

335302

IBM Docket 10862



MEMORIA DESCRIPTIVA

que se presenta para unir a la solicitud  
de

PATENTE DE INVENCIÓN

formulada el 5 de Enero de 1.967, con el nº 335.302

en

E S P A Ñ A

por VEINTE años

a nombre de INTERNATIONAL BUSINESS MACHINES CORPORATION,  
entidad norteamericana, establecida en Armonk, N.Y., Es-  
tados Unidos de América, por:

"UN DISPOSITIVO DE PROTECCION DE UNA MEMORIA"

---

La presente invención se refiere a sistemas de protec-  
ción de memorias, y más especialmente a un sistema para con-  
trolar de modo flexible las interacciones entre diversos  
programas que residen en la memoria de un sistema calcula-  
dor aritmético o numérico (por dígitos).

5

La memoria de una máquina calculadora numérica con-  
tiene en general instrucciones y datos, que se combinan  
y agrupan para formar una pluralidad de programas diferen-  
tes. Uno o más de estos programas pueden considerarse como  
aplicaciones o programas utilitarios que, puestos en mar-

10



cha, hacen que se resuelva un determinado problema, o que se efectúe alguna otra función utilitaria. Los demás programas son de control (esto es, de supervisión, vigilancia o monitores, etc.), que determinan la sucesión (secuencia) en que han de realizarse los programas de aplicación, la verificación o búsqueda de errores en estos programas, y ejecutan otras varias funciones de control y gobierno interior, esenciales para el funcionamiento de la calculadora. Cuando la calculadora se esté haciendo trabajar en un modo de programa múltiple ("multiprograma") como, por ejemplo, compartida en el tiempo, pueden ponerse en acción los diversos programas de aplicaciones para diferentes usos, y por ello es deseable organizar la calculadora de manera que puede impedirse que uno de los programas de aplicaciones obtenga acceso a otro. Asimismo, aun cuando muchas veces ello sea necesario para inscribir en uno de los programas de aplicaciones la información procedente de uno de los programas de control, y para que el sistema sea capaz de transferir desde diversos puntos en un programa de control hasta unos puntos seleccionados en los programas de aplicaciones, el funcionamiento de la calculadora podría trastornarse gravemente si los programas de control fuesen alterados por uno de los programas de aplicaciones, o si, por error de programación, se efectuara una transferencia desde un programa de aplicaciones a uno de los programas de control. Una transferencia espuria introducida en medio de un programa de control podría originar en éste alteraciones que trastornarían gravemente el funcionamiento normal de la calculadora.

En el pasado, el problema de impedir que se trasla-



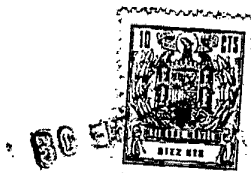
daran o inscribieran datos de uno de los programas a otro, o que se traspasara el control de un programa a otro de manera no deseada, se ha venido tratando por el recurso de disponer una clave o combinación de bitios de identificación por cada programa, e impedir el acceso al programa de todos los programas que no poseyeran esta clave o combinación de bitios. Este vinculación de la característica de protección de la memoria a la propia memoria de la calculadora impone serias limitaciones de la manera en que los programas pueden agruparse para permitir y negar el acceso. Es posible alcanzar una mucho mayor flexibilidad de control de la manera en que se vaya a dar acceso de unos programas a otros, disponiendo controles lógicos y memoria por separado, para la característica de protección de la memoria. Esta separación de control lógico y de memoria dejaría permitir o negar el acceso de un programa cualquiera deseado a otro programa cualquiera, y permitiría también hacer distinciones entre accesos de diversos tipos. Por ejemplo, podría permitirse a uno de los programas efectuar inscripciones de datos en otro programa, pero no transferir el control a ese programa.

Por todo ello, es objeto principal de esta invención un sistema perfeccionado de protección de la memoria.

Un objeto más concreto y específico de esta invención reside en un sistema de protección de la memoria, que permite una mayor flexibilidad de control de las interacciones entre diversos programas.

Un objeto todavía más concreto de esta invención reside en un sistema de protección de la memoria, que tiene la facultad de distinguir entre los diversos tipos de acceso

335302



a un programa de calculadora y de permitir un tipo de acceso entre dos programas, negando al mismo tiempo otros tipos de acceso.

5 Es característica de esta invención la provisión de una memoria y control lógicos por separado para efectuar la función de protección de la memoria.

10 Conforme a los objetos expuestos más arriba, esta invención proporciona una memoria lógica que contiene por lo menos un asiento de entrada o introducción por cada bloque lógico de información de la memoria de la calculadora. Este asiento indica si el bloque asociado tiene el privilegio de efectuar accesos de diversos tipos a otros bloques, y también si el bloque asociado está condicionalmente protegido contra accesos de diversos tipos, procedentes de otros bloques. Cuando en la memoria lógica exista más de un asiento para un bloque de información, los asientos contienen un campo adicional de identificación del programa que está en marcha. Por consiguiente, un bloque de información puede estar condicionalmente protegido en cuanto a ciertos tipos de acceso para un determinado programa, y sea sin proteger, sea condicionalmente protegido en cuanto a accesos de otros tipos para otro programa.

15 Al reconocerse una instrucción que exija acceso, se efectúa la determinación de si el bloque que contiene esta instrucción es un bloque "privilegiado". Entonces se investiga el asiento en la memoria lógica para el bloque que contiene la "dirección" a la cual se va a hacer el acceso y el programa en marcha, y se efectúa la determinación de si el bloque está condicionalmente protegido para el tipo de acceso que el programa vigente o en marcha está efectuando.

30



Si se descubre que la instrucción de acceso no es privilegiada, y que el bloque al que se está obteniendo acceso está condicionalmente protegido, se genera entonces una situación de interrupción. Si se da una de las otras tres condiciones que pueden aparecer, se permite el acceso.

