(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 366 753**

(51) Int. Cl.:
H04L 9/06 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Número de solicitud europea: **07871853 .3**
(96) Fecha de presentación : **13.12.2007**
(97) Número de publicación de la solicitud: **2218208**
(97) Fecha de publicación de la solicitud: **18.08.2010**

(54) Título: **Método de procesamiento criptográfico de datos, en particular con la ayuda de una caja S, dispositivo y programas asociados.**

(73) Titular/es: **OBERTHUR TECHNOLOGIES**
50 Quai Michelet
92300 Levallois Perret, FR

(72) Inventor/es: **Rivain, Matthieu y**
Prouff, Emmanuel

(74) Agente: **Lehmann Novo, María Isabel**

(45) Fecha de publicación de la mención BOPI:
25.10.2011

(45) Fecha de la publicación del folleto de la patente:
25.10.2011

DESCRIPCIÓN

Método de procesamiento criptográfico de datos, en particular con la ayuda de una caja S, dispositivo y programas asociados

La invención se refiere a un método de procesamiento criptográfico de datos, así como un dispositivo y un programa asociados.

En dichos métodos, se suele recurrir al enmascaramiento de los datos con el fin de luchar contra los ataques, por ejemplo del tipo de análisis de corriente (en particular, los ataques de tipo DPA (Análisis de Potencia Diferencial)) *o* del tipo de análisis de radiación electromagnética.

Las técnicas de enmascaramiento consisten en combinar el dato (es decir, en la práctica, el número) que se desea utilizar (en la práctica, al que se desea realizar una operación) con un número imprevisible para un atacante exterior (en general, un número aleatorio o pseudo-aleatorio); así, los valores implicados son, cada vez, diferentes incluso utilizando un dato constante en la entrada, lo que hace imposible para el atacante deducir los datos internos del método (y, en particular, las claves criptográficas que utiliza) a partir de medidas realizadas desde el exterior.

Una parte de la seguridad criptográfica se obtiene mediante la utilización de funciones no lineales. Por ejemplo, se suele modelizar un cifrado por bloque *(block cipher)* mediante la combinación de funciones afines y de funciones no lineales. Las realizaciones de tales funciones no lineales son particularmente difíciles de proteger por enmascaramiento, debido a la no linealidad con respecto a la operación de enmascaramiento.

Un ejemplo del método de procesamiento criptográfico se describe por la solicitud de patente internacional WO 2007/116140 que pone en práctica una función no lineal del tipo caja S ("*S-Box",* según la terminología anglosajona especializada del dominio considerado) aplicada a estos datos enmascarados.

Dichas cajas S o "S-Boxes" se realizan, en la práctica, por medio de una tabla de correspondencia (que suele denominarse tabla-S o *"look-up table"* [LUT] en este mismo dominio) memorizada en el dispositivo criptográfico.

La solución propuesta en este documento puede, sin embargo, no convenir cuando se desee evitar la puesta en práctica de un número importante de adiciones.

En este contexto, la invención prevé un método de procesamiento criptográfico de datos representados bajo forma digital, poniéndose en práctica dicho método por una entidad electrónica y que comprende una transformación de un dato de entrada, enmascarado por una máscara de entrada, en un dato de salida, poniendo en práctica dicha transformación una tabla de conversión, caracterizado por las etapas siguientes:

- para al menos una pluralidad de valores posibles para la máscara de entrada, la transferencia del valor de salida de la tabla de conversión correspondiente al dato de entrada enmascarado, transformado aplicando una operación de desenmascaramiento por medio del valor posible, en una tabla en una posición correspondiente a un valor determinado enmascarado por la máscara de entrada y transformado aplicando la operación de desenmascaramiento por medio del valor posible;

- determinación del dato de salida por medio del dato situado en la tabla en la posición correspondiente al valor determinado.

Se reorganiza, así, una parte al menos de la tabla de conversión en la tabla, de tal modo que el valor de salida buscado (que corresponde, en esta tabla de conversión, al valor de entrada sin enmascaramiento) se coloque (con un posible enmascaramiento), en la tabla en la posición definida por el valor determinado.

