar la etapa 963 iniciando una búsqueda a nivel múltiple como se ha descrito anteriormente, para cada una de las dos indicaciones o "pointers", a fin de determinar cuál es la adecuada.

5                   En las figuras 17 a 19 inclusive se incluye un grupo particular de circuitos para obtener el método secuencial de las figuras 14D a 14F. Los circuitos para las etapas 910-913 se han descrito ya al hablar de las operaciones de búsqueda a nivel múltiple. Los circuitos para las etapas 940-943 son iguales a los circuitos para las etapas 920-923, con la salvedad de que se activa la línea 513K de descodificador, en lugar de la línea 513J. La línea 530E de EOF está reservada en la figura 19 para señalar U.E. (en la etapa 943a de la figura 14D), lo que da por resultado un final de búsqueda secuencial a nivel único, y satisface la condición (5) de la precedente Tabla resumen.

15                   En la figura 18, se da una señal de C.E. y D.E. procedente del circuito de báscula 649 cuando éste es puesto en acción sea (1) por la línea 530E de EOF, sea (2) al final de una transferencia de indicación a la memoria baja 10. Esta señal de C.E. y D.E. indica a la CPU que ha de seleccionar la orden siguiente de la tabla de órdenes de la figura 7A, en función de las condiciones existentes.

20                   Al final de una transferencia de indicación se pone en acción el circuito de báscula 649 por activación de un circuito de coincidencia 648 cuando la línea 666C es activada por el descodificador 666 de la figura 19, al ser seleccionado para acceso al último byte del



registro de indicaciones de la memoria baja 10 durante  
la señal de orden de "buscar a nivel único" en curso, pre-  
sente en la línea 513K que va al circuito disyuntivo 648a.  
Después de aceptadas por la CPU la indicación transferida  
5 y las señales de estado, la CPU contesta por el enlace  
con señales en la línea 513D de aceptación de estado, que  
reponen el circuito de báscula 649.

La señal de C.E. y D.E. puede presentarse  
sola o en combinación con las de ATTN y/o S.M., al trans-  
10 ferirse una indicación a la CPU para completar la ejecu-  
ción de una orden de búsqueda a nivel único. La particu-  
lar combinación de estas señales indica a la CPU cómo ha  
de manejar la indicación recibida. La Tabla resumen pre-  
cedente explica el significado de cada una de estas seña-  
15 les para la CPU. A continuación puede enviarse a la CPU  
la programación normal (no descrita en la presente) para  
manipular el almacenaje en el acceso de la indicación o  
las indicaciones a la memoria 510.

Así, tras cada transferencia de indicación  
20 a la CPU se da una señal de C.E. y D.E. Si al final de  
esta orden a nivel único no se da señal de ATTN, S.M. o  
U.E., la señal de C.E. y D.E. aparece sola, lo que sig-  
nifica para la CPU que no se ha reconocido como adecuada  
la indicación, y que ha de continuar la búsqueda secuen-  
25 cial con la lectura del siguiente bloque de la secuen-  
cia.

Ahora bien, si la etapa 961, la 959 o la  
943a dan también la señal de ATTN, la de S.M. o la de  
U.E., esta señal (y no la concurrente de C.E. y D.E.)  
30 es la que controla la CPU en lo que se refiere a la in-



17  
dicación transferida. La señal de C.E. y D.E. indica a  
la CPU que la indicación transferida era la última del  
bloque en curso, y que ha de leerse el bloque siguiente.  
La señal de U.E. indica a la CPU un final de archivo  
5 (EOF) y que la indicación ultimamente transferida es la  
última indicación del índice de nivel único y, por lo tan-  
to, es la correcta. Las señales de S.M. y de C.E. y D.E.  
indican que la indicación o "pointer" tomada por lectura  
es correcta como indicación intermedia. Cuando se trans-  
miten ambas señales de S.M. y ATTN, indican a la CPU el  
10 caso especial ilustrado en la figura 14G, en el que la  
indicación transferida es la primera del bloque, y en el  
que bien ella o bien la última indicación del bloque con-  
tiguo precedente es la correcta. La CPU debe entonces  
15 iniciar la determinación.

La figura 19 muestra los circuitos para ge-  
nerar las señales de ATTN, S.M. y U.E. Son respectivamen-  
te generadas por un circuito de coincidencia 664b, un  
circuito disyuntivo 664c y una línea 530K. Las señales  
de ATTN y S.M. son controladas por unos circuitos de coin-  
20 cidencia 660, 664d, y 664b, condicionados cada uno por  
activación de la línea 513K y de la línea 666C de final  
de registro de indicaciones.

El circuito de coincidencia 664d genera la  
25 señal de S.M. para la etapa 957; el circuito de coinci-  
dencia 664a genera la señal de S.M. para la etapa 959;  
el circuito de coincidencia 664b genera la señal de ATTN  
para la etapa 961; y la línea 530K genera la señal de  
U.E. para la etapa 943A, partiendo de la señal de EOF.

30 El circuito de coincidencia 660 es activado

**380987**



por la puesta en acción del circuito de báscula 661 de  
 última indicación por parte de la línea 332A, que se ac-  
 tiva desde la figura 13 cuando el bloque en curso da en  
 lectura su última indicación o "pointer", es decir, da  
 5      alto en F=0. El circuito de báscula 661 es repuesto por  
 la activación en CPU de la línea 512D de aceptación de  
 estado. Al habilitarse el circuito de coincidencia 660,  
 activa un circuito de báscula 660a que indica no tratar-  
 se de un primer bloque. El circuito de báscula 660a es  
 10     inicialmente repuesto por activación de la línea de ini-  
 ciación al principio de una búsqueda en el nivel único  
 en cuestión. El circuito de báscula 661 es activado por  
 la señal de CK para F=0, con la última indicación del  
 primer bloque, si ninguna CK anterior dio alta en compa-  
 15     ración con el S.A. Esta activación del circuito de báscu-  
 la 660a indica que a partir de entonces ninguno de los  
 bloques individuales de ese nivel único en cuestión es  
 el primer bloque.

Un circuito de coincidencia 664a genera  
 20     una señal de S.M. mediante activación concurrente del  
 circuito de báscula 660a que indica no tratarse del pri-  
 mer bloque, y de un circuito de coincidencia 664b de pri-  
 mera indicación. El circuito de coincidencia 664b está  
 condicionado por la activación de un circuito de báscula  
 25     663 de selección de primera indicación, que es puesto en  
 acción por la activación de un circuito de coincidencia  
 662. Un circuito de báscula 665 de primera indicación  
 controla la activación del circuito de coincidencia 662.  
 El circuito de báscula 665 se activa, al principio del  
 30     bloque de lectura en curso, por su señal de reloj, de



byte denotativo de longitud de indicación, RL CY; y es  
reque<sup>17</sup>sto por la señal de reloj de lectura de primera in-  
dicación, R CY. Por consiguiente, el circuito de báscula  
665 está activado durante el tiempo que se está comparan-  
do la primera CK del bloque con el S.A. Si dicha primera  
5 CK da alta en la comparación, se activa un circuito de  
báscula 328 de selección de indicación (figura 13) que  
pone en acción la línea 328A de selección de R que va al  
circuito de coincidencia 662 de la figura 19. Por tanto,  
10 la activación del circuito de coincidencia 662 indica que  
se va a seleccionar la primera indicación (que está a  
punto de leerse); y pone en acción el circuito de báscu-  
la 663 de selección de primera indicación; se activa lue-  
go el circuito de coincidencia 664b, al final de la trans-  
15 ferencia de la indicación al registro de indicaciones  
de la memoria baja 10, por activación de la línea 666C.

El circuito de coincidencia 664a, por lo  
tanto, es habilitado por la activación del circuito de  
coincidencia 664b, cuando el bloque en curso no es el  
20 primero del nivel único en cuestión. El hecho de no ser  
el bloque en curso un primer bloque viene indicado por  
la lectura de la última indicación del bloque anterior,  
que activa el circuito de báscula 660a indicador de no  
tratarse del primer bloque.

25 Por consiguiente, el circuito de coinciden-  
cia 664a en-vía la señal de S.M. por medio del circuito  
disyuntivo 664c a la línea 664B, al ponerse en acción en  
estas condiciones en la etapa de señal 959, de manera que  
la CPU tiene la información de que debe examinar las dos  
30 indicaciones ultimamente transferidas a ella.



Una señal presente en la línea 530E que viene del control 530 de I/O de la figura 16, indica que el bloque en curso representa un final de archivo (EOF). Esta señal se pasa a la línea 530K que va a los controles de enlaces 512 (figura 8) para señalar a la CPU que la última indicación del bloque anterior era la última indicación del índice de nivel único y, por tanto, es la indicación correcta.

#### Búsqueda binaria a nivel único

Otra búsqueda alternativa realizada en los bloques de un solo nivel puede hacerse recurriendo a la "búsqueda binaria". Se emplea para ello también la tabla de indicaciones de la figura 7C, para el particular nivel que se esté investigando. Como antes se ha dicho, cada tabla de indicaciones tiene sus indicaciones dispuestas en la secuencia de clasificación de los bloques representados por la indicación. Los programas para la investigación o búsqueda binaria en tablas son ya conocidos en la técnica de las calculadoras; por tanto, la búsqueda binaria no es sino un elemento ya conocido, usado por esta invención. Así, pues, la búsqueda binaria no se estudia en profundidad aquí.

La búsqueda binaria da principio también con la etapa de iniciación 910 de la figura 14D. La etapa 911 es ejecutada por la orden de "inscribir argumento de búsqueda y primera indicación", que elige de la tabla la indicación de enmedio (o la inmediata, si la tabla tiene un número par de indicaciones). En la etapa 912 se carga el registro de indicaciones, y el registro de argumentos de búsqueda, de la memoria baja 10, con esa indi-

17 JUN



cación y con el argumento de búsqueda requerido, respectivamente.

5 En la etapa 940 se emite entonces una orden de "buscar a nivel único"; y en la etapa 941 se obtiene acceso a ese bloque. En la etapa 941a se repone el contador de igualdad con argumento de búsqueda, y en la etapa 942 se toma el bloque para su investigación, llevándolo por la salida D12 a la figura 14A. La etapa 943 se salta u omite en una búsqueda binaria, porque ninguna de las  
10 indicaciones seleccionadas representa el final del bloque de archivo.

Después de investigado el bloque por las etapas de las figuras 14A a 14C inclusive, se entra por C10 en la etapa 930 de la figura 14D y se sale de ella por D11, ya que no se está usando la orden de "buscar".  
15 Se entra así en la etapa 926 de la figura 14E.

En la figura 14E se opera, para la búsqueda binaria, de la misma manera que para la búsqueda secuencial.

20 Siempre que la indicación de lectura en curso sea una indicación intermedia (es decir, ni la primera ni la última del bloque en curso), la indicación de lectura es correcta, y la búsqueda finaliza entrándose por E12 en la figura 14F, en la que se opera como antes se ha descrito para la búsqueda secuencial,. Se satisface así la condición (2) de la Tabla resumen precedente.  
25

Ahora bien, si por el bloque en curso es leída la indicación primera o última (condición (1), (3) o (4) de la Tabla resumen) se toma una de las entradas  
30

11-7-70

- 63 -

**380987**



(E14 o E13, respectivamente) a la figura 14F, para conti-  
nuar la búsqueda binaria. Una toma o lectura de primera  
indicación da por resultado las señales de APTW y de C.E.  
y D.E. Una toma o lectura de última indicación da lugar  
a la señal de C.E. y D.E. sola.

5

La lectura en curso de la primera o la última  
indicación da a elegir entre las dos iniciaciones de  
que se disponga en el siguiente nivel de la búsqueda bi-  
naria. El algoritmo ya conocido de la búsqueda binaria  
elige el par de indicaciones de que se dispone a continua-  
ción. Una lectura de primera indicación elige la inmedia-  
tamente disponible en la parte superior de la tabla, para  
seleccionar un bloque inferior (que está más bajo) en el  
índice; una lectura de última indicación elige la inmedia-  
tamente disponible en la parte inferior de la tabla para  
seleccionar un bloque superior (que está más alto) en el  
índice.

10

15

El procedimiento se repite respecto al bloque  
seleccionado siguiente, que se convierte entonces en blo-  
que en curso. Cada nivel siguiente de la búsqueda binaria  
va estrechando o reduciendo el máximo número de bloques  
que todavía quedan por investigar. Este número máximo es  
de  $1/2^M$  del número total de bloques del nivel único que  
se esté investigando en binario, siendo M el nivel bina-  
rio en curso de la búsqueda. Así, M es la unidad al prin-  
cipio de la búsqueda binaria, y va aumentadndo en uno ca-  
da vez que se tiene acceso a unanueva indicación para la  
búsqueda binaria. Así, el número máximo de bloques del  
nivel único en cuestión que requieren nueva investigación  
disminuye exponencialmente, al aumentar el nivel binario

25

30

M.

17 JUN



Si en la etapa 926 se halla que la última indicación del bloque en curso ha sido leída, se pasa desde ella por E13 a la etapa 950 (figura 14F). La etapa 950 responde transfiriendo la última indicación desde el registro de indicaciones de la memoria baja 10 a la CPU, como se indica en la figura 14G. Se ejecuta luego la etapa 951 de C.E. y D.E., que tiene lugar sola por cada bloque que no tenga una CK alta en comparación con el S.A., es decir, cuya última CK, artificial y de P=0, sea la última que de alta en la comparación. Esto satisface la condición (1) de la Tabla resumen precedente.

Se entra a continuación en la etapa 952, para determinar si la búsqueda en curso a nivel único es secuencial o binaria. Si es binaria, se entra en la etapa 954a, para que la CPU determine si el bloque en curso (representado por la indicación binaria en curso) es el último del árbol de búsqueda binaria en curso. En general, existe un número fijo  $M_t$  de niveles binarios en todo árbol de búsqueda aplicado a una tabla de un tamaño particular en la figura 7C. Existen asimismo métodos ya conocidos para dar fin a una búsqueda binaria. Por ejemplo si la indicación en curso es la última de un total de  $M_t$  indicaciones usadas para investigar con el argumento de búsqueda en curso, de la etapa 954a se sale a la etapa 955 para determinar cualquier condición especial de doble ambigüedad indicada en la figura 14G.

Si en la etapa 954a se halla que en el árbol binario en curso se han leído menos de  $M_t$  bloques, se pasa a la etapa 954b para tener acceso a la indicación



inmediata superior (puesta a disposición por el programa de búsqueda en tabla binaria), que es superior a la indicación en curso en la tabla de indicaciones. De la etapa 954b se sale por F10.

5 Por tanto, una búsqueda binaria en un nivel cualquiera del índice de varios niveles exige un número de entre una y  $M_t$  (inclusive) búsquedas de bloque, para hallar una indicación correcta; y  $M_t$  es igual al logaritmo de base dos del número total de bloques de ese nivel, 10 es decir, a  $\log_2 T$ . Por comparación, el número máximo de investigaciones de bloque necesarias en la búsqueda secuencial de un único nivel está comprendido entre la unidad y el número total de bloques, T, al nivel en cuestión. Por tanto, la búsqueda binaria, por término medio, es 15  $T/M_t$  veces más rápida que la búsqueda secuencial, para la investigación completa de un solo nivel.

La salida por F10 trae como consecuencia la reiteración de las etapas -911-913 y 940-943 de la figura 14D, emitiéndose en la etapa 911 la orden de "inscribir ar 20 gumento de búsqueda y primera indicación". La etapa 912 da por resultado la transferencia del mismo S.A. (la búsqueda no se ha terminado) y de la nueva indicación a la que se ha tenido acceso en la etapa 953. En la etapa 940 se vuelve a emitir la orden de "buscar a nivel único", 25 que da lugar a que en la etapa 942 (figura 14D) se lea o tome el bloque siguiente. A continuación, de no llegarse a un final de archivo (EOF), el método de las figuras 14D, 14A, 14B y 14C se reitera con salida por C10 de la figura 14C, que vuelve a la etapa 930 de la figura 14E, 30 tomándose la salida D11 a la etapa 926 de la figura 14E.



Si la lectura de bloque en curso es su primera indicación o "pointer", en la etapa 926 se responde entonces negativamente, y se pasa a la etapa 928. La salida por lectura de la última indicación del bloque anterior no produce efecto en la etapa 926, durante el bloque en curso.

Así, en la etapa 928 se detecta si del bloque en curso se ha tomado por lectura la primera indicación. Si es así, se toma la salida E14 a la etapa 958 de la figura 14E. La determinación, en la etapa 958, de si el bloque en curso es el primero del índice a nivel único, es ignorada en esta búsqueda binaria; es decir, se hace caso omiso de si en la etapa 959 se ha generado una señal de S.M., ya que no importa para la búsqueda binaria que se tome la salida de la etapa 958 para llegar a la etapa 960, en la que se lee la indicación tomada del registro de indicaciones de la memoria baja 10, y se lleva a la CPU.