Los indicadores y otros objetos, rasgos característicos y ventajas de la invención se irán desprendiendo de la siguiente descripción pormenorizada de una forma preferida de realización del invento, ilustrada en el dibujo adjunto, cuya única figura es un esquema funcional o de conjunto de una forma de ejecución ilustrativa del invento.

Con referencia ahora a la figura, puede verse que el sistema incluye un dispositivo o unidad central de tratamiento (CPU) 10 que lleva asociada una memoria principal 12. La CPU 10 puede ser una calculadora numérica normal cualquiera de uso general, de las muchas que hay. La memoria 12 puede ser, por ejemplo, una matriz o disposición regular de memoria de núcleos magnéticos. La memoria principal 12 contiene, en su sección 12A, uno o más programas de control, y en la sección 12B uno o más programas de aplicaciones. Las partes o secciones 12A y 12B de la memoria 12 pueden además subdividirse en bloques que contengan un número igual de asientos o entradas, conteniéndose un programa dado en uno o más de los bloques.

El sistema de la invención contiene asimismo una memoria de control 14 que tiene por lo menos un asiento por cada uno de los bloques de la memoria principal 12. Cuando se tiene acceso a un vocablo de uno de los bloques de la memoria principal 12, bajo el control de un programa en marcha en la CPU 10, se aplican también señales por medio de



las líneas 16 a la memoria de control 14, haciendo que el correspondiente asiento que hay en ella salga por las líneas 18 y pase el registro de datos de la memoria (MDR) 20. Por el MDR 20 se ve que, para la forma ilustrativa de realización del invento, cada asiento en la memoria de control 14 contiene seis campos. Leyendo de izquierda a derecha, estos campos son: un campo ID de programa, de varios bitios, que identifica el programa vigente o en marcha con el cual está sociado el asiento del bloque; un campo de direcciones de bloque, de varios bitios; un campo de un solo bitio, designado campo S1, que se activa o pone a 1 para aquellos bloques que contienen programas, que están condicionalmente protegidos respecto a la inscripción; un campo de un solo bitio, designado campo S2, que se pone a 1 si el programa del bloque correspondiente tiene el privilegio de alterar bloques condicionalmente protegidos; un campo de un solo bitio, designado campo S3, asociado con aquellos bloques a los cuales no está permitido efectuar transferencias como no sea desde instrucciones de ramal contenidas en un bloque con privilegio de transferencia (esto es, con los bloques que están condicionalmente protegidos en cuanto a transferencia); y un campo de un solo bitio, designado campo S4, que se pone a 1 para aquellos bloques con privilegio de transferencia. Cuando varios de los programas de la memoria 12 tienen las mismas características de protección, puede emplearse un código común de ID (identificación) de programa para designar todos ellos; y en la memoria de control 14 puede usarse, por cada bloque de la memoria 12 y para todos los programas, un solo asiento que lleve en su campo de más a la izquierda el código común de ID de programa. A los fi-

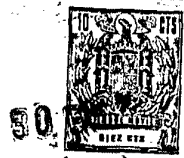
335302

80 ENE



5 nes de esta ilustración, se supondrá que los bitios S1 a S4 inclusive, asociados a los bloques destinados a los programas de control en la sección 12A de la memoria principal 12, están todos puestas a 1, indicando con ello que los programas de control son al mismo tiempo protegidos y privilegiados en cuanto a transferencia, en tanto que los bloques destinados a los programas de aplicación en la sección 12B de la memoria 12 tienen puesto a 1 solamente el campo S1, indicando así que estos bloques están condicionalmente protegidos en cuanto a inscripción, pero no están protegidos en cuanto a transferencia, ni tienen privilegio para inscribir o transferir.

10 Con referencia de nuevo a la CPU 10, puede verse que ésta, además de las salidas ya mencionadas, tiene un "reloj" electrónico que, cada vez que se efectúa un acceso a la memoria principal 12, hace que se apliquen unos impulsos en secuencia a las líneas de reloj 31 a 34 inclusive. Las líneas 31 a 34 se designan también como líneas T1 a T4 inclusive, respectivamente. Para mayor sencillez del dibujo, no se ha intentado conectar las líneas 31-34 a los diversos puntos del circuito en los cuales se utilizan. En lugar de ello, en cada uno de estos puntos aparece una línea que lleva la designación de letra y número apropiada. La CPU 10 genera también unas señales de salida por las líneas 41 a 43 inclusive, que están conectadas de modo que activan a su estado de "uno" unos multivibradores biestables respectivos, 46 a 48 inclusive. El biestable ("flip-flop") 46 se designa también FF A, y se activa a su estado de "uno" cuando la instrucción que esté ejecutando la CPU 10 sea una referencia activa (de guardar). En otros términos, el biesta-



ble 46 se halla en su estado de "uno cuando se está alterando el contenido de un vocablo de la memoria principal 12. El biestable 47 se designa también FF I, y se halla en su estado de "uno" cuando se está ejecutando una búsqueda de instrucción. La búsqueda de instrucción precederá siempre a toda instrucción que incluye una instrucción del tipo de almacenar, o guardar. Por consiguiente, existirá siempre un ciclo I (esto es, el FF I estará en "uno") antes de que haya un ciclo A (se ponga en "uno" el FF A). El biestable 48 se designa también FF X, activándose a su estado de "uno" para el ciclo que sigue a la realización por parte de la CPU 10, de una instrucción de ejecutar. La instrucción de ejecutar es aquella que exige que se ponga en práctica la instrucción existente en una dirección de acceso especificada de la memoria 12. Difiere, no obstante, de una instrucción de ramal, en que, una vez efectuada o ejecutada la instrucción en la dirección de acceso especificada y concreta, se devuelve el control del sistema a la instrucción que sigue a la de ejecutar, y no a la que sigue a la dirección de acceso especificada por la instrucción de ejecutar.