A continuación, en la etapa 961 se dan las señales de ATTN y C.E. y D.E., y la etapa 962 debe iniciar positivamente la existencia de una búsqueda binaria. Sigue la etapa 966, que es ejecutada de manera idéntica a la etapa 954a para determinar si el bloque en curso es el último del árbol de búsqueda binaria en curso. Si no lo es, se entra en la etapa 967 para tener acceso a la indicación inmediata inferior (puesta a disposición por el programa de búsqueda en tabla binaria), que es inferior o más baja que la indicación en curso de la tabla de indicaciones. Se toma luego la salida por F10 para buscar en el bloque a nivel único siguiente.

Si se sale de la etapa 966 a la etapa 955,



se obtiene como resultado la operación anteriormente descrita para la salida en caso afirmativo de la etapa 954a.

5 En las figuras 17 a 19 se incluye un grupo particular de circuitos para obtener el método de búsqueda binaria de las figuras 14D a 14F inclusive. Los circuitos para las etapas 910-913 se han descrito anteriormente, en relación con las operaciones de búsqueda a nivel múltiple. Los circuitos para las etapas 940-943 son  
10 iguales a los de las etapas 920-923, con la excepción de que la línea de descodificador 513K es la que se activa, en lugar de la línea 513J.

En la figura 18, se obtiene una señal de C.E. y D.E. del circuito de báscula 649, al ser éste  
15 puesto en acción al final de una transferencia de indicación a la memoria baja 10. (La condición de final de archivo no se usa durante una búsqueda binaria.)

La señal de C.E. y D.E. indica a la CPU que debe seleccionar la orden siguiente de la tabla de órdenes de la figura 7A, en función de las condiciones  
20 existentes.

El circuito de báscula 649 es puesto en acción al final de una transferencia de indicación, por activación de un circuito de coincidencia 648 cuando la línea 666C es activada por el descodificador 666 de la figura 19, al obtenerse acceso al último byte en el registro de indicaciones de la memoria baja 10, durante la señal de orden de "buscar a nivel único" en curso en la línea 513K que va al circuito disyuntivo 648a. Después de  
25 haber aceptado la CPU las señales de indicación y de es-  
30





indica que en el bloque en curso se tomó por lectura la primera indicación; y que hay que leer un bloque inferior del índice, ya que la primera indicación no es necesariamente la correcta en una búsqueda binaria. Este bloque siguiente se elige de entre los dos bloques (uno superior y otro inferior al bloque en curso) puestos a disposición por el programa de búsqueda en tabla binaria anteriormente estudiado.

La figura 19 ilustra los circuitos para generar la señal de ATTN. Es generada por el circuito de coincidencia 664b como se describe más arriba bajo el epígrafe de "Búsqueda secuencial a nivel único".

#### Búsqueda combinada, binaria y secuencial

Una búsqueda a nivel único puede a veces efectuarse con mayor eficacia combinando los métodos de investigación binario y secuencial. Debido a la naturaleza exponencial de la búsqueda binaria, es ésta particularmente eficaz durante sus niveles iniciales para reducir y estrechar el número máximo restante de bloques del nivel único en cuestión que puedan necesitar investigarse. Hacia el final de la búsqueda binaria, puede ésta no ser más eficaz que la búsqueda secuencial. En tal caso, la búsqueda binaria puede terminarse en un nivel binario B que tenga un valor inferior al número máximo M de niveles binarios de un árbol de búsqueda. El valor de M depende del número particular de indicaciones que haya en la tabla de indicaciones en curso. El cambio de búsqueda binaria a búsqueda secuencial puede estar controlado por la CPU, al percibir ésta que se han terminado B niveles de búsqueda en binario. Entonces se determina la fracción

380987



restante del índice que necesita ser investigada, y esta  
determinación se hace por medio de la indicación o "poin-  
ter" seleccionada por la etapa 954b o 967 durante la bús-  
queda en el último nivel binario B. Esta indicación sitúa  
5 el lugar central o de enmedio de la fracción restante del  
índice, que se investigará secuencialmente. La restante  
fracción es  $1/2^B$  del número de bloques del nivel de índi-  
ce, que es igual a  $1/2^B$  del número total T de indicaciones  
contenidas en la tabla de indicaciones en curso; es decir,  
10 a  $T/2^B$ .

Por lo tanto, la búsqueda secuencial empe-  
zará con una indicación que está a la mitad del número  
 $T/2^B$  de indicaciones por encima de la indicación o "poin-  
ter" últimamente definida por la etapa 954b o 967; esto  
15 es, a  $T/2^{B+1}$  indicaciones por encima de la de búsqueda  
binaria seleccionada por la etapa 954b o 967 durante la  
búsqueda del último nivel binario B.

Otro uso para la búsqueda combinada se ha-  
lla en la opción (1) de la etapa 955 de la figura 14F.  
20 Aquí la búsqueda binaria ha completado su último nivel  
binario M al final de su árbol de búsqueda, como lo indi-  
ca la salida en sentido afirmativo de la etapa 954a o  
966. La etapa 955 sólo se necesita en el caso especial  
en que el último bloque de búsqueda binariaa haya dado  
25 por lectura su primera o su última indicación. La opción  
(1) inicia entonces una búsqueda secuencial para obtener  
ambas indicaciones para la situación ilustrada en la fi-  
gura 14G. Si la última indicación de lectura fué la pri-  
mera del último bloque investigado en binario, se inicia  
30 una búsqueda secuencial en dos bloques, con la indicación

380987



anterior contigua de la tabla de indicaciones (respecto a la tabla de indicaciones para el último bloque investigado en binario).

5 Si la última indicación de lectura fué la última del último bloque investigado en binario, se iniciaba una búsqueda secuencial de un solo bloque, con la indicación inmediata contigua de la tabla de indicaciones (en relación también con la tabla de indicaciones para el último bloque investigado en binario).

10 La opción (2) para la etapa 955 ejecuta la misma función enviando información de búsqueda, en la que cada indicación de la tabla de indicaciones representa un bloque investigado con el mismo argumento de búsqueda. Por lo menos se dan tres bitios de envío con cada  
15 indicación, tales como un bitio de búsqueda efectuada, un bitio de primera o última indicación y un bitio de búsqueda terminada. La condición de doble indicación de la figura 14G puede detectarse entonces examinando los bitios enviados, para determinar el caso en que dos indicaciones  
20 contiguas, con sus bitios de búsqueda efectuada en activo, tengan también en activo sus bitios de primera o última indicación para indicar que una de éstas es la última, y la inmediata inferior de la tabla es la primera. Finaliza entonces toda continuación de la búsqueda.

25 La presente solicitud, que corresponde a la presentada en los Estados Unidos de América, el 26 de Junio de 1.969, bajo el N° 836.825, se acoge a los beneficios del artículo 51 del vigente Estatuto sobre Propiedad Industrial.

30

**380987**

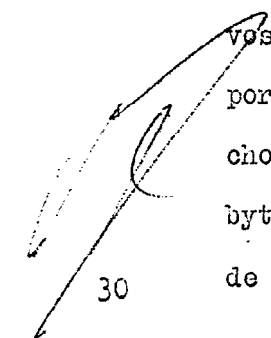


REIVINDICACIONES

5

Los puntos de invención propia y nueva que se presentan para que sean objeto de esta solicitud de Patente de Invención en España, por VEINTE años, son los siguientes:

10  
15  
20  
25  
30  
1.- Un método de buscar un argumento de búsqueda en un índice de varios niveles que tiene un formato de claves comprimidas de nivel alto en el que hay un par de claves comprimidas asociado a una indicación respectiva, y que tiene un formato de claves comprimidas de nivel bajo en el que hay una sola clave comprimida asociada a una indicación correspondiente, método que comprende las etapas de: seleccionar a máquina un bloque inicial de nivel alto en dicho índice, como primer bloque comprimido de la búsqueda; reponer a máquina al estado inicial un contador de igualdad de argumento de búsqueda, antes de investigar en un bloque cualquiera de dicho índice; tomar por lectura a máquina dicho bloque de nivel alto; comparar a máquina el argumento de búsqueda con los respectivos pares de claves comprimidas de dicho bloque de cima, por el orden de su secuencia; incrementar a máquina dicho contador de igualdad hasta indicar la posición de byte actual (indicada en un momento dado) que se busca, de dicho argumento de búsqueda, al dar cada byte de clave



11-7-70

380987



17

la igualdad de comparación con el byte actual investigado de dicho argumento de búsqueda; indicar a máquina la posición de byte de orden inmediato inferior de dicho argumento de búsqueda como posición de byte actual investigado, - en respuesta a dicha etapa de incrementar a máquina; recuperar a máquina la indicación asociada a un par de claves comprimidas del cual cada una de las claves comprimidas - dé un nivel alto en comparación con el byte actual del argumento de búsqueda; y transferir a máquina a una unidad central de tratamiento la indicación obtenida por dicha etapa de recuperar a máquina, para obtener acceso al dispositivo de memoria al que se dirige dicha indicación, para tener acceso al bloque de índice de nivel inmediato inferior, con lo cual dicha indicación se dirige a un bloque del nivel inmediato inferior de dicho índice.

2.- El método de la reivindicación 1, en el cual dicha etapa de seleccionar a máquina elige un bloque de cima del índice de varios niveles, como bloque inicial de nivel alto citado.

3.- El método de la reivindicación 1, en el cual dicha etapa de seleccionar a máquina elige un bloque cualquiera no cimero del índice de varios niveles, como bloque inicial de nivel alto citado, con lo cual una búsqueda a nivel múltiple inferior empieza con dicho bloque no cimero.

4.- El método de la reivindicación 1, en el que dicha indicación obtenida por dicha etapa de transferir a máquina se dirige a un bloque de nivel alto del nivel inmediato inferior, método que comprende las etapas de: seleccionar a continuación a máquina el bloque al que

*[Handwritten signature]*  
11-7-70



se dirige dicha indicación; aplicar a máquina dichas etapas de reponer a máquina, tomar por lectura a máquina, comparar a máquina, incrementar a máquina, indicar a máquina, recuperar a máquina y transferir a máquina, al bloque obtenido por dicha etapa de seleccionar a continuación a máquina; con lo cual la indicación últimamente recuperada se dirige a un nivel inmediato inferior de dicho índice.

5

5.- El método de la reivindicación 1 o la 4, en el que dicha indicación obtenida por dicha etapa de transferir a máquina se dirige a un bloque de nivel bajo, método que comprende las etapas de: seleccionar a máquina adicionalmente el bloque a que se dirige dicha indicación; reponer a máquina dicho contador de igualdad de argumento de búsqueda, volviéndolo al estado inicial, antes de investigar en dicho bloque; tomar por lectura a máquina el bloque; comparar a máquina el argumento de búsqueda con las respectivas claves comprimidas de dicho bloque, por su orden de secuencia; incrementar a máquina dicho contador de igualdad hasta indicar la posición de byte actual del argumento de búsqueda, al producirse una igualdad de comparación cualquiera de un byte de clave con el byte actual de dicho argumento de búsqueda; indicar a máquina la posición de byte de orden inmediato inferior de dicho argumento de búsqueda como su posición de byte actual, en respuesta a dicha etapa de incrementar a máquina; y recuperar a máquina la indicación asociada a una clave comprimida que dé primero un nivel alto en dicho bloque en comparación con el byte actual del argumento de búsqueda, con lo cual la indicación últimamen-

10

15

20

25

30  
11-7-70



te recuperada se dirige para acceso al bloque de datos que se esté investigando con dicho argumento de búsqueda.

6.- Un método de buscar un argumento de  
5 búsqueda en un determinado nivel de un índice comprimido, usando una tabla de indicaciones con las direcciones de acceso de los bloques de dicho primer o determinado nivel puestas en secuencia por el orden de clasificación de dichos bloques, método que comprende las etapas de: ordenar  
10 a máquina la selección de las indicaciones de dicha tabla, efectuándose cada selección sucesiva de una indicación en respuesta a un determinado tipo de señal de posición de indicación; seleccionar a máquina el bloque al que se dirige una primera indicación elegida de dicha tabla de  
15 indicaciones; reponer a máquina un contador de igualdad de argumento de búsqueda, llevándolo al estado inicial antes de investigar en dicho bloque; tomar por lectura a máquina dicho bloque obtenido por dicha etapa de seleccionar a máquina; comparar a máquina el argumento de búsqueda  
20 con las respectivas claves comprimidas de dicho bloque, por el orden de su secuencia; incrementar a máquina dicho contador de igualdad hasta indicar la posición de byte actual del argumento de búsqueda, al producirse una comparación de igualdad de cada byte de clave con el byte  
25 actual de dicho argumento de búsqueda; indicar a máquina el byte de orden inmediato inferior de dicho argumento de búsqueda como su byte actual, en respuesta a dicha etapa de incrementar a máquina; recuperar a máquina la indicación asociada a cualquier clave comprimida de dicho bloque que dé primero un nivel alto de comparación

30

11-7-70



con el byte actual del argumento de búsqueda; y señalar a máquina la posición de las indicaciones con señales que relacionen la indicación obtenida por dicha etapa de recuperar a máquina con el lugar que ocupa la indicación en su bloque.

5

7.- El método de la reivindicación 6, en el que dicha etapa de señalar a máquina incluye el recurso de generar a máquina una señal de última posición, para indicar el momento en que la indicación obtenida por dicha etapa de recuperar indicaciones es la última indicación de su bloque.

10

8.- El método de la reivindicación 6, en el que dicha etapa de señalar a máquina incluye el recurso de generar a máquina una señal de posición intermedia, para indicar que la indicación obtenida por dicha etapa de recuperar indicaciones está comprendida entre la primera indicación y la última de dicho bloque.

15

9.- El método de la reivindicación 6, en el que dicha etapa de señalar a máquina incluye el recurso de generar a máquina una señal de primera posición, para indicar que la indicación obtenida por dicha etapa de recuperar indicaciones es la primera indicación de un bloque de dicho índice.

20

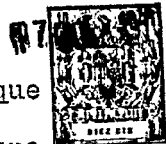
10.- El método de la reivindicación 6, en el que dicha etapa de señalar a máquina incluye el recurso de generar a máquina una señal de primera posición, para indicar en qué momento la indicación obtenida por la etapa de recuperar a máquina es la primera indicación de su bloque, y su bloque no es el primer bloque de su nivel.

25

30

11-7-70

380987



11.- El método de la reivindicación 6, que comprende las etapas de: seleccionar a máquina el bloque sucesivo, al que se dirige una indicación sucesiva seleccionada de la tabla de indicaciones, en respuesta a dicha etapa de ordenar a máquina que responde a la señal de posición de indicación procedente de dicha etapa de señalizar a máquina; y repetir a máquina dichas etapas de responder a máquina, tomar por lectura a máquina, comparar a máquina, incrementar a máquina, indicar a máquina, recuperar a máquina y señalizar a máquina para dicho bloque al que últimamente se ha dirigido dicha etapa de seleccionar a máquina.

12.- El método de búsqueda de la reivindicación 11, que comprende las etapas de: detectar a máquina una determinada forma o secuencia de dichas señales de posición de indicación, para indicar una señal de terminación de búsqueda; y reiterar a máquina dicha etapa de repetir a máquina, hasta que dicha etapa de detectar a máquina indique dicha señal de terminación de búsqueda.

13.- El método de la reivindicación 11, en el que dicha etapa de ordenar a máquina selecciona indicaciones, tomándolas de dicha tabla de indicaciones, en la secuencia de clasificación de los bloques de dicho primer nivel, con lo cual se obtiene una búsqueda secuencial en un solo nivel.

14.- El método de la reivindicación 11, en el que dicha etapa de ordenar a máquina elige indicaciones, tomándolas de dicha tabla de indicaciones, en una secuencia de búsqueda binaria capaz de responder a dicha

30  
11-7-70



etapa de señalizar a máquina.

15 5      15.- El método de la reivindicación 12, que comprende además el recurso de: finalizar a máquina dicha búsqueda en dicho primer nivel, en respuesta a la obtención de dicha señal de posición intermedia en dicha etapa de generar a máquina, con lo cual la indicación últimamente recuperada se dirige para acceso al bloque inmediato requerido del nivel inmediato inferior.

10      16.- El método de la reivindicación 13, que comprende las etapas de: detectar a máquina un final de índice en dicho primer nivel; y finalizar la búsqueda en dicho primer nivel, en respuesta a dicha etapa de detectar a máquina, con lo cual una última indicación del último bloque secuencial contenido en el índice de dicho nivel se dirige para acceso a un bloque requerido del nivel inmediato inferior.

20      17.- Un método de buscar un argumento de búsqueda en un bloque cualquiera de nivel alto de un índice comprimido de varios o muchos niveles, método que comprende las etapas de: seleccionar a máquina un bloque de nivel alto cualquiera de dicho índice, en el que hay claves comprimidas en parejas con una indicación respectiva, para obtener acceso a un bloque correspondiente del nivel inmediato inferior de dicho índice; reponer a máquina un contador de igualdad de argumento de búsqueda, llevándolo al estado inicial, antes de investigar en dicho bloque; tomar por lectura a máquina dicho bloque; comparar a máquina el argumento de búsqueda con los respectivos pares de claves comprimidas de dicho bloque, por el orden de su secuencia; incrementar a máquina di-

17 JUL



cho contador de igualdad hasta indicar la posición inme-  
diata a la actual de byte de argumento de búsqueda al  
producirse cada igualdad de comparación de byte de clave  
con un byte actual de dicho argumento de búsqueda; indi-  
5 car a máquina la posición de byte de orden inmediato infe-  
rior de dicho argumento de búsqueda como su posición ac-  
tual de byte buscado, en respuesta a dicha etapa de in-  
crementar a máquina; y seleccionar a máquina toda indica-  
ción asociada a un par de claves comprimidas en el que  
10 cada clave comprimida del par dé un nivel alto de compa-  
ración con el byte actual investigado.

13.- Un método de buscar un argumento de  
búsqueda en un nivel cualquiera de un índice dotado de  
una tabla de indicaciones con las direcciones de acceso  
15 de los bloques de dicho nivel puestas en secuencia por  
el orden de clasificación de dichos bloques, método que  
comprende las etapas de: seleccionar a máquina dichos  
bloques en dicho nivel, para su investigación en un orden  
cualquiera prefijado; reponer a máquina un contador de  
20 igualdad de argumento de búsqueda, para investigar en un  
bloque cualquiera al que se ha tenido acceso por dicha  
etapa de seleccionar a máquina; y almacenar a máquina  
una indicación asociada a una clave cualquiera comprimida  
de un bloque investigado, que dé primero un nivel alto  
25 de comparación con dicho argumento de búsqueda.

19.- El método de la reivindicación 13, que  
comprende las etapas de: señalar a máquina el momento  
en que dicha indicación almacenada por dicha etapa de al-  
macenar a máquina está comprendida entre la indicación  
30 primera y la última de su bloque; y finalizar dicha bús-

11-7-70

380987



queda en respuesta a dicha etapa de señalar a máquina.

20.- El método de la reivindicación 18, que comprende otra etapa de señalar a máquina el momento en que dicha indicación guardada por dicha etapa de almacenar a máquina es la última indicación de su bloque, y seleccionar a máquina otro bloque del mismo nivel para su investigación, en respuesta a dicha otra etapa de señalar a máquina.

21.- Un método de buscar un argumento de búsqueda en un índice de varios niveles.

Tal y como se ha descrito en la Memoria que antecede, representado en los dibujos que se acompañan, y para los fines que se han especificado.

Esta Memoria consta de ochenta y una hojas escritas a máquina por una sola cara.

Madrid, 17 JUL 1970

P.A.

Alberto de Aizoburu  
Por Poder

14-7-70 RME

580987

380987

FIG. 1A

FIG. 1B

1	2	3	4	5	
0	0	0	0	0	—
A	B	C	0	0	R1
A	B	C	E	F	—
D	H	M	N	0	R2
D	I	0	0	0	—
M	A	P	0	0	R3

P1	K1	P2	K2	
1	A	4		R1
1	D	2		R2
1	M	0		R3

FIG. 2 A

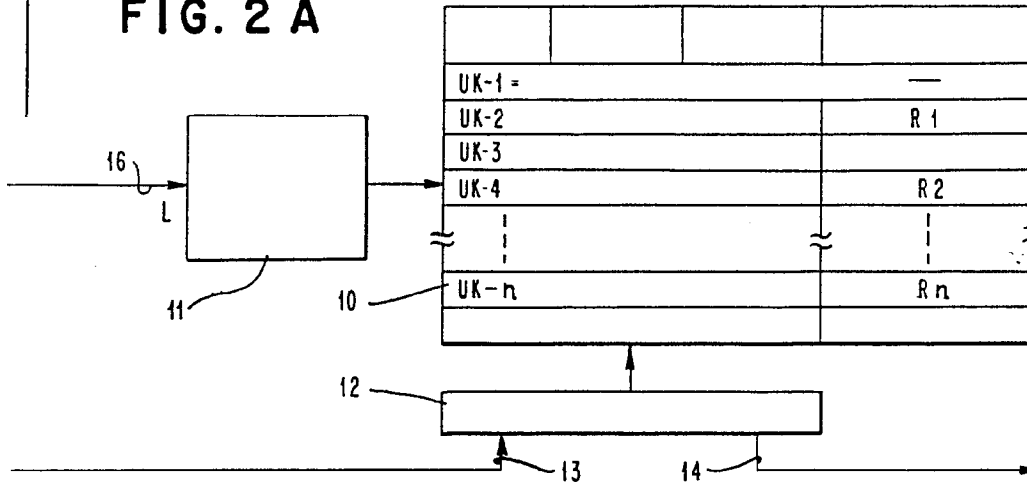
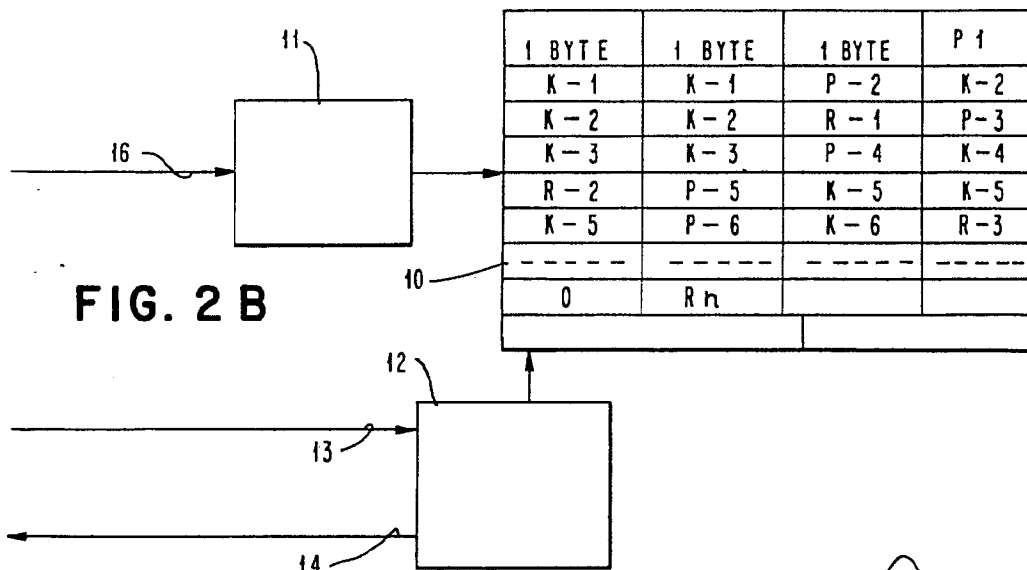


FIG. 2 B



380987



FIG. 3

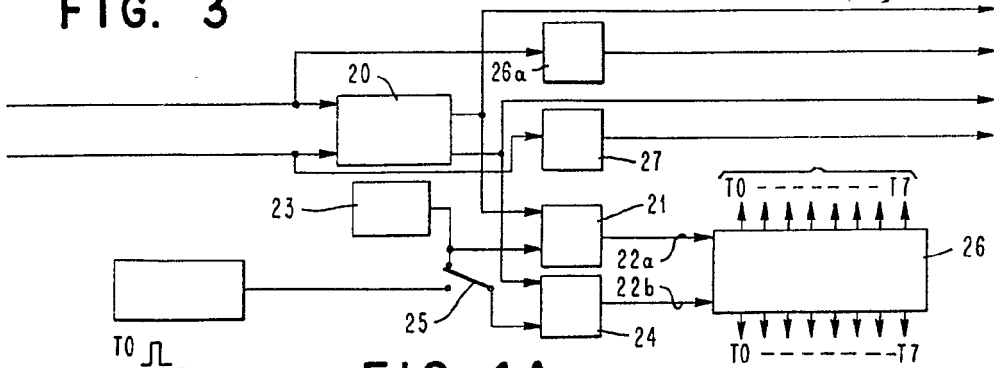


FIG. 4A

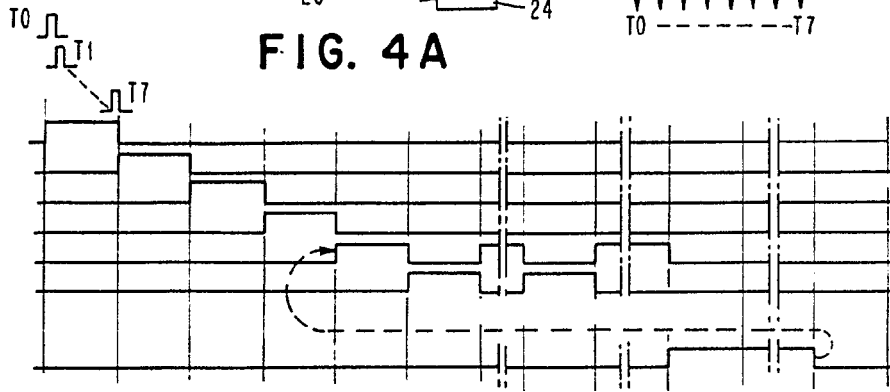


FIG. 4B

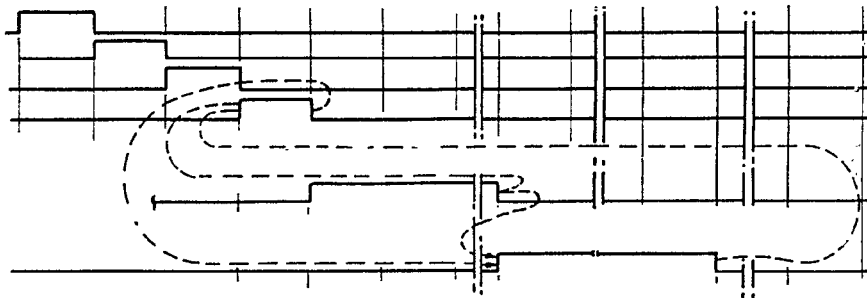


FIG. 5A

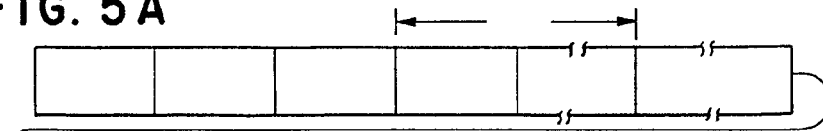
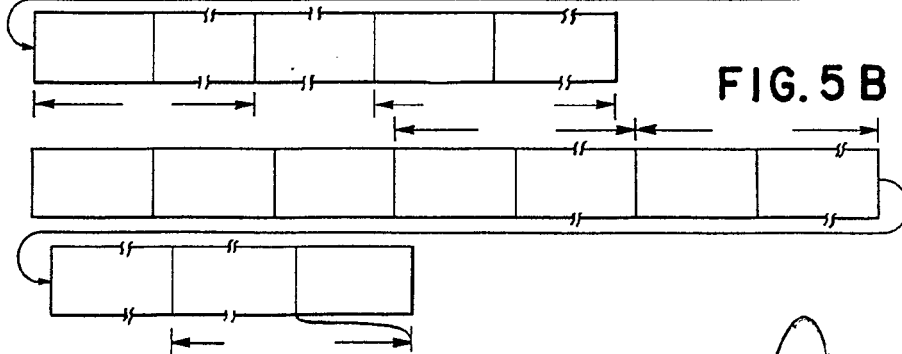


FIG. 5B



Alberro de *[Signature]*  
Por Poder.

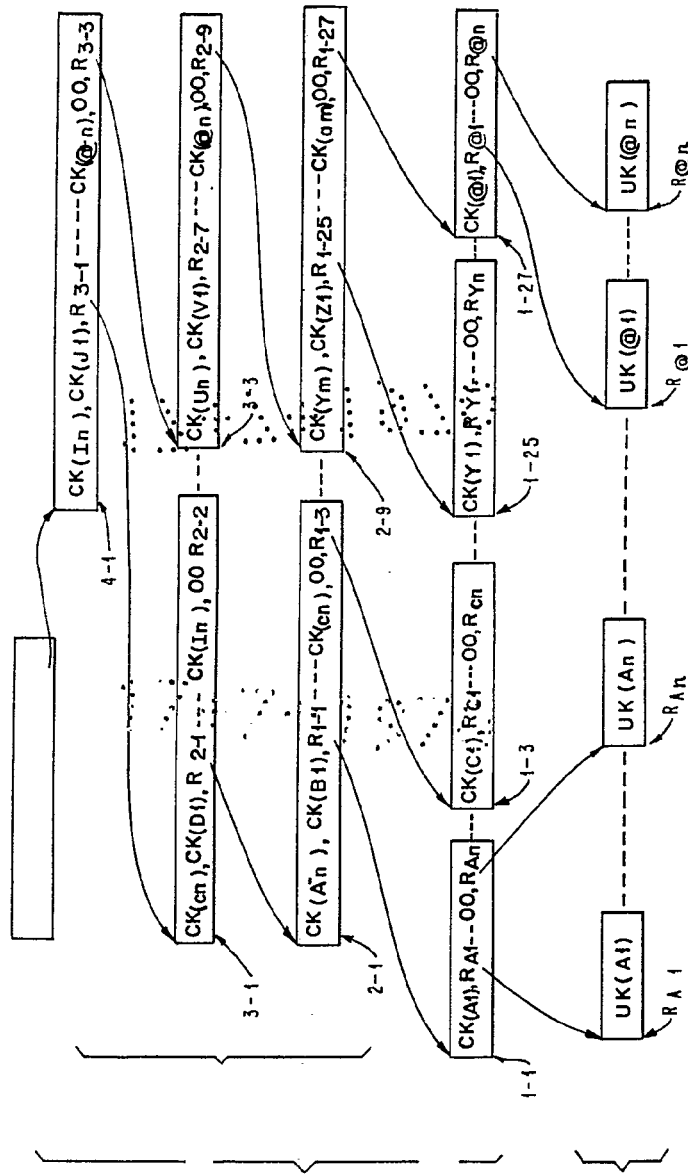


FIG. 6

Alberto *Barbieri*  
 Per Padre.

300007

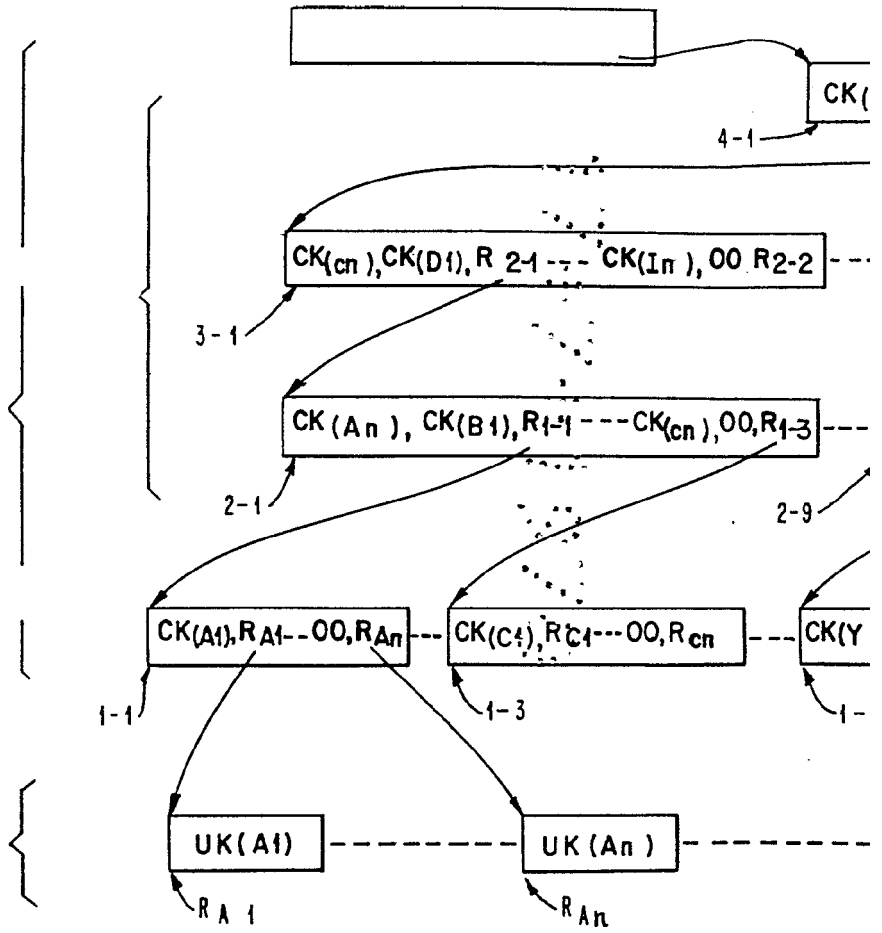


FIG. 6

380987

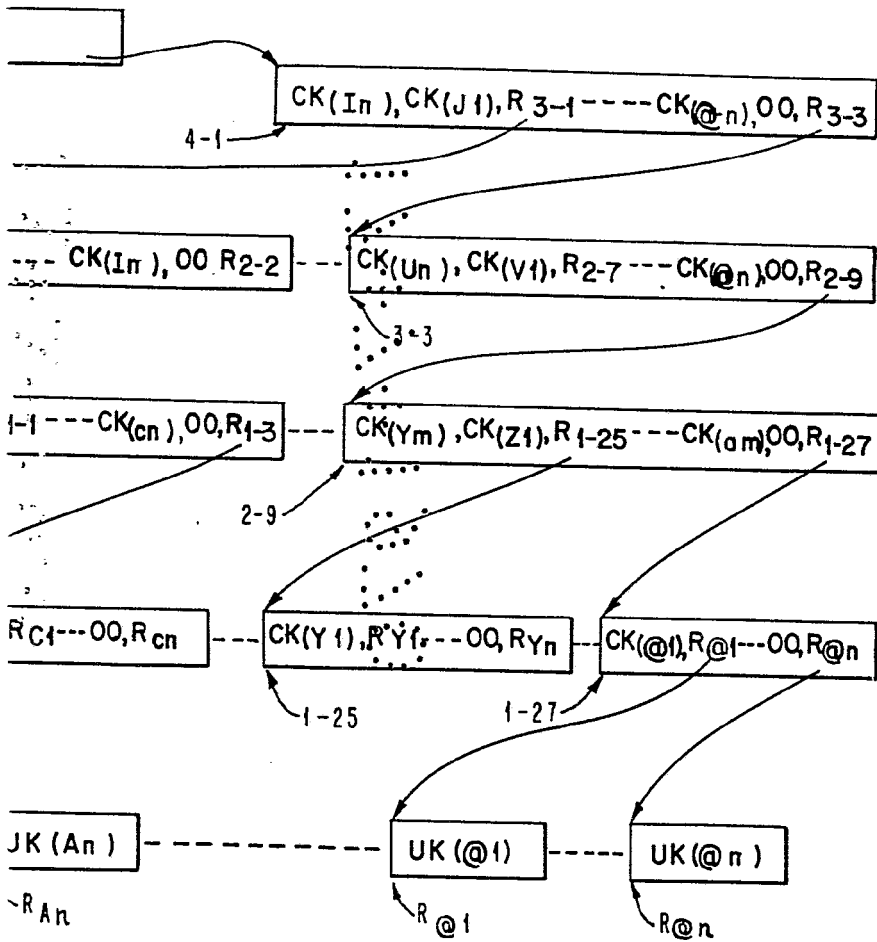


FIG. 6

Alberto  
Par Poder.