FUNCIONAMIENTO

Para describir el funcionamiento del sistema, supóngase primero que está en marcha un programa de aplicaciones y que, por alguna razón, este programa intenta efectuar una inscripción en uno de los programas de control. Como antes se ha indicado, es ésta una operación no deseada, y que ha de dar lugar a una supresión o interrupción. Como primer escalón del funcionamiento, se busca la instrucción procedente del programa de aplicaciones, en la sección 128 de la memoria principal 12, y se aplica por medio de líneas 50 a



la CPU 10. Como se está efectuando una búsqueda de instrucción, se aplica a la línea 42 de I una señal que active a su estado de "uno" el biestable I 47. También se aplican señales, por las líneas 16, a la memoria de control 14, haciendo que el asiento de la misma, correspondiente al programa en marcha, y el bloque de la memoria principal 12 que contiene la instrucción arriba indicada, pasen por lectura al MDR 20. Si la memoria de control 14 es asociativa, las señales presentes en las líneas 16 pueden dar lugar, por ejemplo, a que se efectúe una operación asociada en los campos de direcciones de bloque e ID de programa, pasando al MDR 20 al asiento que tenga unos campos de direcciones de bloque e ID de programa correspondientes. Como el bloque que contiene la instrucción está en la sección de aplicaciones 12B de la memoria principal 12, el asiento pasado al MDR 20 en este momento contiene un bitio 1 en el campo S1 y bitios 0 en los campos S2 a S4 inclusive.

Terminadas las operaciones preliminares arriba descritas, la CPU 10 aplica una señal a la línea 31 de T1. La señal que aparece en la línea 31 de T1 es aplicada de modo que activa a su estado de "uno" el biestable 52, y condiciona las barreras de coincidencia 54 y 56. Las demás operaciones que se efectúan bajo el control de la línea 31 de T1 no hacen al caso en este momento, y se describirán más adelante. Como el biestable 47 de I se halla en su estado de "uno", por medio de la línea 58 de salida del lado de "uno" de este biestable se aplica una señal a una segunda entrada de las barreras de coincidencia 54 y 56. De igual modo, por hallarse el biestable 48 de X en su estado de "cero", se da origen a la aplicación de una señal proceden-



te de este biestable, por la línea 60 de salida del lado de "cero" del mismo, a una tercera entrada de las barreras de coincidencia 54 y 56. Como el campo S2 del MDR-20 está a 0 en este momento, el inversor 62 aplica una señal a la cuarta entrada de la barrera de coincidencia 56, condicionándose por completo esta barrera y generándose en la línea 64 una señal de salida que es aplicada para poner el biestable 66 de P1 a su estado de "cero".

La señal presente en la línea 31 de T1 va seguida de una señal en la línea 32 de T2, señal que es aplicada como entrada a la barrera de coincidencia 68. Ahora bien, como el biestable A.46 se halla en su estado de "cero", la barrera de coincidencia 68 no se condiciona por completo. Como los demás puntos del circuito a los cuales se aplica la línea 32 de T2 no hacen al caso en este momento, no se efectúan operaciones en el tiempo T2. La señal presente en la línea 32 de T2 va seguida de una señal por la línea 33 de T3, aplicada para condicionar la barrera 70. Los demás puntos del circuito a los que se aplica la línea 33 de T3 se describirán más adelante. Como el biestable 52 se puso al estado de "uno" en el tiempo T1, la barrera 70 está condicionada de modo que deja pasar la señal presente en la línea de salida 72 del lado de "uno" de este biestable, por la barrera disyuntiva 74, hasta la línea 76. La línea 76 se aplica a la CPU 10, haciendo que prosiga la secuencia normal del programa. La señal presente en la línea 33 de T3 va seguida de una señal por la línea 34 de T4 que, a los fines del presente estudio, sólo sirve para reponer los biestables 46 a 48 inclusive a su estado de "cero".

Cuando la instrucción está en la CPU 10, se descodi-



5      fica. Supóngase que la instrucción es la de "guardar el con-  
tenido del acumulador en la dirección N", siendo N una di-  
rección de acceso de la sección 12A de la memoria princi-  
pal. Una vez descodificada esta operación, se aplica por la  
10     línea 41 una señal para poner el biestable A 46 a su estado  
de "uno", y por las líneas 16 se aplican señales para pasar  
al MDR 20, por las líneas 18, el asiento de la memoria de  
control 14 correspondiente a la identificación (ID) del pro-  
grama en curso, y el bloque de la memoria principal 12 que  
15     contiene la dirección en la que se va a efectuar el almace-  
naje ordenado. A los fines de este ejemplo se ha supuesto  
que la dirección en donde se va a guardar la información es-  
tá en la parte de control 12A de la memoria principal 12.  
Como se recordará, los asientos en esta parte de la memoria  
15     tienen bitios 1 en los campos S1 a S4 inclusive. Terminadas  
las operaciones primarias, la CPU 10 aplica una señal a la  
línea 31 de T1. Como antes, sólo se mencionarán, de los pun-  
tos del circuito a los que se aplican las señales presen-  
tes en las líneas 31 a 34, aquellos que correspondan al pre-  
20     sente análisis. La señal presente en la línea 31 de T1 se  
aplica para volver a poner a "uno" el biestable 52, y tam-  
bién como entrada condicionante a las barreras de coinci-  
dencia 54 y 56.

25             Como el biestable I 47 está en "cero" en este momen-  
to, no hay señal alguna en la línea 58 ni, por tanto, se  
condicionan las barreras de coincidencia 54 ó 56. La señal  
de la línea 31 de T1 va seguida de una señal en la línea  
32 de T2, que se aplica como entrada a la barrera de coinci-  
dencia 68. El biestable A 46, por estar a "uno" en este mo-  
30     mento, hace que por la línea 78 se aplique una señal a una

335302



una segunda entrada de la barrera de coincidencia 68. Como se recordará, el biestable 66 de P1 se puso a "cero" durante el ciclo anterior. Por consiguiente, por la línea de salida 80 del lado de "cero" de este biestable se aplica una señal a una tercera entrada de la barrera de coincidencia 68. La entrada final de la barrera de coincidencia 68 es la línea de salida 82 que viene del campo S1 del MDR 20. Como en este campo hay un bitio 1, la barrera de coincidencia 68 se condiciona por completo en el tiempo T2, generado por la línea 84 una señal de salida que es aplicada para llevar al biestable 52 a su estado de "cero".