380987



FIG. 7A

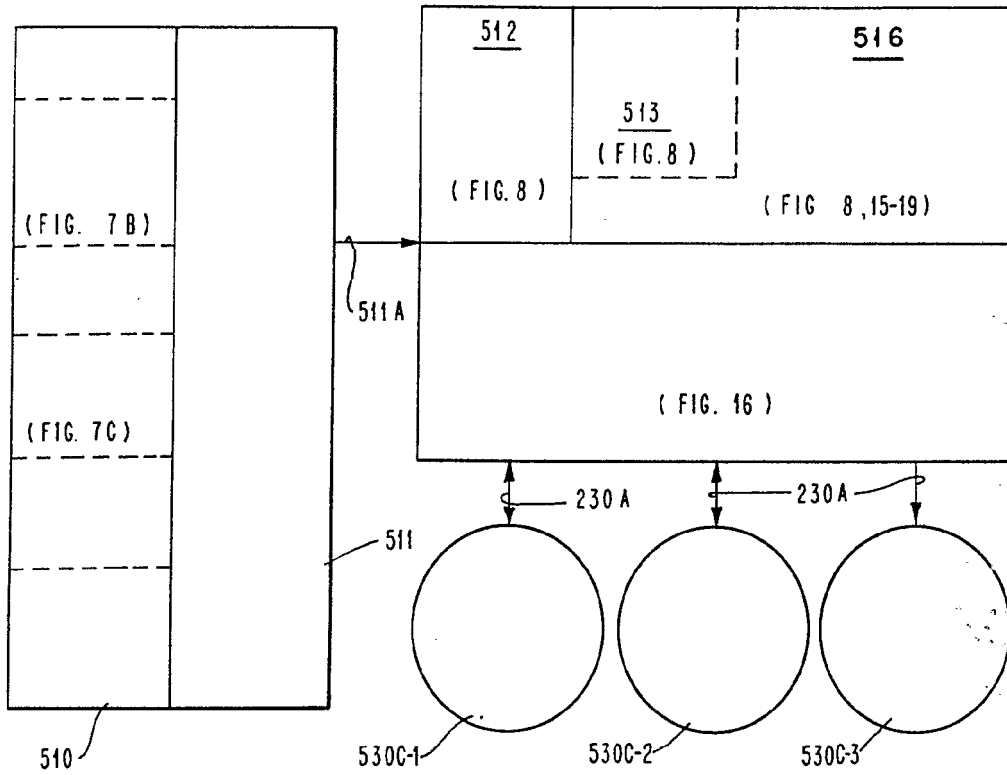


FIG. 7B

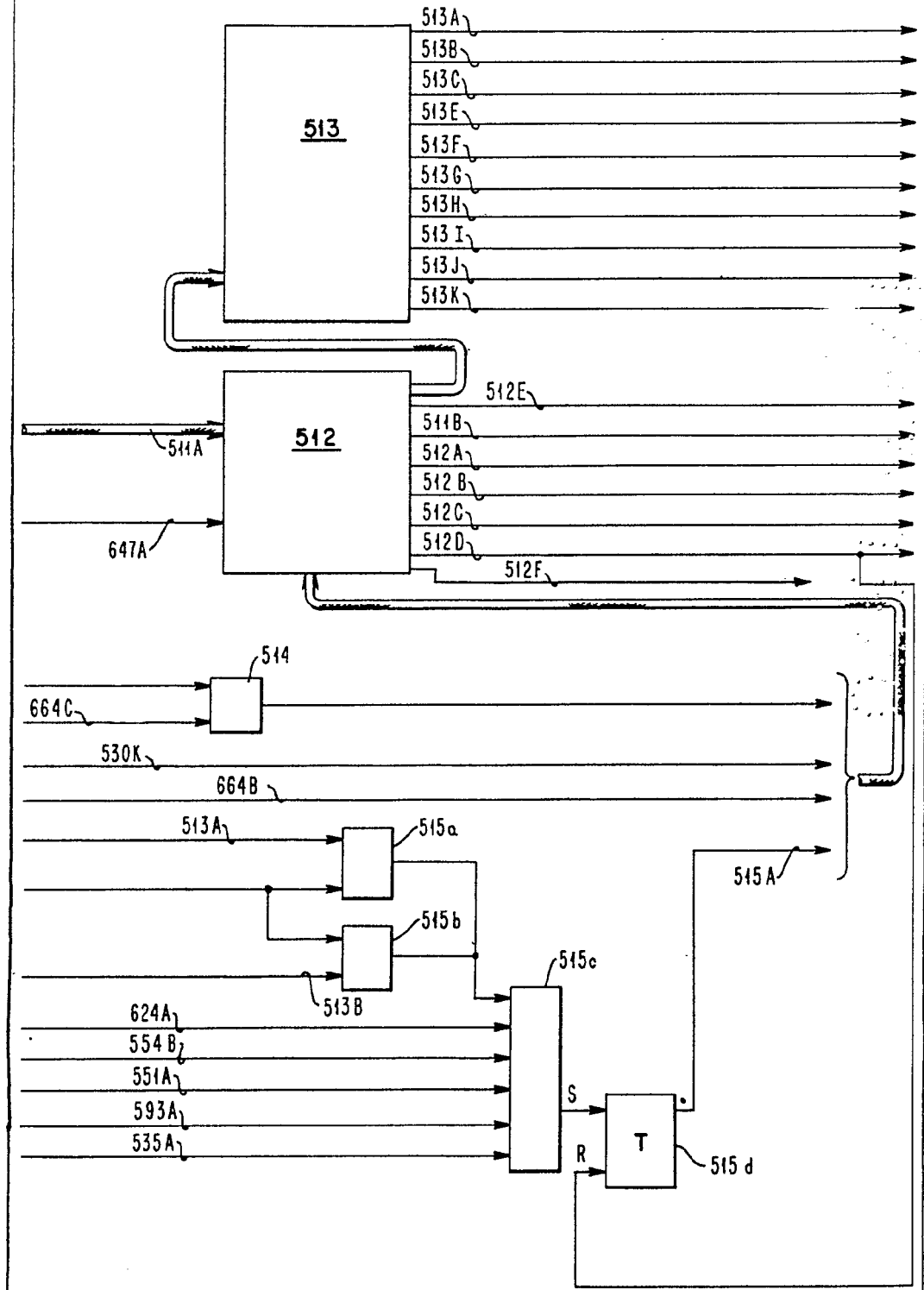
<u>L1</u>	<u>L2</u>		<u>LK</u>

FIG. 7C

<u>L1</u>	<u>L2</u>	<u>L3</u>	<u>LK</u>
R1-1	R2-1	R3-1	RK-1
⋮	⋮	⋮	⋮
R1-h	R2-h	R3-h	RK-h

FIG. 8

380987

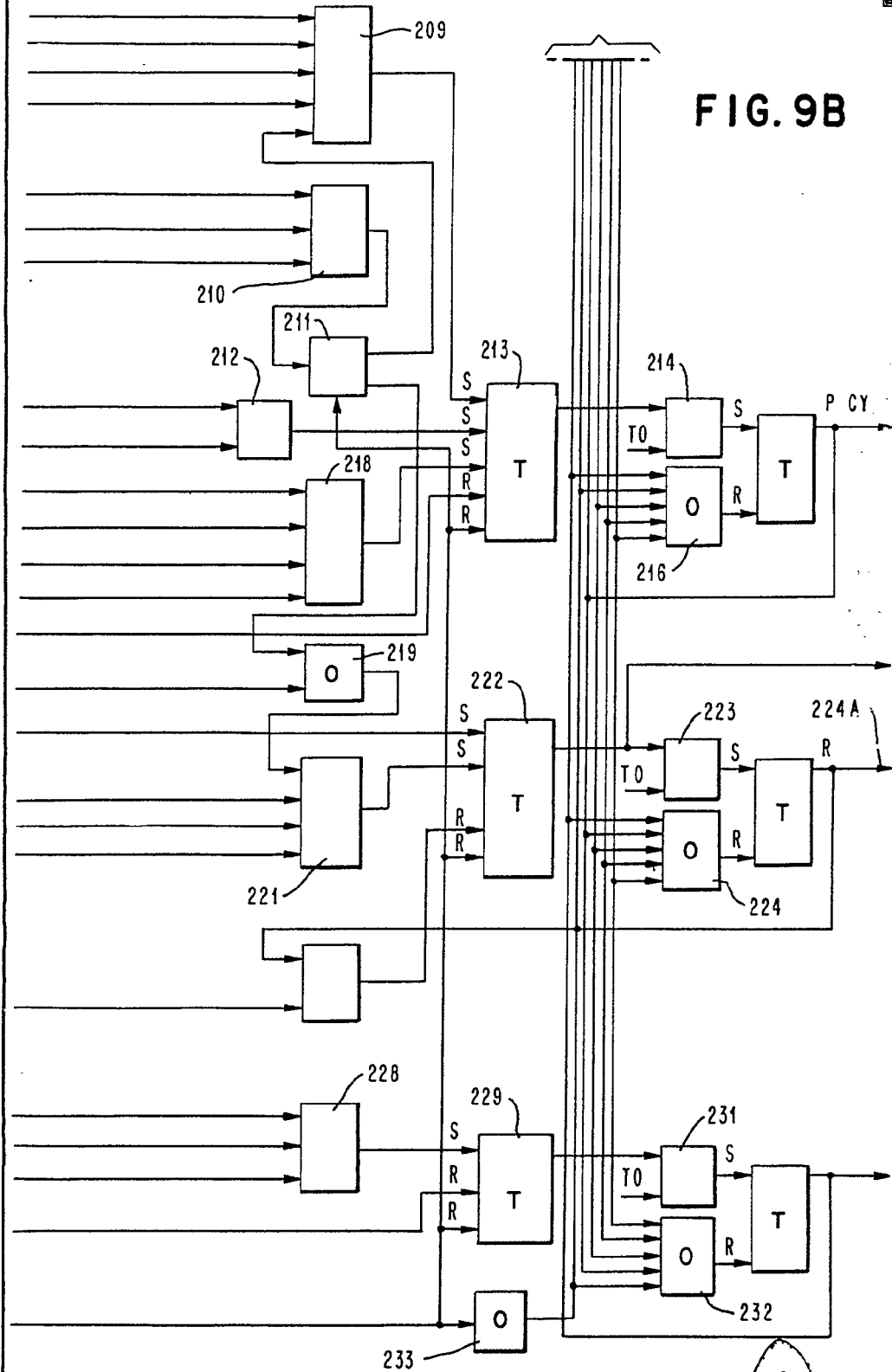




380987



FIG. 9B





RA 151

380087

17 Oct

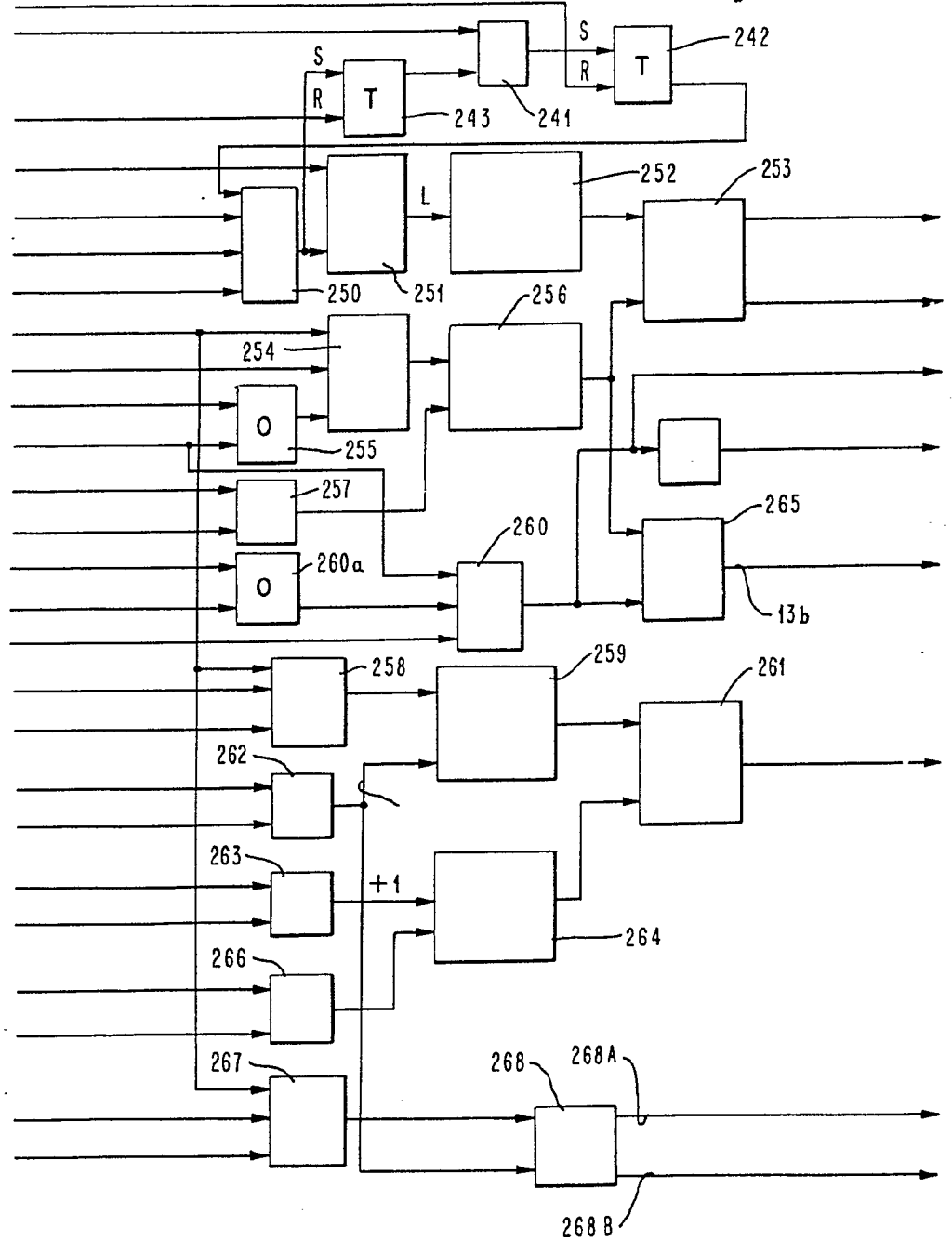


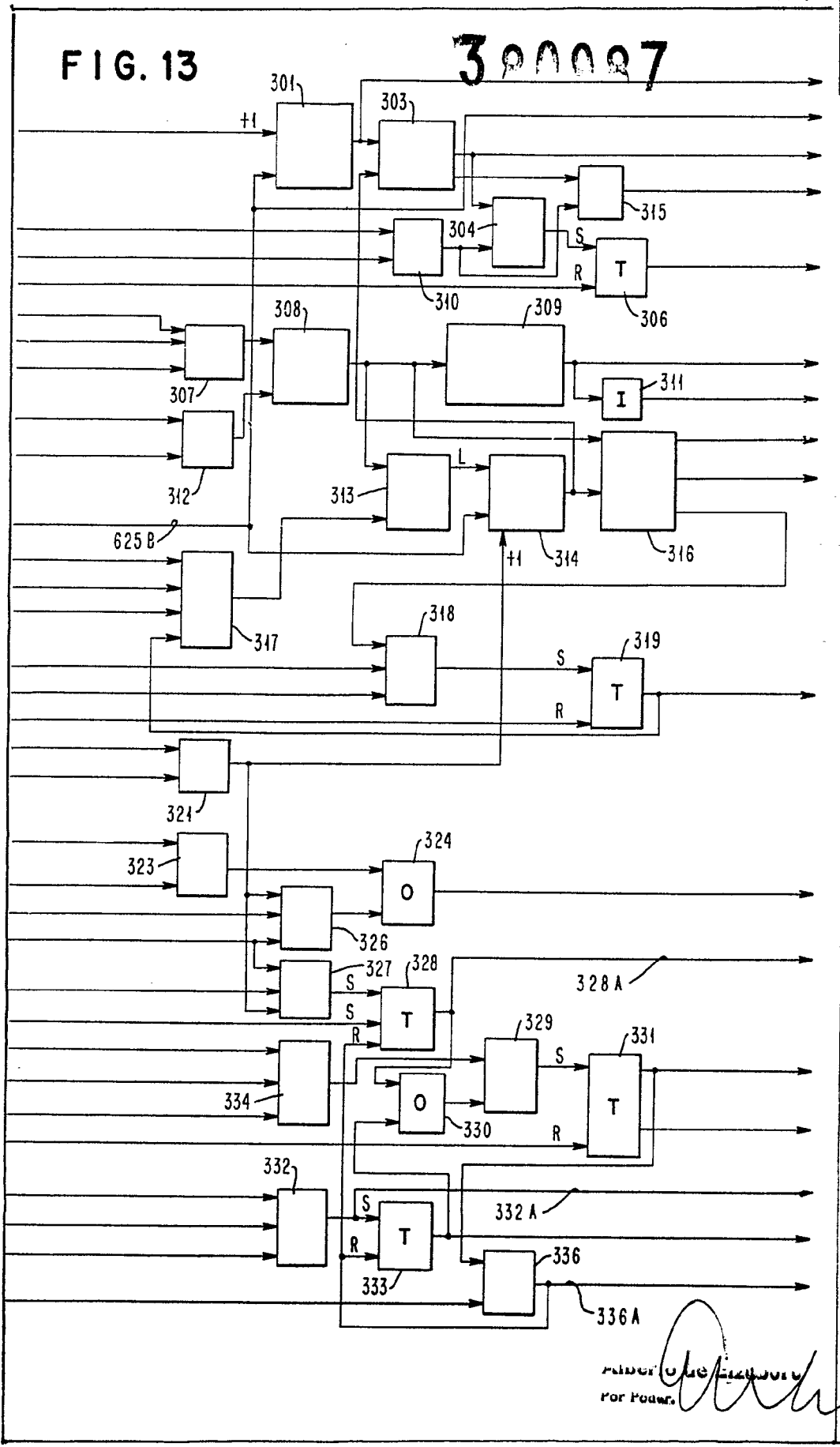
FIG. 12

Alfonso de Rivas  
Por Poderes



FIG. 13

300027

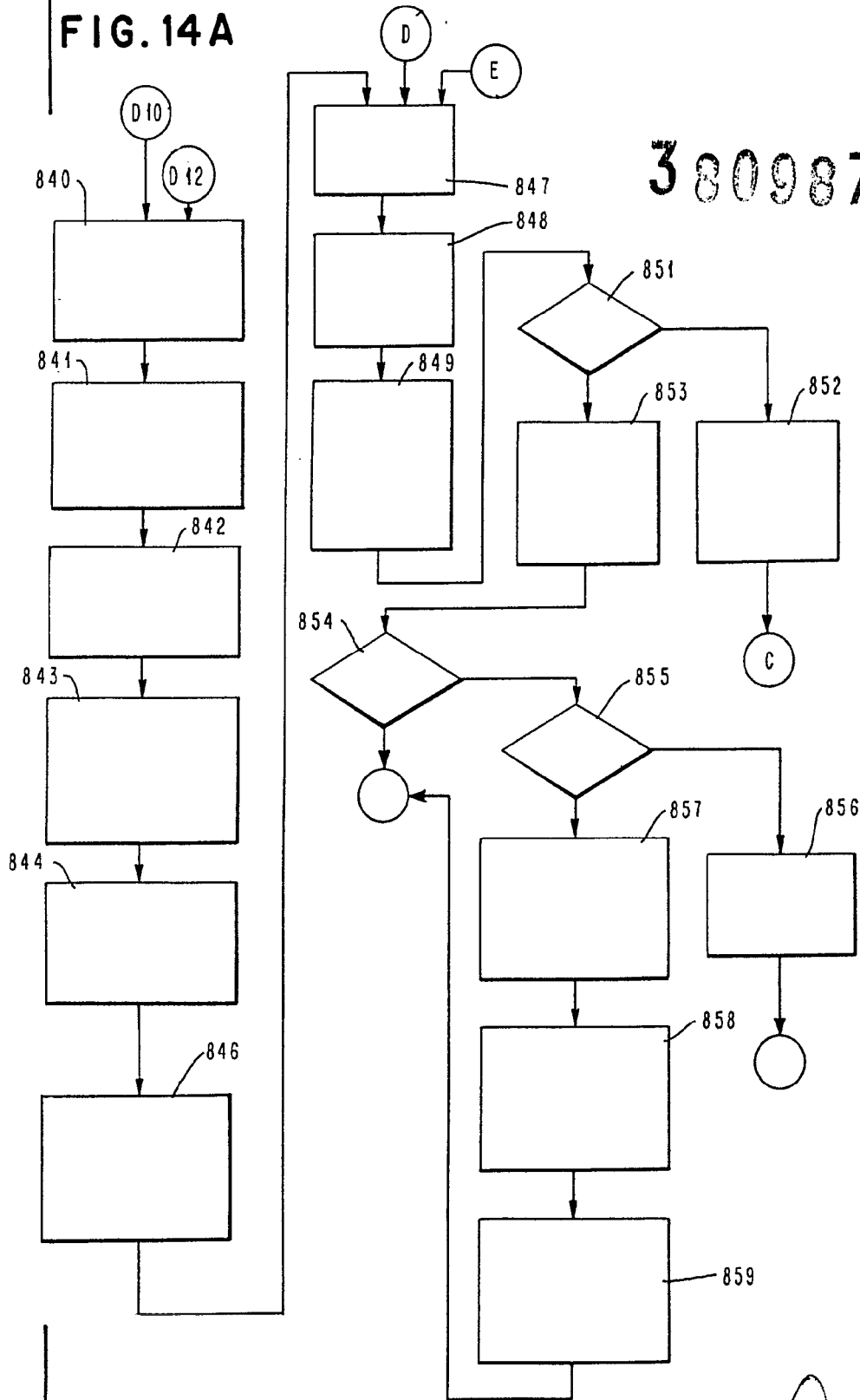


IBM CORPORATION  
FOR POWER



FIG. 14A

380987



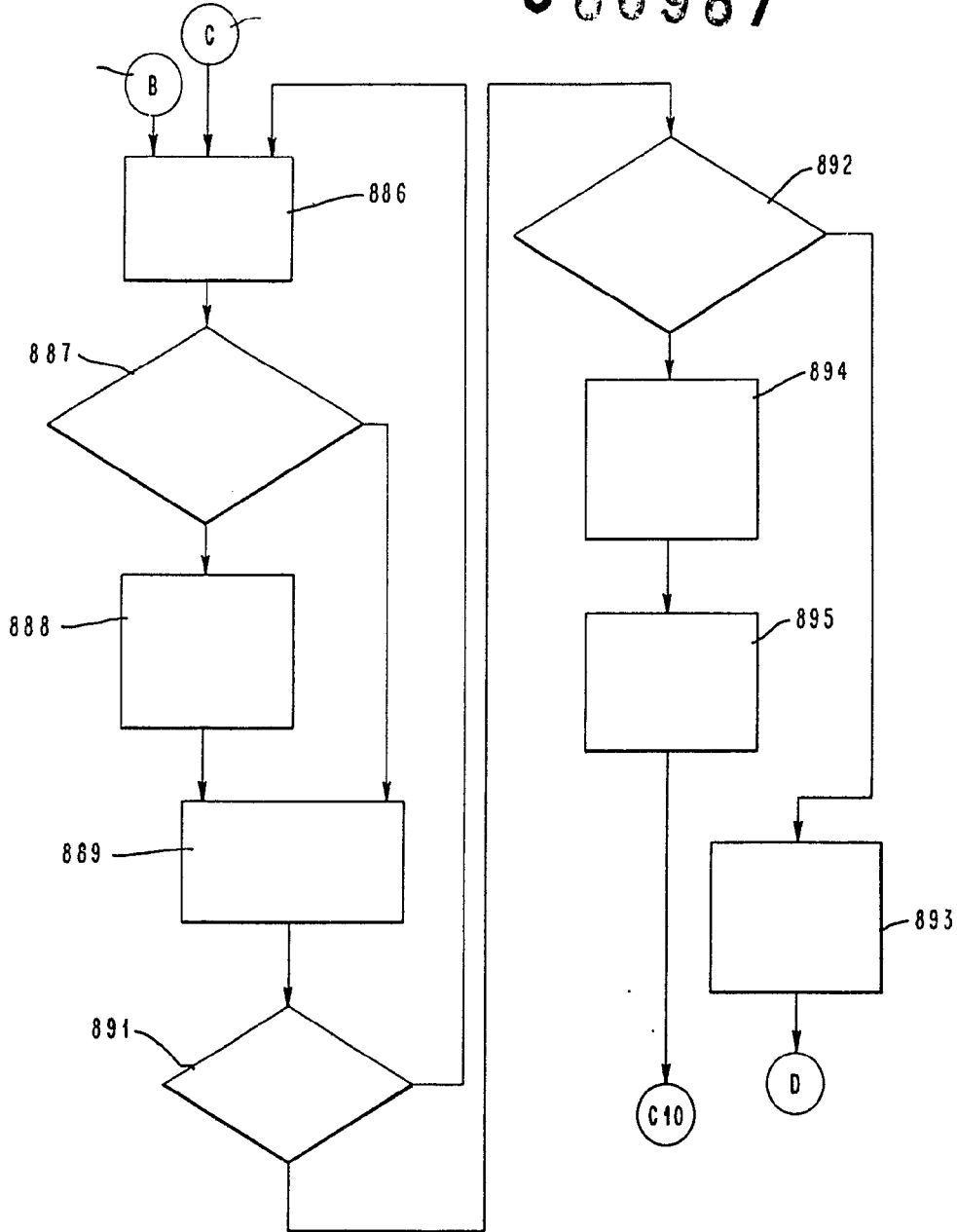
For Podar *[Signature]*





FIG. 14C

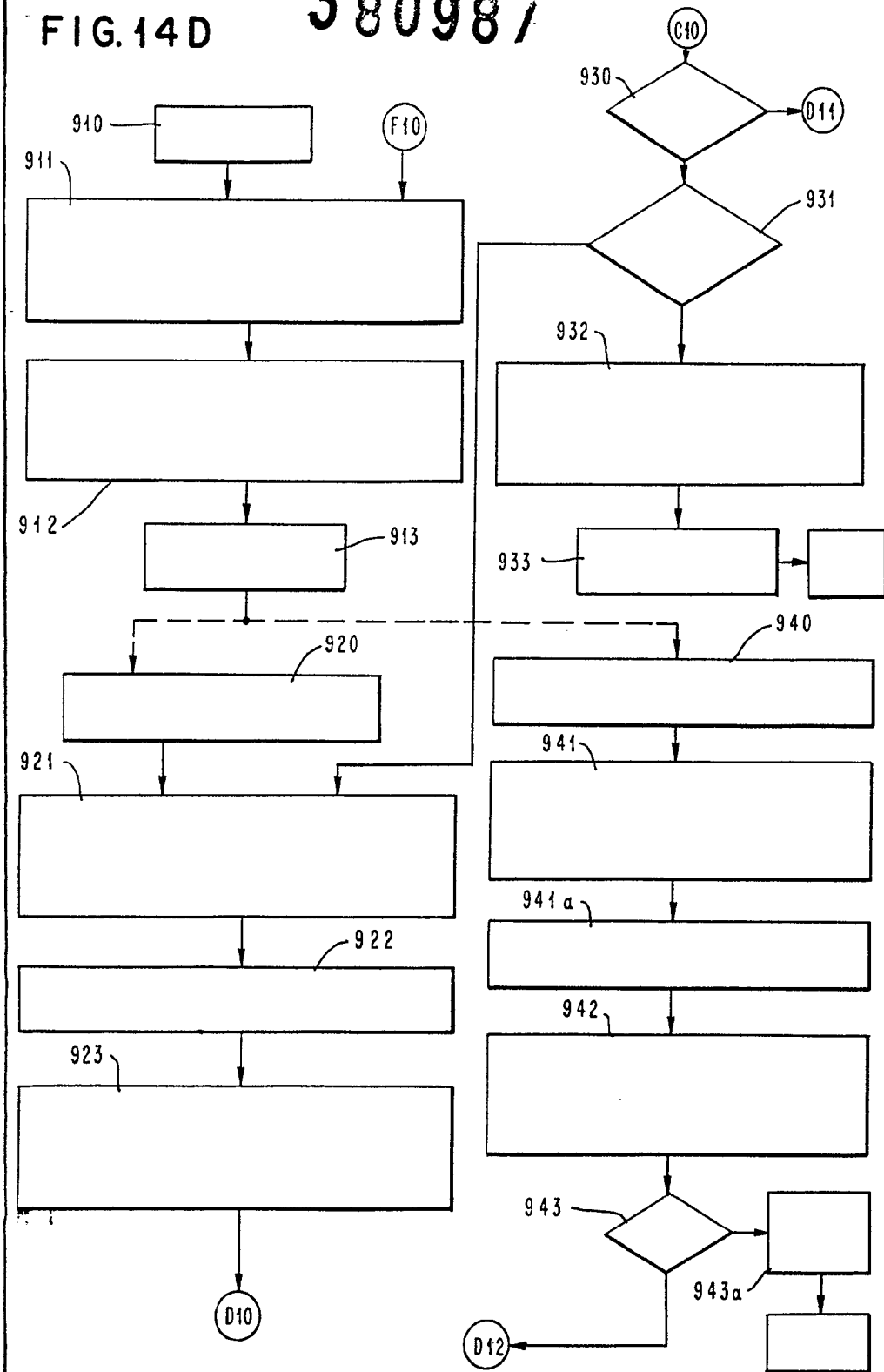
380987



Alberto de S. *[Signature]*  
For Podor.

FIG. 14D

380987



*[Handwritten Signature]*  
For Podes.