La señal que aparece en la línea 32 de T2 va seguida de una señal por la línea 33 de T3, que se aplica para condicionar la barrera 70. Como el biestable 52 se halla ahora a "cero", está dando una señal por su línea de salida 86 del lado de "cero". Esta señal se aplica a la línea 90 por medio de la barrera condicionada 70 y de la barrera disyuntiva 88. La señal de la línea 90 se aplica a la CPU 10 haciendo que tenga lugar una interrupción. Cuando la CPU 10 haciendo que tenga lugar una interrupción. Cuando la CPU 10 reconoce la condición de interrupción, pide una subrutina de un programa de control, para devolver el control del programa a un determinado lugar, y volver a iniciarlo. La señal de la línea 33 de T3 va seguida de una señal que aparece en la línea 34 de T4, señal que es aplicada para poner a su estado de "cero" el biestable A 46.

Para ilustrar aún más el funcionamiento del circuito, supóngase ahora que la instrucción de "guardar el contenido del acumulador en la dirección N" se origina en un programa de control de la sección 12A de la memoria princi-



pal 12, y que la dirección N está en la sección 12B de la memoria. En estas condiciones, para el primer ciclo del reloj de la CPU, el biestable I 47 está en "uno", y hay bitios 1 en los campos S1 a S4 inclusive del MDR 20. En el

5 instante T1, esto hace que se condicione plenamente la barrera de coincidencia 54, generando por la línea 92 una señal de salida que se aplica para poner a "uno" el biestable 66 de P1. En el tiempo T1, se activa también a su estado de "uno" el biestable 52. Como el biestable A 46 se halla a

10 "cero", no ocurre nada en el tiempo T2. En cambio, en el tiempo T3 se condiciona la barrera 70, dejando pasar la señal que hay en la línea de salida 72 del lado de "uno", desde el biestable 52, por medio de la barrera disyuntiva 74, hasta la línea 76 de "proseguir con el programa". Esto

15 permite a la CPU 10 proseguir con la secuencia de programa normal. Durante el segundo ciclo del reloj de la CPU, el biestable A 46 está a "uno", hay un bitio en el campo S1 del MDR 20, y los restantes campos S del MDR están a 0. En estas condiciones, la señal de T1 vuelve a poner a "uno" el

20 biestable 52, pero no llega a producir alteración alguna en la posición del biestable 66 de P1. Este biestable P1 66, por consiguiente, sigue en su estado de "uno". En el tiempo T2 hay señales en las líneas 32, 78 y 82. Ahora bien, como el biestable 66 de P1 está a "uno", no hay señal alguna

25 en la línea 80 de salida del lado de "cero" de este biestable. Por consiguiente, a la barrera de coincidencia 68 le falta una de sus entradas, y el biestable 52 se queda, por lo tanto, en su estado de "uno". En el tiempo T3 se vuelve a condicionar la barrera 70, dejando pasar la

30 ñal que hay en la línea de salida 72 del lado de "uno" del



biestable 52, por la barrera disyuntiva 74 y la línea 76 de "proseguir con el programa", hasta la CPU 10. La operación se concluye con una señal, en la línea 34 de T4, que repone a "cero" el biestable A 46.

5                   Por lo que antecede se ha visto de que modo tiene lugar una interrupción cuando la instrucción procedente de un bloque no privilegiado trata de efectuar una inscripción en un bloque protegido, y cómo al bloque que tiene privilegio de inscripción se le permite inscribir en un bloque protegido en cuanto a la inscripción. También puede verse  
10 que cualquiera de los programas de control de la sección 12A de la memoria principal, programas que tienen privilegio de inscripción, puede inscribir en cualquiera de los demás programas de control, aun cuando estos programas estén  
15                   condicionalmente protegidos en cuanto a inscripción, y también inscribir en los programas de aplicaciones condicionalmente protegidos. Igualmente, ninguno de los programas de aplicaciones, que no tienen privilegio de inscripción, puede inscribir en ninguno de los demás programas  
20 de aplicaciones que estén condicionalmente protegidos en cuanto a inscripción.

                  Los bitios de situación o estado de los campos S3 y S4 de la memoria de control 14 y del MDR 20 se utilizan para ilustrar un modo ligeramente distinto de operación de  
25 proteger. Para ilustrar de qué modo se utilizan estos bitios de estado, supóngase primero que está en curso uno de los programas de aplicaciones de la sección 12B de la memoria principal, y que aparece una instrucción de "ramificar" (o de ramal) que da origen a una ramificación o "conexión"  
30 a uno de los programas de control de la sección 12A. En



estas condiciones, cuando se va a buscar la instrucción a la memoria 12, la CPU 10 aplica por la línea 42 una señal para poner a "uno" el biestable I 47, y por las líneas 16 aplica a la memoria de control 14 unas señales que hacen que el asiento correspondiente a la ID del programa en curso, y el bloque de la memoria principal 12 que contiene la instrucción de ramificar, pasen por las líneas 18 a MDR 20. Como antes se ha indicado, se supone que el asiento en la memoria de control 14, correspondiente a un bloque de la sección 12B de la memoria principal, tiene puesto a 1 su bitio de estado S1, y puestos a 0 los restantes bitios de estado. Cuando la CPU 10 ha completado las operaciones preliminares, aplica a la línea 31 de T1 una señal que, además de las funciones anteriormente mencionadas, se aplica también, como entrada, a la barrera de coincidencia 96. Una segunda entrada de la barrera de coincidencia 96 es la línea de salida 98 del campo S3 del MDR 20. Como este campo está puesto a 0 en este momento, no se condiciona la barrera de coincidencia 96, ni se efectúa operación alguna en el tiempo T1.