FIG. 14 E

380987

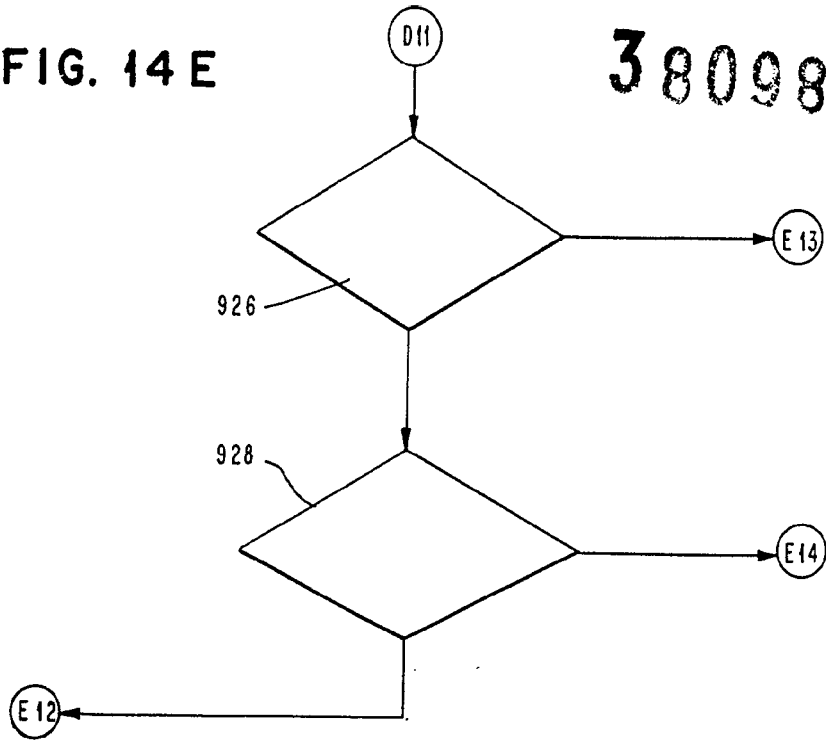
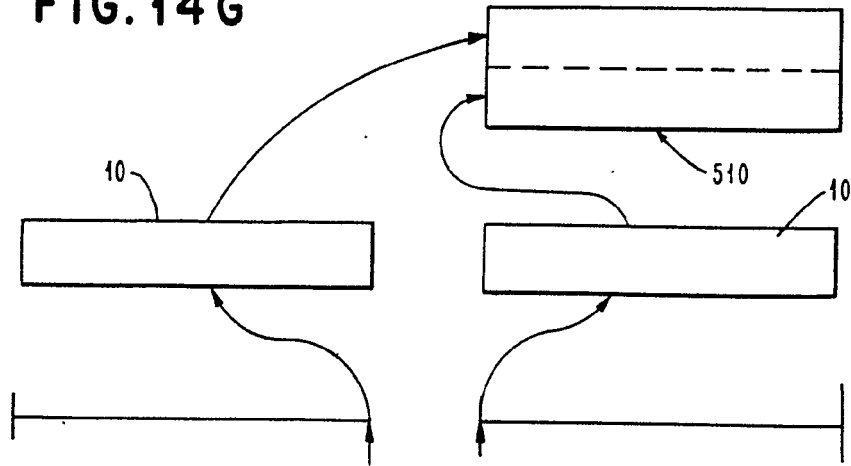


FIG. 14 G

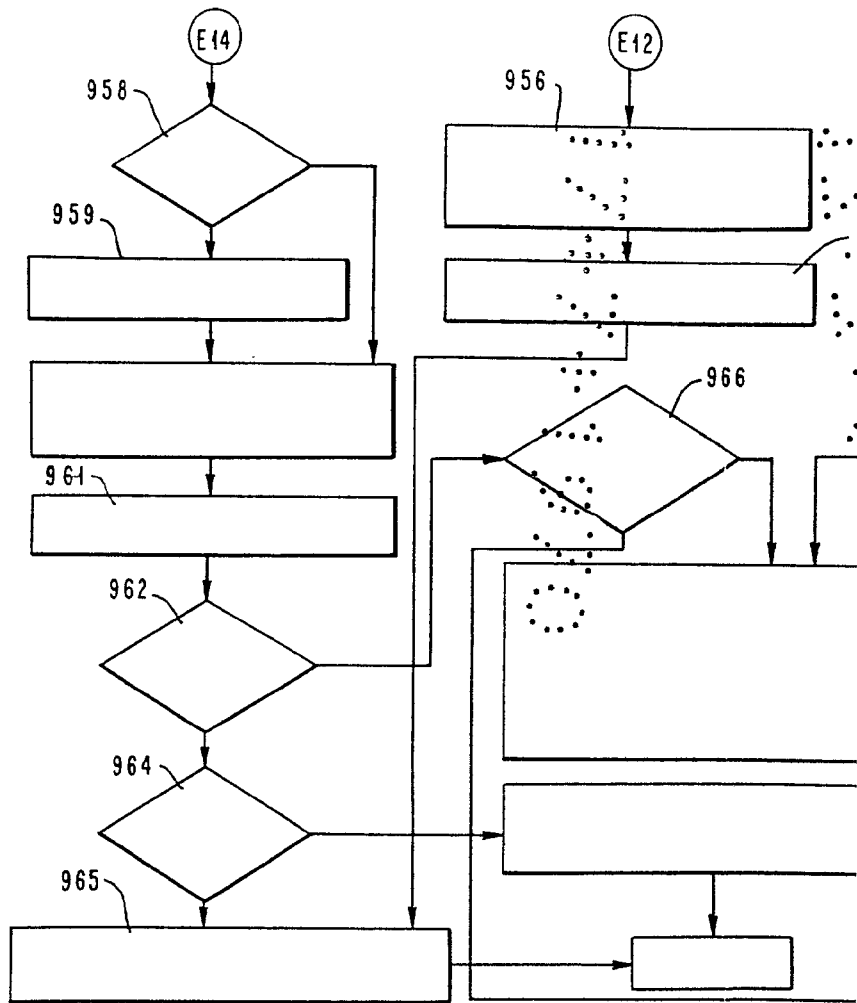


Alberto de Mazonu  
for Podar



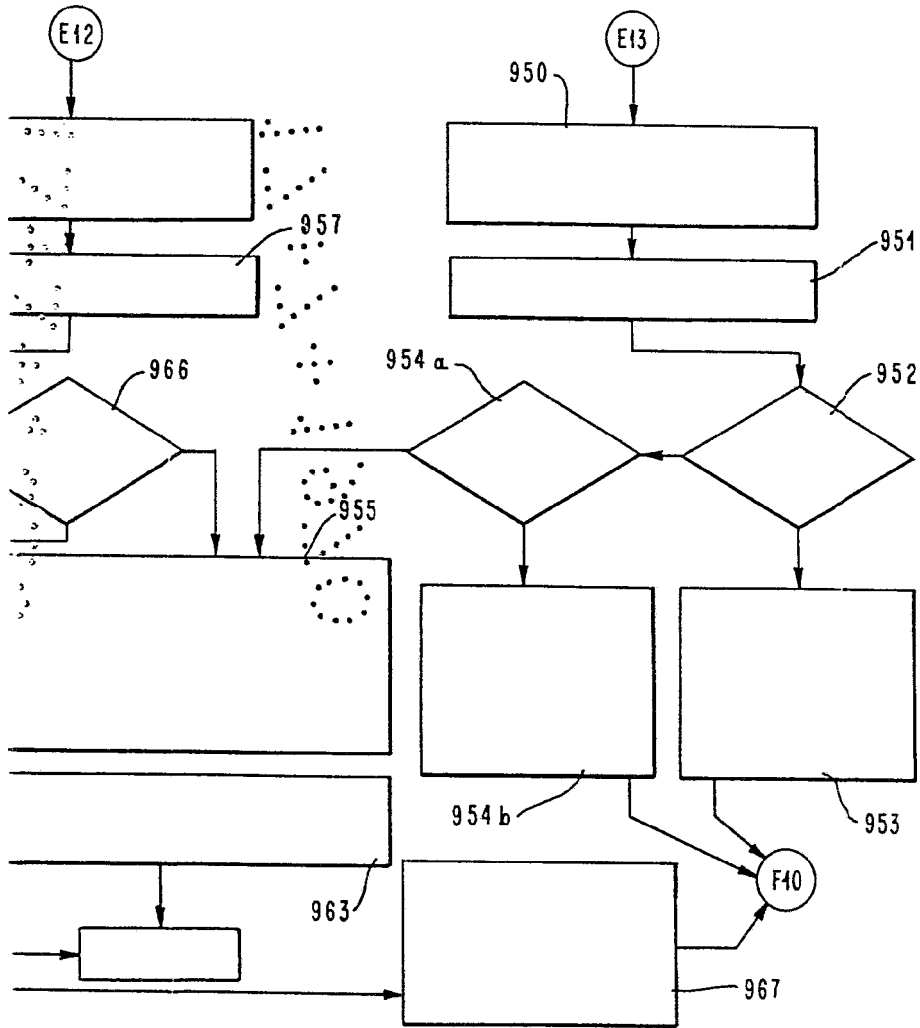
380987

FIG. 14F





380987



Alberto de Elizalde  
Por Poder

380987

380987

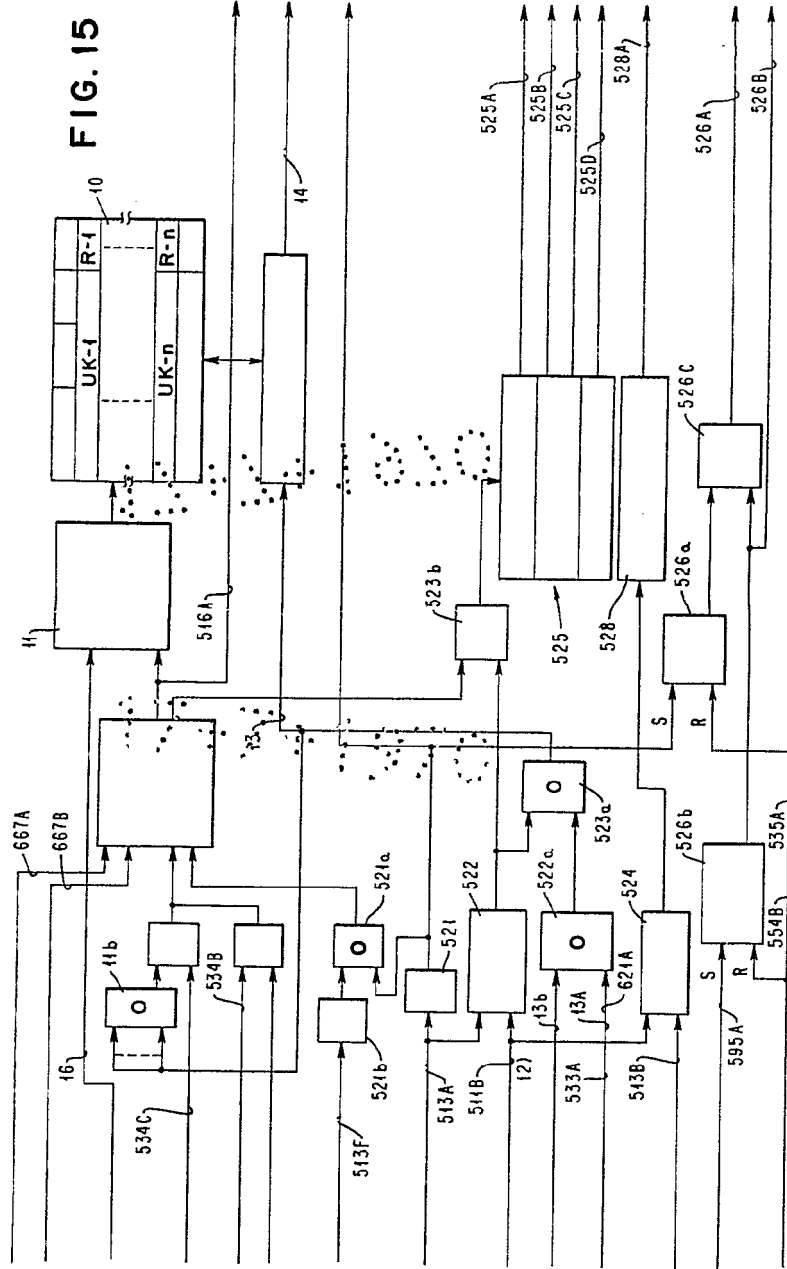
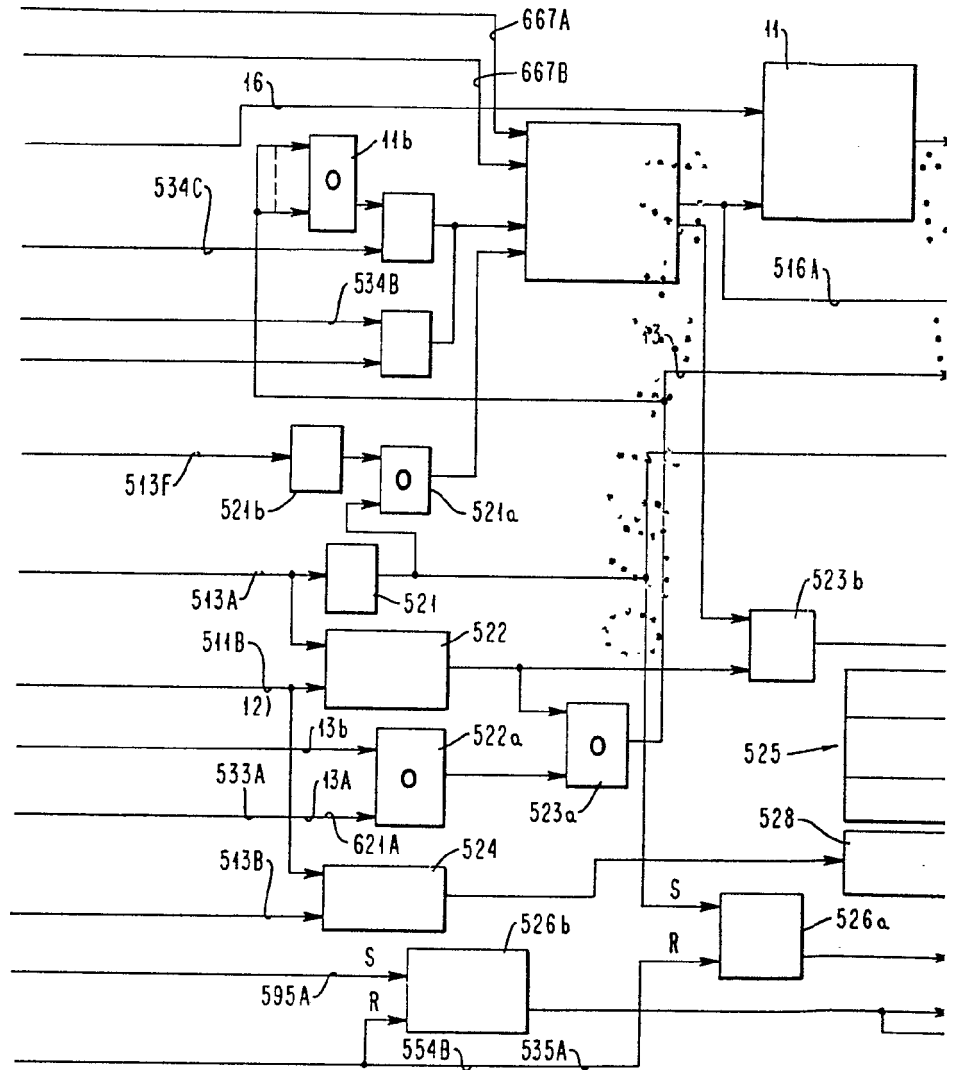


FIG. 15

Alberto de  
Per Federa

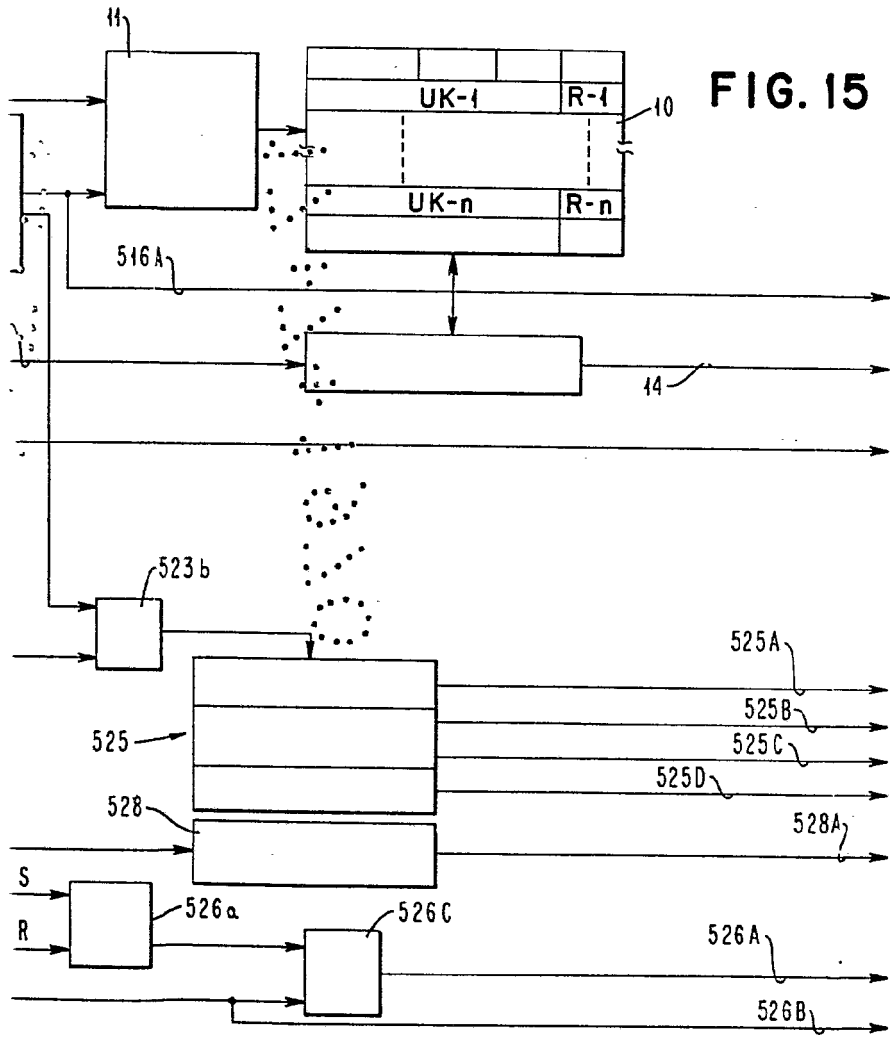
380987





3 00987

FIG. 15

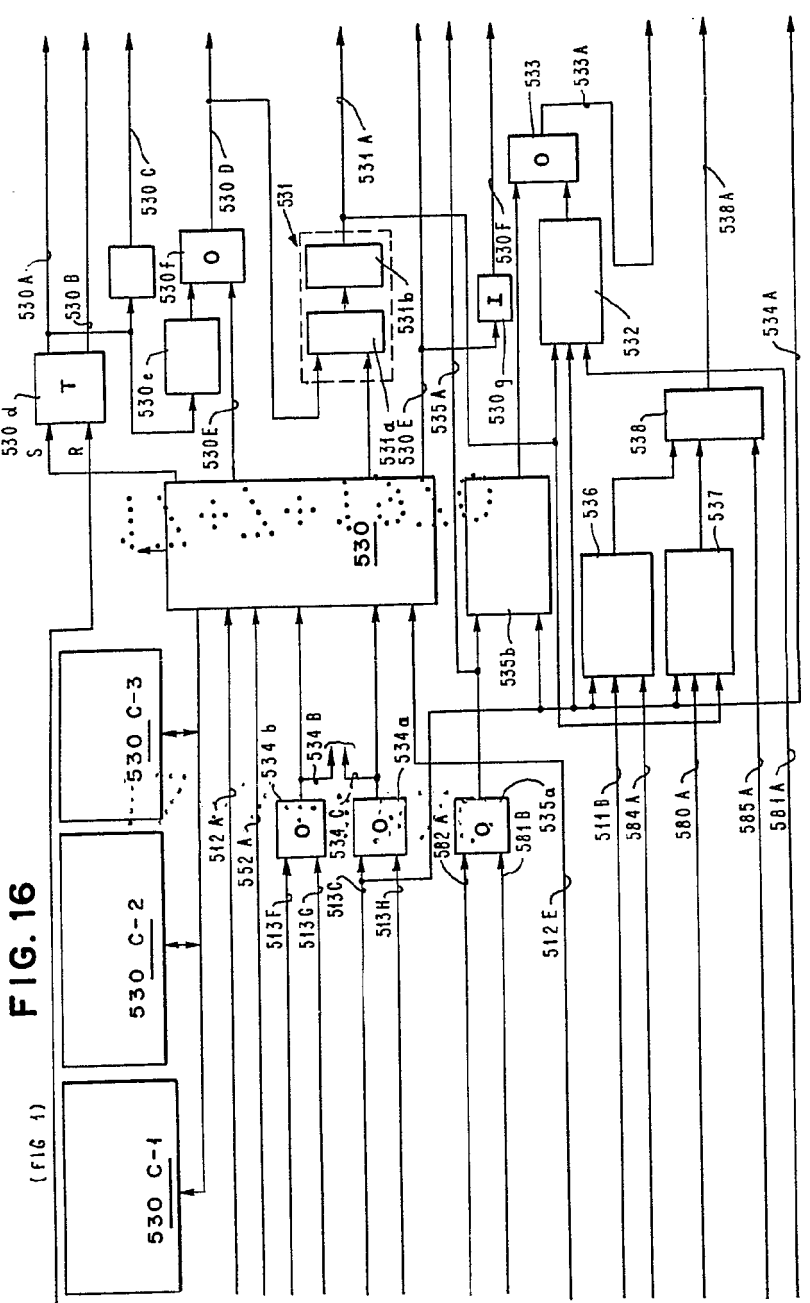


Alberto de Cizouru  
Per Podar



380987

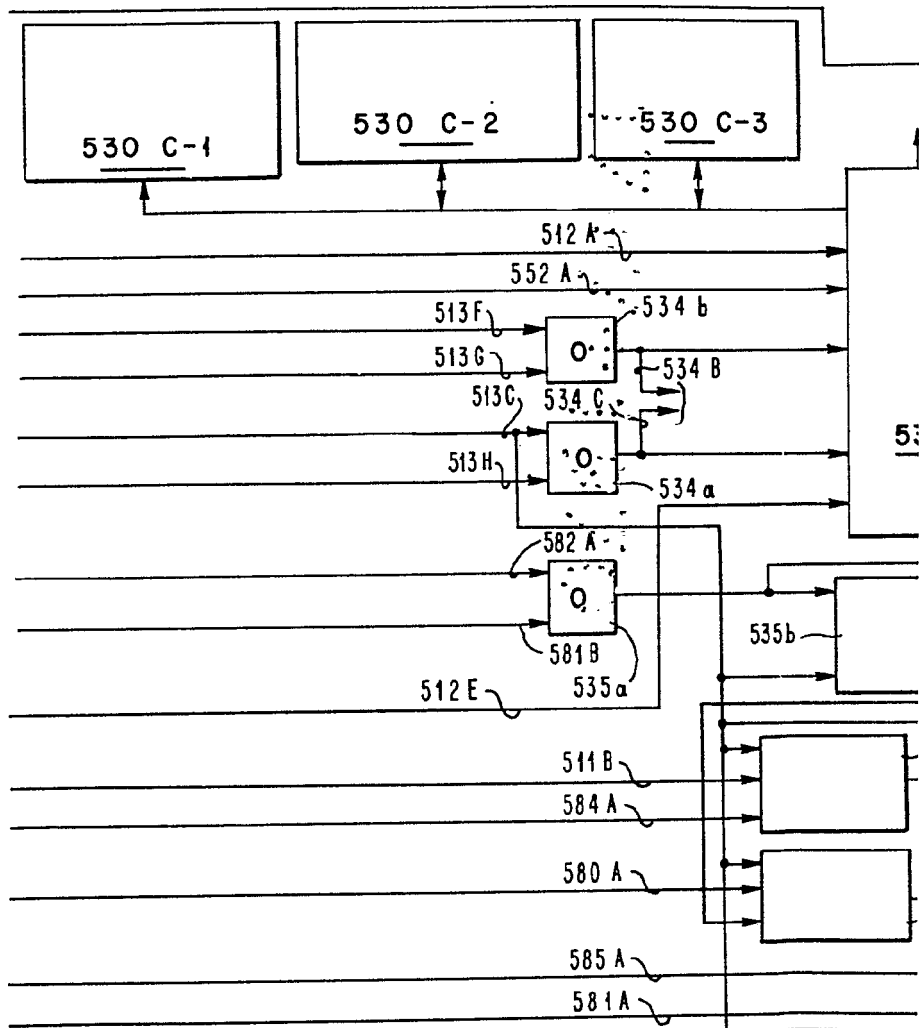
380987



Albert E. Podak  
For Podak

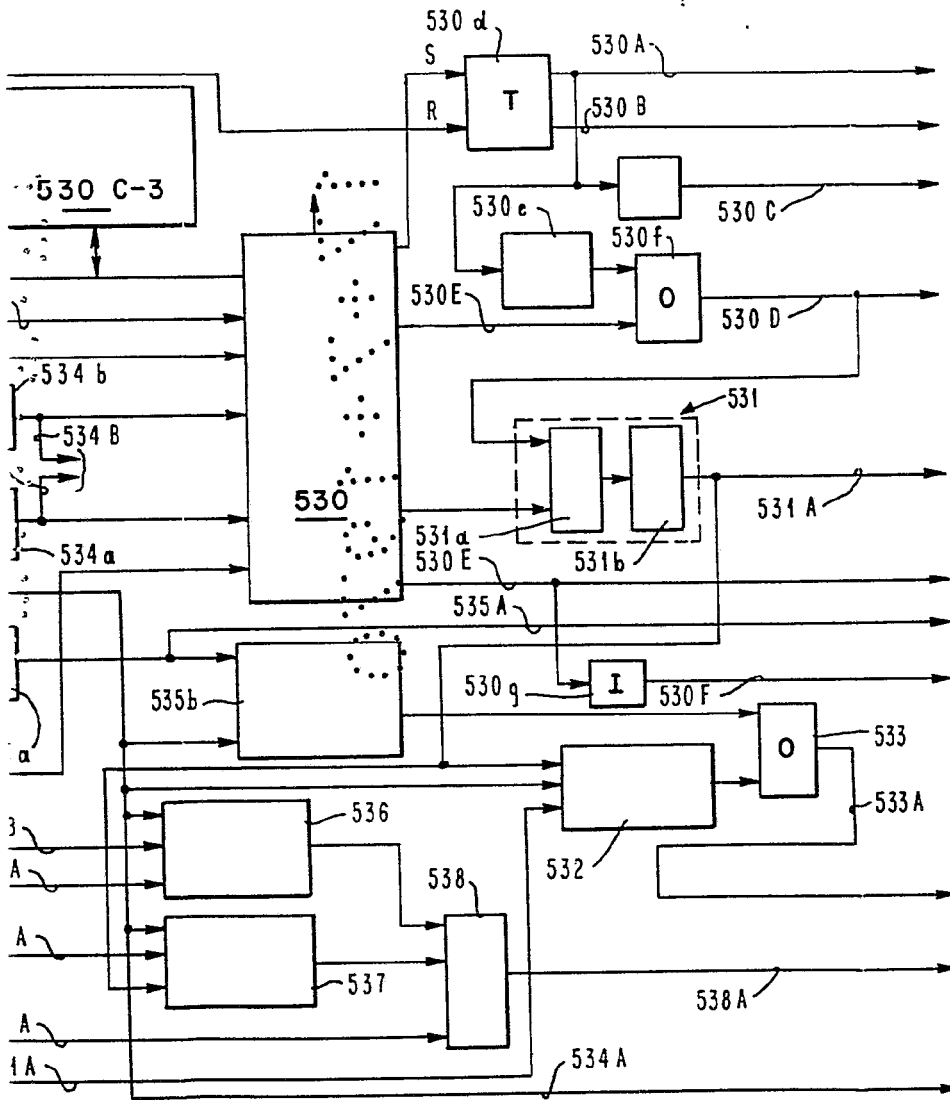
887

(FIG. 4) FIG. 16



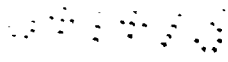


380987



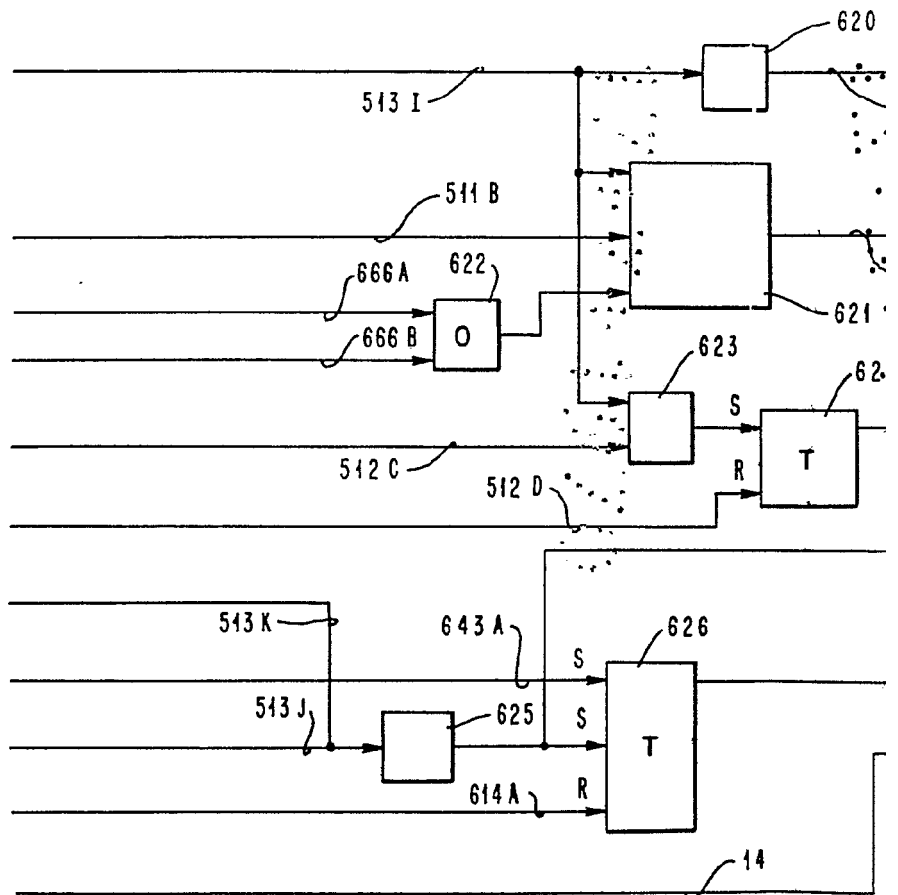
Alberto de M. ...  
Por Poder...





3 8 2 3 3 7

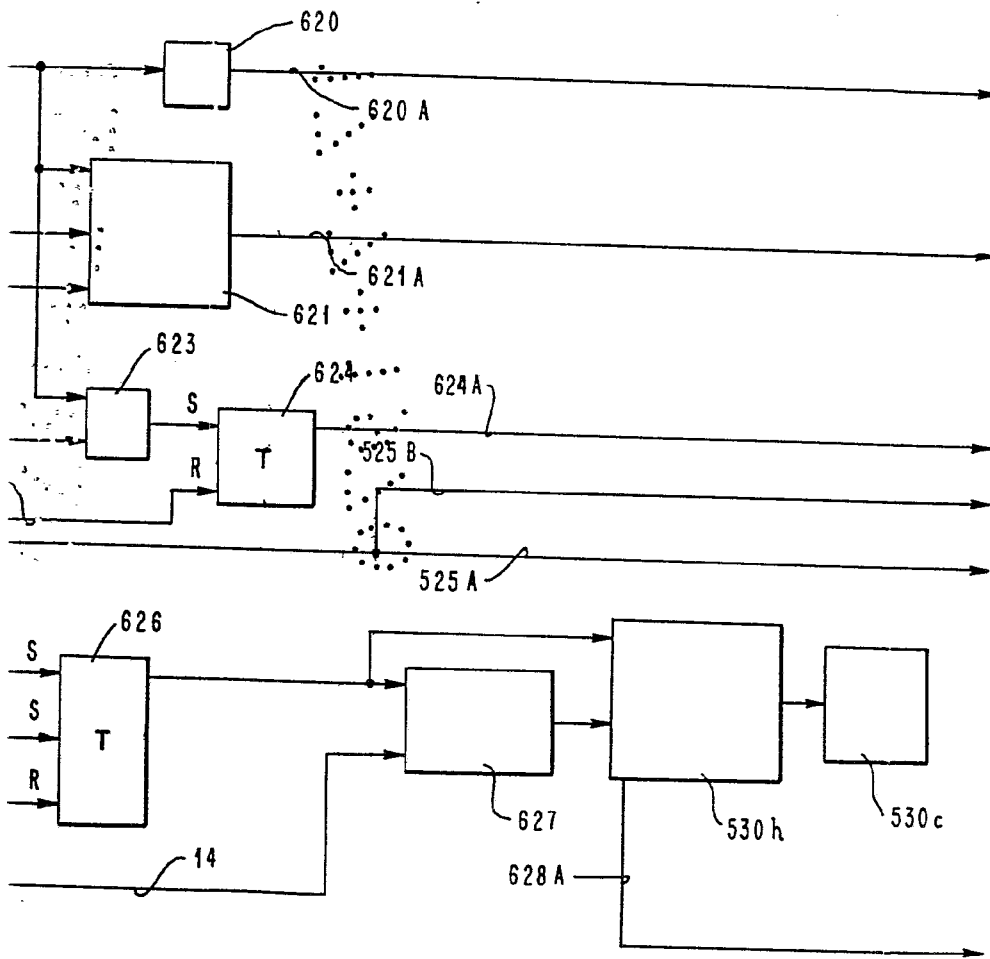
FIG. 17





380987

17



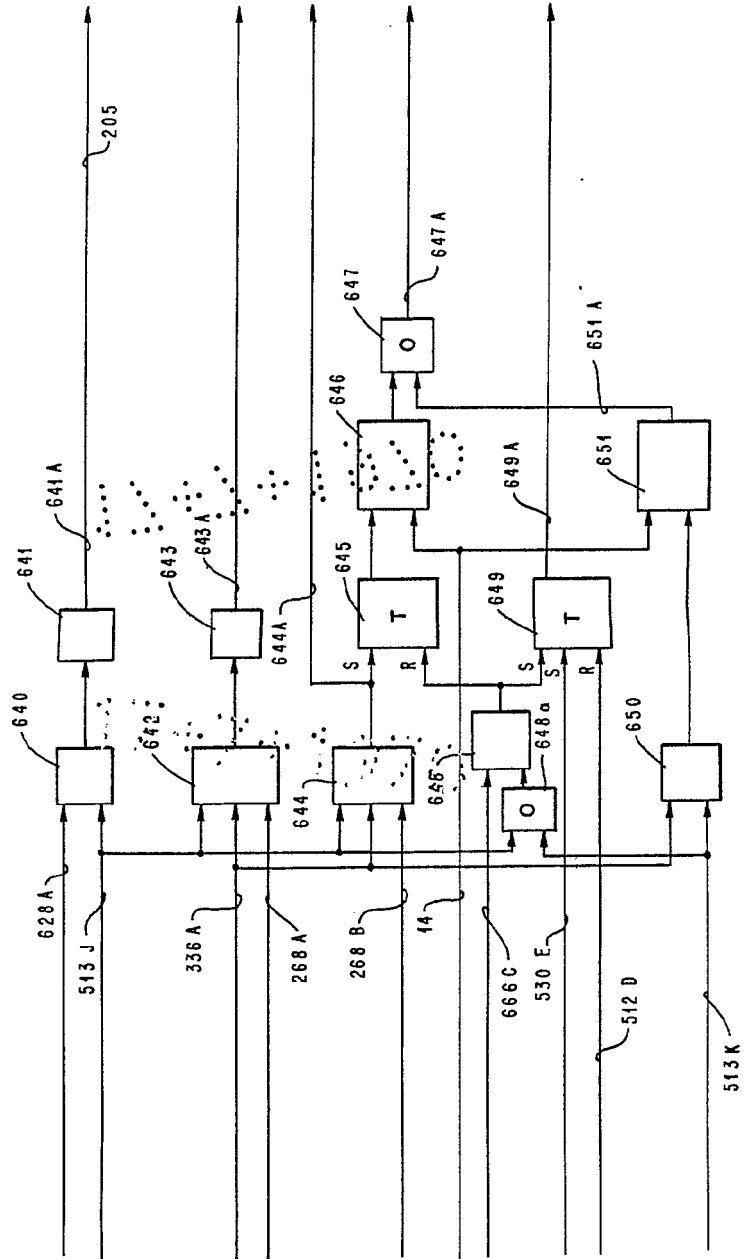
ALBERT G. ...  
FOR PATER.