La señal de la línea 31 de T1 va seguida de una señal por la línea 32 de T2, que es aplicada como una de las entradas a las barreras de coincidencia 100 y 102. La línea de salida 58 del lado de "uno" del biestable I 47 y la línea de salida 60 del lado de "cero" del biestable X 48 se aplican, como otras dos entradas, a las barreras de coincidencia 100 y 102. La entrada final a la barrera de coincidencia 100 es la línea de salida 104 procedente del campo S1 del MDR 20, y la entrada final a la barrera de coincidencia 102 es la línea de salida 106 procedente del inver-

335302



5           sor 108, mientras la entrada al inversor 108 es la mencio-  
nada línea 104. Como el campo S4 del MDR está a 0 en este  
momento, el inversor 108 está generando por la línea 106  
una señal de salida que condiciona por completo la barrera  
de coincidencia 102, dando lugar a una señal de salida por  
la línea 110, que es aplicada para poner a su estado de  
"cero" el biestable P2 112.

10           La señal de la línea 32 de T2 va seguida de una se-  
ñal por la línea 33 de T3, aplicada para condicionar la  
barrera 114. Se supone que el biestable 116 se puso a su  
estado de "uno" durante un tiempo T4 anterior. Como este  
biestable no se alteró durante el actual ciclo de reloj, se  
halla ahora generando una señal de salida por la línea de  
salida 18 del lado de "uno", señal que es aplicada a la CPU  
15 10 por medio de la barrera condicionada 114, la barrera dis-  
yuntiva 74 y la línea 76 de "proseguir con el programa",  
permitiendo que continúe el programa en curso en aquella.

20           La señal que aparece en la línea 33 de T3 va seguida  
de una señal por la línea 34 de T4, que se aplica a la en-  
trada del lado de "uno" del biestable 116, y a la entrada  
del lado de "cero" del biestable I 47.

25           Se ha supuesto que la instrucción de ramificar, una  
vez decodificada, produce una transferencia a una instruc-  
ción contenida en la sección de control 12A de la memoria  
principal 12. Antes de ejecutar esta operación, por lo tan-  
to, la CPU aplica por la línea 42 una señal, para poner a  
"uno" el biestable I 47, y por las líneas 16 unas señales  
para leer o tomar el asiento de la memoria de control 14  
que corresponda a la ID del programa en curso y al bloque  
30 que contiene la instrucción a la que se está ramificando. Co

335302



mo se recordará, los asientos para los bloques de la parte de control 12A tienen todos sus bitios de estado puestos a 1. Completadas todas estas operaciones preliminares, se vuelve a aplicar una señal por la línea 31 de T1, a una de las entradas de la barrera de coincidencia 96. La señal de la línea de salida 98 procedente del campo S3 del MDR 20 se aplica como segunda entrada a la barrera de coincidencia 96, y la señal de la línea de salida 58 del lado de "uno" del biestable I se aplica a esta misma barrera de coincidencia como tercera entrada. Como el biestable P2 112 se puso a su estado de "cero" durante el ciclo precedente en la línea de salida 120 del lado de "cero" de este biestable hay una señal que se aplica para condicionar plenamente la barrera de coincidencia 96 y generar una señal de salida por la línea 122. La señal de la línea 122 se aplica para reponer a su estado de "cero" el biestable 116. La señal de la línea 31 de T1 va seguida de una señal por la línea 32 de T2 que, en unión de la señal que aparece en la línea de salida 58 del lado de "uno" del biestable I 47, de la señal que hay en la línea de salida 60 del lado de "cero" del biestable X 48, y de la señal presente en la línea de salida 104 procedente del campo S4 del MDR, condiciona plenamente la barrera de coincidencia 100 generando por la línea 124 una señal de salida que se aplica para activar o poner a su estado de "uno" el biestable 112 de P2. La activación de este biestable P2 a su estado de "uno" en este momento, indica que la instrucción que se está ejecutando ahora es una instrucción privilegiada en cuanto a transferencia. La señal de la línea 32 de T2 va seguida de una señal por la línea 33 de T3, señal que se aplica para condi-

335302



5 cionar la barrera 114. Como el biestable 116 está ahora  
en "cero", aparece una señal por la línea de salida 126  
del lado de "cero", señal que es aplicada por la barrera  
114, la barrera disyuntiva 88 y la línea 90 a la CPU 10,  
10 haciendo que se produzca una interrupción. De esta manera  
se impide la transferencia no deseada de información de con-  
trol a la sección 12A de la memoria principal 12. La señal  
de la línea 33 de T3 va seguida de una señal por la línea  
34 de T4, que se aplica activando el biestable 116 a su es-  
tado de "uno" y reponiendo a su estado de "cero" el bies-  
table I 47.

15 Supóngase ahora que la instrucción ejecutada durante  
el segundo ciclo de reloj arriba descrito, instrucción que  
tenía privilegio en cuanto a transferencia, era en sí una  
instrucción de ramificar, y que el asiento o entrada a es-  
ta instrucción era tal que no se producía interrupción. En  
estas condiciones, al final del tiempo T4, el biestable P2  
112 y el biestable 116 estarían ambos a "uno". Al estar a  
punto de ejecutarse la instrucción a la cual se ha pasado,  
20 o hecho la ramificación, se aplica una señal por la línea  
42 para poner a "uno" el biestable I 47, y se aplican por  
las líneas 16 unas señales que dan lugar a la toma o sali-  
da, por lectura, del asiento apropiado de la memoria de con-  
trol 14. En el momento T1 se vuelve a aplicar una señal con-  
25 dicionante a la barrera de coincidencia 96. La señal de la  
línea de salida 58 del lado de "uno" del biestable I 47  
aplica una segunda entrada a esta barrera de coincidencia.  
Si la instrucción a la que se ha pasado es una instrucción  
de control que haya en la sección 12A de la memoria, existe  
30 también señal en la línea de salida 98 del campo S3 del MDR,

335302

80 ENE



señal que se aplica como entrada adicional a la barrera de coincidencia 96. Si se pasa a una instrucción no protegida en cuanto a transferencia, procedente de la sección 12B de la memoria principal 12, no hay señal alguna en la línea 98 en este momento. En uno y otro caso, como el biestable P2 está a "uno", la barrera de coincidencia 96 no se condiciona por completo, y el biestable 116 permanece en su estado de "uno". En el tiempo T2 se vuelve a aplicar una señal por la línea 32 a las barreras de coincidencia 100 y 102, haciendo que el biestable P2 se ponga a uno u otro de sus estados, "uno" o "cero", según la instrucción a punto de ser ejecutada tenga o no privilegio de transferencia (esto es, según haya o no un bitio en el campo S4 del MDR 20). En el tiempo T3 se vuelve a condicionar la barrera 114, dejando pasar a la CPU la señal de la línea de salida 118 del lado de "uno" del biestable 116, por la barrera disyuntiva 74 y la línea 76, para dejar que se ejecute la instrucción a la que se va, y prosiga el programa. En el tiempo T4 se vuelve a aplicar señal por la línea 34 a la entrada del lado de "uno" del biestable 116, y a la entrada del lado de "cero" del biestable I 47. Por lo que antecede puede verse que cuando la instrucción de ramificar esté en un bloque privilegiado en cuanto a transferencia, la instrucción a la cual se pasa es ejecutada, ya se halle en un bloque protegido de transferencia o no.

Se han analizado hasta aquí tres de las cuatro condiciones posibles, que pueden surgir con los bitios de control de transferencia. La cuarta condición posible se produce cuando una instrucción no privilegiada en cuanto a transferencia da lugar a una ramificación hasta una instrucción no



protegida contra transferencia. En estas condiciones, en  
 el tiempo T2 del ciclo durante el cual se está buscando  
 la instrucción no privilegiada para transferir, hay un bi-  
 5        bitio 0 en el campo S4 del MDR 20 y, por lo tanto, la barme-  
       rra de coincidencia 102 se condiciona plenamente para poner  
       a "cero" el biestable P2 112. Ahora bien, en el tiempo T1  
 10       del ciclo siguiente hay un bitio 0 en el campo S3 del MDR  
       20 y la barrera de coincidencia 96, por lo tanto, no se  
       condiciona plenamente. Por consiguiente, el biestable 116  
 15       sigue en "uno", haciendo que la barrera 114, en el tiempo  
       T3, aplique una señal por la barrera disyuntiva 74 y la lí-  
       nea 76 a la CPU 10, y dé lugar a que se ejecute la instruc-  
       ción a la que se ha ramificado, y que prosiga el programa.

En el estudio hasta aquí realizado, los bitios de  
 15       estado S1 a S4 se han puesto o activado de una manera para  
       todos los programas de control de la sección 12A de la me-  
       moria principal, y se han puesto de otra manera para todos  
       los programas de aplicaciones de la sección 12B de la memo-  
       ria. Ahora bien, esto se ha hecho meramente para simplifi-  
 20       car los ejemplos ilustrativos, y el sistema es, de hecho,  
       capaz de una mucho mayor flexibilidad de funcionamiento.  
       Por ejemplo, supóngase que hay dos programas de aplicaciones,  
       designados programa 1 y programa 2, ninguno de los cuales  
       es privilegiado en cuanto a inscripción o transferencia, y  
 25       que en la sección 12B de la memoria hay un bloque de datos,  
       designado bloque 3, que puede ser utilizado por el progra-  
       ma 1, pero no por el programa 2, Por consiguiente, para el  
       bloque 3 habría dos asientos en la memoria de control 14:  
       uno con una ID de programa para el programa 1, en su campo  
 30       de identificación de programas, y el otro con la ID de pro

335302



grama 2 en este campo. El asiento para el programa 1 tendría ceros en sus campos S1 y S3 (protegidos en cuanto a inscripción y en cuanto a transferencia, respectivamente), y el asiento para el programa 2 tendría unos en estos campos. De esta manera el programa no privilegiado 1 se le da acceso al bloque 3, acceso que se le niega al programa no privilegiado 2.

En la descripción que antecede se ha hecho notar asimismo que la posición de ajuste del biestable P1 66 o del biestable P2 112 se ha alterado solamente cuando el biestable X 48 está a "cero". Para comprender la razón a que esto se debe, es necesario investigar la función del biestable X. Como se ha indicado anteriormente, la CPU 10 es capaz de engendrar instrucciones de ejecución. Una instrucción de este tipo exige que se efectúe la instrucción en la dirección de acceso N y que, una vez ejecutada esta inscripción, se devuelva el control a la instrucción que sigue a la de ejecutar. En estas condiciones, es la naturaleza de privilegiada de la instrucción de ejecutar, y no la naturaleza de privilegio de la instrucción que hay en la dirección N, la que tiene el control a los fines de la protección. Por ejemplo, si la instrucción de ejecutar no está en un área privilegiada en cuanto a inscribir, el biestable P1 22 se pondría a su estado de "cero" durante la investigación de esta instrucción. Cuando la instrucción de la dirección N está a punto de ejecutarse, se aplican señales a ambas líneas 42 y 43 para activar a su estado de "uno" los biestables I y X. Por consiguiente, aun cuando la dirección N esté en un área privilegiada, el biestable P1 no se pone a "uno". Si la instrucción de la dirección N hace que se ejecute una



operación de almacenar en activo, y la dirección a guardar o almacenar está en un área condicionalmente protegida, la puesta a "uno" del biestable P1 antes de efectuarse la interrupción de ejecutar da origen a que se produzca una interrupción, de la manera anteriormente descrita. El biestable X funciona de igual manera respecto a los bitios de estado S3 y S4.

Por lo que antecede puede verse que, ajustando o activando adecuadamente los bitios de estado del asiento correspondiente en la memoria de control 14, puede darse acceso a cualquier bloque de la memoria principal 12, o negarse al acceso a cualquier otro bloque de la misma. Además, disponiendo varios grupos de bitios de estado, puede permitirse un determinado tipo de acceso entre dos bloques de la memoria principal, y negarse un acceso de otro tipo. Es evidente asimismo que, si bien se ha hablado, con fines ilustrativos, de la característica de protección en cuanto a inscribir y en cuanto a transferir, puede disponerse asimismo de otras formas de protección de acceso, utilizando los conceptos de esta invención.