INTERNET



380987

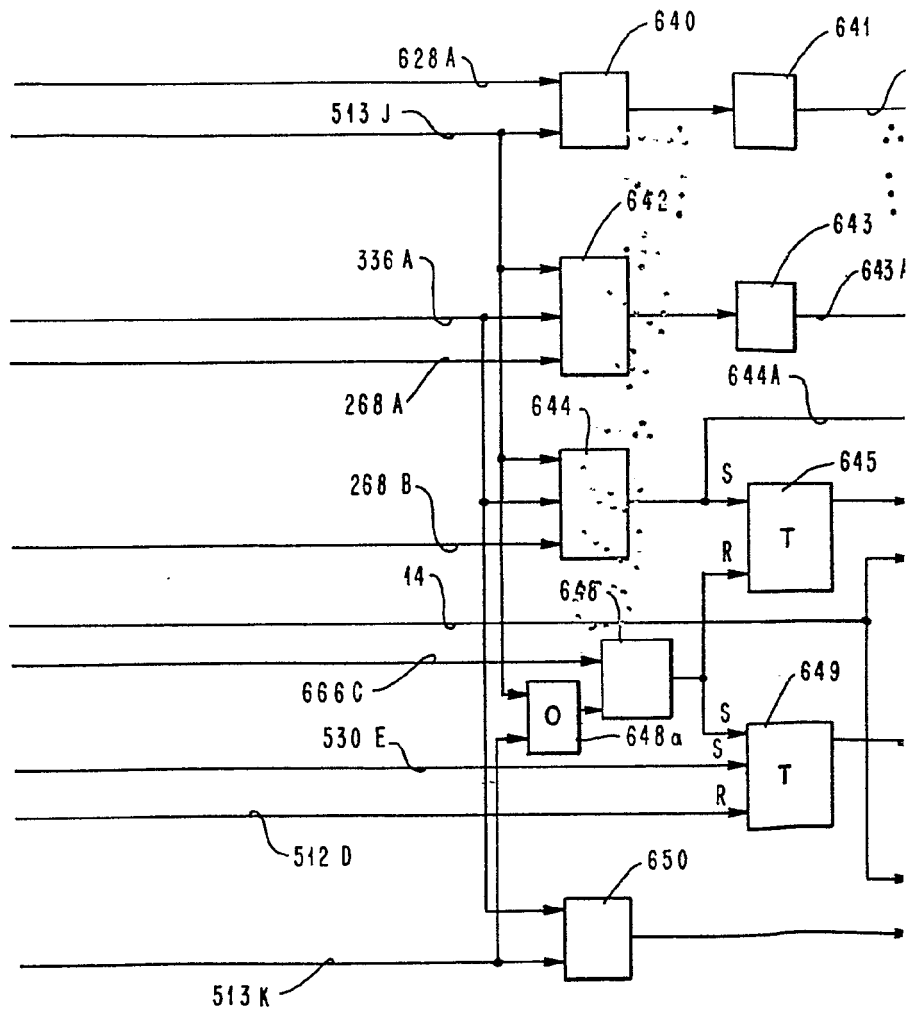
FIG. 18



*Approved*  
For Filing.

1987

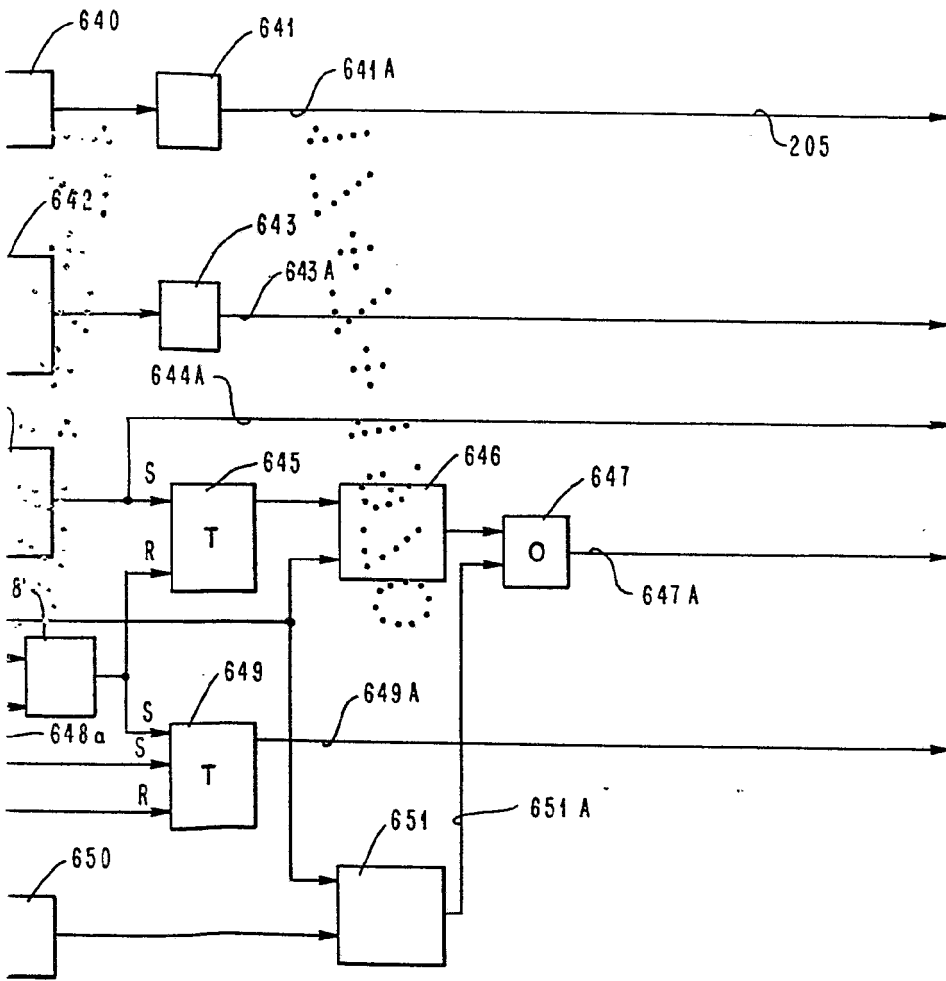
FIG. 18



SECRET



380987



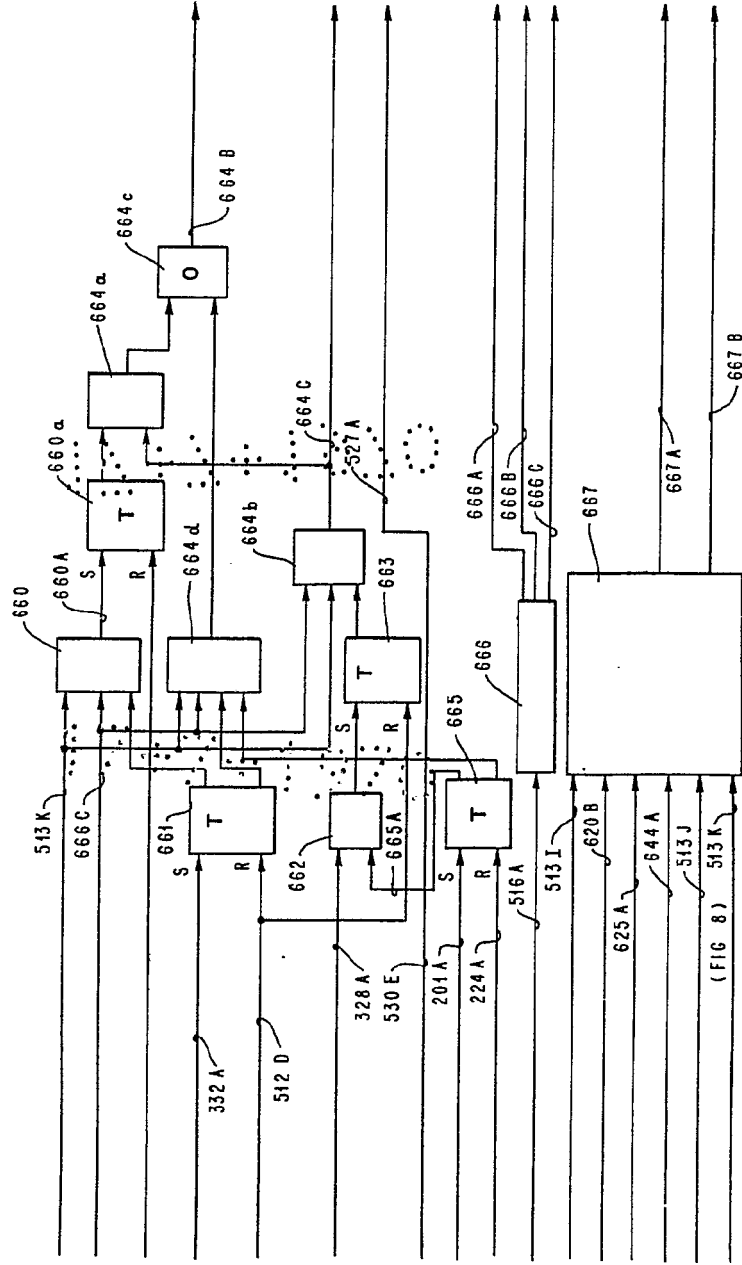
Approved for Release  
For Foden.



380987

380987

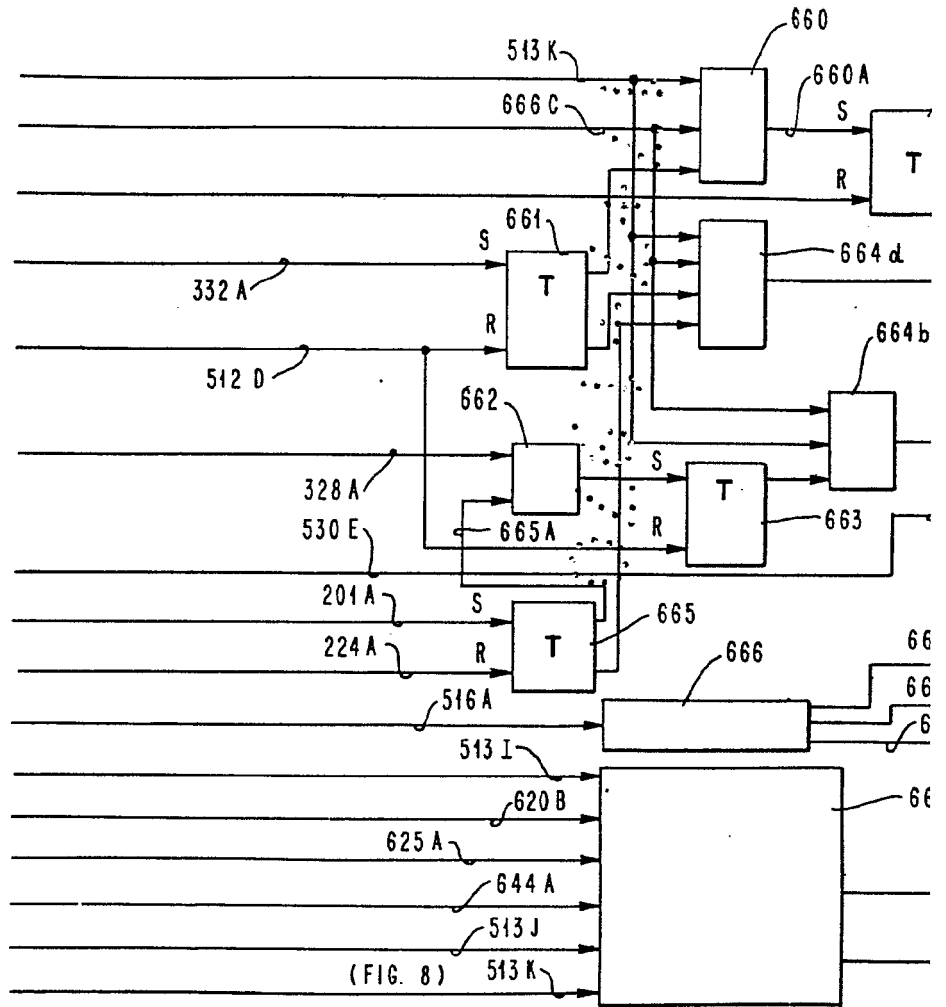
FIG. 19



*Handwritten signature or initials.*

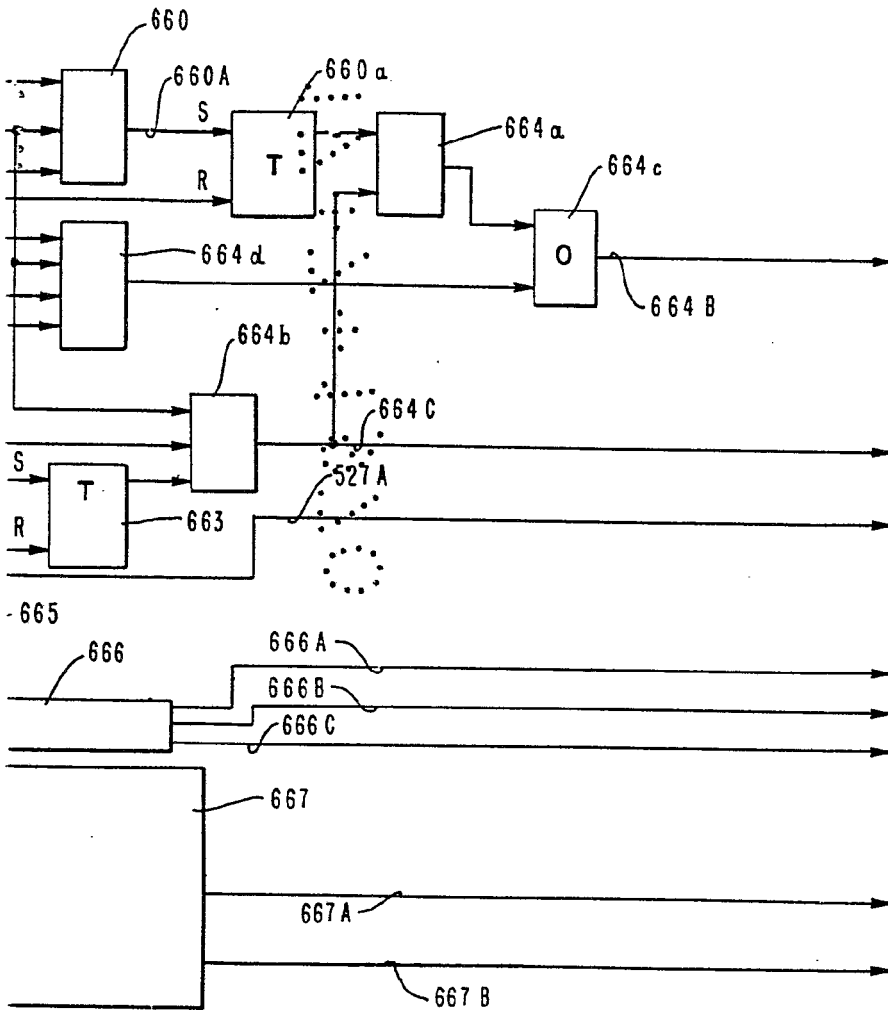
380987

FIG. 19





300987



Approved: \_\_\_\_\_  
Per: \_\_\_\_\_

*[Handwritten signature]*