Si bien la invención se ha ilustrado y descrito en particular haciendo referencia a una forma preferida de ejecución de la misma, se sobreentiende, para aquellas personas versadas en la materia, que pueden hacerse en ella los indicados y otros cambios de forma y de detalle sin por ellos salirse del ámbito ni apartarse del espíritu de la invención.

La presente solicitud que corresponde a la presentada en Estados Unidos de América, con fecha 7 de Enero de 1.966, bajo el Número 519.347, se acoge a los beneficios.

30 ENE.



del artículo 51 del vigente Estatuto sobre Propiedad Industrial.

N O T A

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

10 1.- Un dispositivo de protección de una memoria, el cual comprende: un primer medio para indicar si tiene privilegio una instrucción que da lugar a que se actúe sobre dicha memoria de una manera prefijada; un segundo medio para indicar si la parte de dicha memoria sobre la que se actúa de dicha manera prefijada está condicionalmente protegida; y medios capaces de responder a la aparición combinada de una indicación procedente de dicho primer medio indicador, de que dicha instrucción no es privilegiada, y a una indicación procedente de dicho segundo medio indicador, de que dicha área de memoria está condicionalmente protegida, generando una interrupción.

20 2.- Un dispositivo de protección para una memoria principal cuyos asientos están agrupados de una manera prefijada, sistema que comprende: una memoria de control que tiene un asiento por cada grupo de asientos de dicha memoria principal; medios, capaces de responder a un asiento procedente de dicha memoria de control para un primer grupo  
25 po que contiene una instrucción, para determinar si dicha

335302



instrucción es privilegiada; medios, capaces de responder a un asiento procedente de dicha memoria de control para un segundo grupo que contiene un asiento al que se tiene acceso por medio de dicha instrucción, para determinar si los asientos de dicho segundo grupo están condicionalmente protegidos; y medios capaces de responder a la determinación de que dicha instrucción no es privilegiada, y de que dicho segundo grupo están condicionalmente protegidos, generando una interrupción.

5  
10           3.- El dispositivo de la reivindicación 2, en el que cada asiento de dicha memoria de control contiene un campo que indica si los asientos del grupo correspondiente de la memoria principal están condicionalmente protegidos, y un campo que indica si los asientos tienen privilegio.

15           4.- El dispositivo de la reivindicación 3, en el que dichos medios determinativos del carácter de privilegio incluyen medios para "muestrear" o examinar el campo indicativo de privilegio del asiento indicado procedente de dicha memoria de control; y en el que dichos medios determinativos del carácter de condicionalmente protegidos incluyen medios para examinar dicho campo indicativo de la calidad de condicionalmente protegidos.

20           5.- El dispositivo de la reivindicación 4, en el que cada asiento de dicha memoria de control incluye un número de campos indicativos del carácter de condicionalmente protegidos y un número semejante de campos indicativos de privilegio, existiendo, por cada tipo de acceso a la memoria principal para el que se busca protección, un par de campos indicativos de la calidad de condicionalmente protegidos y del privilegio; y en el que dichos medios de examinar

30



o "muestrear" incluyen medios para examinar el par de campos adecuado, en relación con el tipo de acceso que se esté efectuando.

5 6.- El dispositivo de la reivindicación 3, en el que en dicha memoria de control existe una pluralidad de asientos por cada grupo de asientos de dicha memoria principal; y dicha pluralidad de asientos incluye un asiento por cada grupo de programas de dicha memoria principal que tienen características de protección semejantes.

10 7.- El dispositivo de la reivindicación 3, en el que dichos grupos son unos bloques, cada uno de los cuales contiene un número semejante de asientos.

15 8.- Un dispositivo de protección para una memoria principal cuyos asientos están agrupados de una manera prefijada, siendo algunos de los asientos de dicha memoria unas instrucciones que pueden ser buscadas y descodificadas, sistema que comprende: una memoria de control que tiene un asiento por cada grupo de asientos de dicha memoria principal; medios capaces de funcionar cada vez que se busca una instrucción en un grupo de la memoria principal, para  
20 "muestrear" o examinar un asiento de dicha memoria de control, correspondiente a dicho grupo, a fin de determinar si la instrucción es privilegiada; medios capaces de funcionar después de descodificada dicha instrucción, para  
25 examinar un asiento de dicha memoria de control, correspondiente al grupo que contiene el asiento al que se ha intentado el acceso por medio de dicha instrucción, a fin de determinar si el asiento al que se ha intentado el acceso está condicionalmente protegido; y medios capaces de  
30 funcionar, cuando la instrucción buscada no es privilegia-

335302



da y el asiento al que se ha intentado el acceso está condicionalmente protegido, generando una interrupción.

5           9.- El dispositivo de la reivindicación 8, que incluye medios capaces de responder a dichos medios determinativos del privilegio, guardando una indicación de si dicha instrucción buscada es privilegiada; sistema en el que cada asiento de dicha memoria de control contiene un campo indicativo de protección condicional, que es examinado por dichos medios determinativos de la calidad de condicionalmente protegidos; y en el que dichos medios generadores de interrupción actúan en respuesta a una indicación guardada, de que dicha instrucción buscada no es privilegiada, y al examen de una indicación de condicionalmente protegidos en dicho campo indicativo de protección condicional.

15           10.- El dispositivo de la reivindicación 9, en el que existen varios tipos de accesos que pueden hacerse a un asiento de dicha memoria principal; en el que cada asiento de dicha memoria de control tiene un campo indicativo de privilegio y un campo indicativo de protección condicional para cada tipo de acceso; sistema que incluye unos medios de registro o almacenaje para cada tipo de acceso, para indicar si dicha instrucción buscada es privilegiada respecto al tipo de acceso correspondiente; sistema en el que dichos medios capaces de funcionar cuando se busca una instrucción examinan todos los campos indicativos de privilegio de dicho asiento de la memoria de control, y activa dichos medios de registro con arreglo al contenido de los mismos; sistema en el que dichos medios determinativos de protección condicional examinan el campo indicativo de protección condicional apropiado para el tipo de acceso pedido por

335302



dicha instrucción buscada; y en el que dichos medios generadores de interrupción actúan en respuesta a una indicación procedente de los medios de registro o almacenaje para el tipo de acceso pedido por dicha instrucción buscada, y a la indicación procedente del campo indicativo de protección condicional examinado.

11.- El dispositivo de la reivindicación 10, en el que uno de los tipos de acceso protegidos es un acceso de inscripción.

12.- El dispositivo de la reivindicación 10, en el que uno de los tipos de acceso protegidos es un acceso de transferencia.

13.- El dispositivo de la reivindicación 9, en el que dicha instrucción buscada puede ser una instrucción del tipo ejecutivo, que hace que se busque otra instrucción; y que incluye medios para inhibir o evitar que se altere la indicación de dichos medios de registro o almacenaje a consecuencia de la búsqueda de la instrucción originada por dicha instrucción de tipo ejecutivo.

14.- El dispositivo de la reivindicación 8, en el que dicha memoria principal contiene una pluralidad de programas de los cuales por lo menos algunos tienen características de protección semejantes; en el que dicha memoria de control contiene, por cada uno de dichos grupos de asientos de dicha memoria principal, un asiento para el grupo de programas que tenga características de protección semejantes; y en el que los asientos de dicha memoria de control examinados por dichos medios determinativos de privilegio y de protección condicional son los asientos correspondientes al grupo de programa que contenga dicha instrucción buscada.

335302

80



15.- Un dispositivo de protección de una memoria.

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

5

Esta Memoria consta de veintiocho hojas, escritas a máquina por una sola de sus caras.

Madrid,

80 ENE 1907

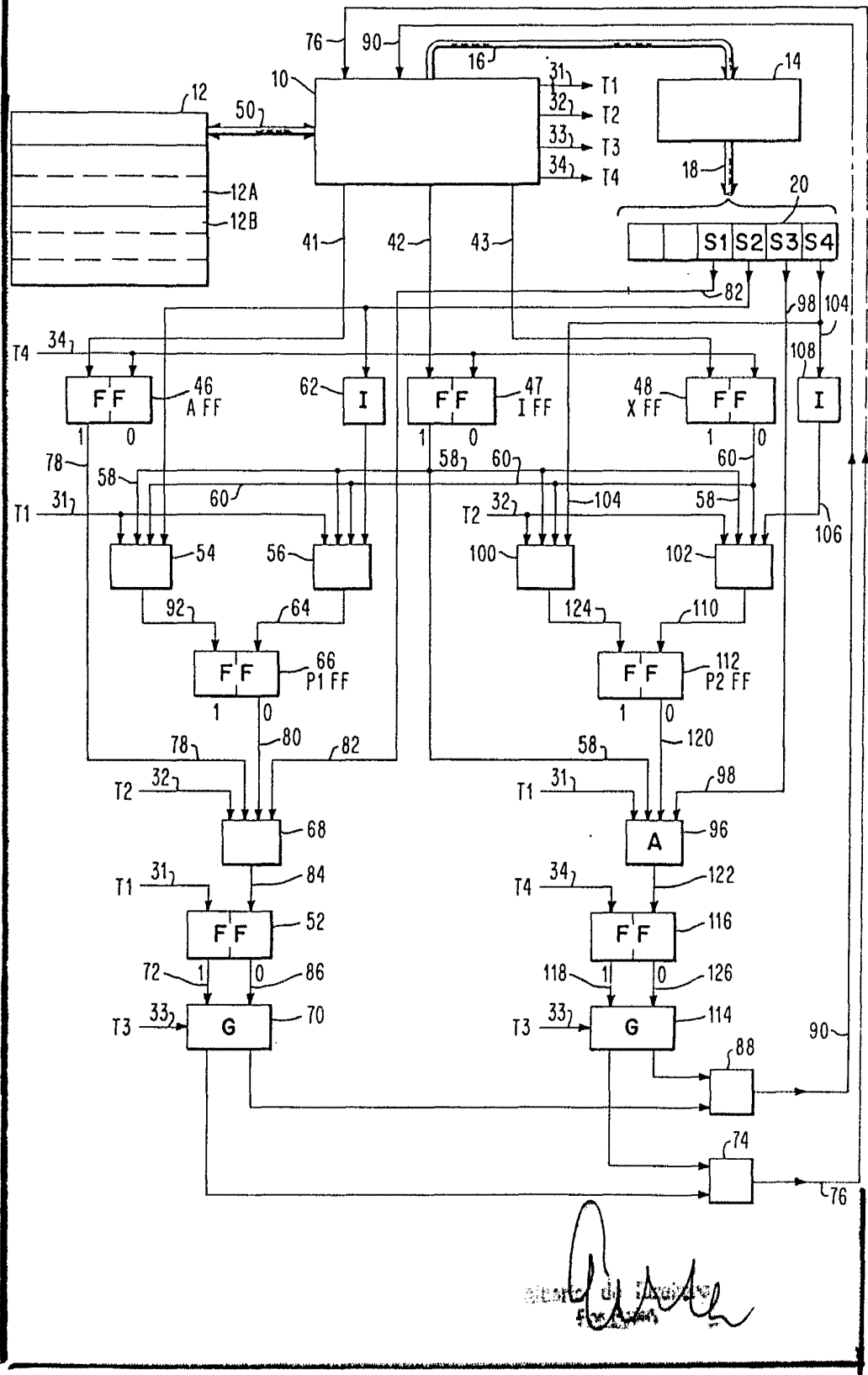
P. A.

Alberdo de Elzaburu

F. P. 2004

335302

335302



Invented by  
 \_\_\_\_\_  
 for IBM