Sin embargo, debido a que se trata de una pluralidad de valores posibles para la máscara y que la reorganización efectuada depende al menos de la máscara de entrada, el método está protegido contra los ataques.

La tabla de conversión define, por ejemplo, en la práctica, una función no lineal, tal como la implicada en el interior de una caja-S.

La máscara de entrada puede ser una máscara del primer orden, por ejemplo la aplicación de un valor aleatorio, o una máscara del segundo orden, que corresponde a la aplicación sucesiva de un primero y luego, un segundo valor aleatorio. En este caso, el valor determinado se puede enmascarar por la máscara de entrada por medio de las etapas siguientes:

- enmascaramiento por un primer elemento de máscara;

- enmascaramiento por un segundo elemento de máscara.

Con el fin de mejorar la seguridad en la salida de la tabla de conversión, la etapa de transferencia puede comprender el enmascaramiento del valor transferido mediante una máscara de salida, que puede ser diferente de la máscara de entrada, por ejemplo por medio de un enmascaramiento del primer o del segundo orden.

En este último caso, la aplicación de la máscara de salida se puede realizar por medio de las etapas siguientes:

- enmascaramiento por un primer valor aleatorio;
- enmascaramiento por un segundo valor aleatorio.

Según una forma de realización descrita a continuación, la etapa de transferencia se realiza para el conjunto de los valores posibles para la máscara de entrada, lo que permite no hacer aparecer, hacia el exterior ningún valor privilegiado y por lo tanto, ninguna fuga de información a este nivel.

Por otro lado, el valor determinado (que define, como se indicó anteriormente, la posición de interés en la tabla) puede ser un valor predeterminado (por ejemplo, con miras a la simplificación) o un valor obtenido mediante muestreo aleatorio, lo que mejora también la seguridad, puesto que la posición interesante, en la tabla, varía en este caso en cada puesta en práctica.

La entidad electrónica es adecuada para manipular los datos digitales procesados, por ejemplo, por medio de un microprocesador.

El método es, por ejemplo, un método de cifrado o de descifrado de datos digitales, que representa en general un mensaje, pero también potencialmente una clave criptográfica, un dato intermedio o una parte solamente de dichos elementos.

En la práctica se utilizará máscaras binarias de la misma longitud que el dato de entrada binaria. Así, el conjunto de las máscaras binarias corresponde al conjunto de las entradas de la caja-S (tabla-S).

Las máscaras binarias asociadas a una función de adición booleana tienen la ventaja de constituir una función de enmascaramiento involutivo simple, que permite encontrar el valor de un dato de entrada cuando está enmascarado dos veces por la misma máscara.

La invención se aplica así a cualquier orden de enmascaramiento, es decir, cualquiera que sea el número de máscaras aleatorias aplicadas al dato.

En particular, la función de enmascaramiento puede ser conmutativa, lo que permite aprehender el doble (incluso más) enmascaramiento cualquiera que sea el orden de aplicación de las máscaras de entrada o de salida, según se menciona más adelante.

Según un modo de realización particularmente interesante, dichas máscaras pueden ser máscaras aditivas, por ejemplo máscaras booleanas. Así, se puede poner en práctica fácilmente la adición booleana con la ayuda de funciones O EXCLUSIVA (XOR).

Según un modo de realización, que utiliza una palabra-máquina de gran longitud y que reagrupa varias sub-palabras de salida, dicho dato situado en la tabla es una palabra que comprende una pluralidad de sub-palabras de salida y la determinación del dato de salida puede comprender, entonces, una etapa de acceso al dato de salida, entre las sub-palabras, por medio de una parte del dato de entrada y de una parte de la máscara de entrada.

Dentro de este contexto, la etapa de acceso al dato de salida en dicha palabra comprende por ejemplo:

i) una etapa de separación de dicha palabra en dos mitades, respectivamente, de los bits más significativos y de los bits menos significativos,

ii) una etapa de selección de una de dichas mitades de dicha palabra en función de los valores de los bits de un mismo índice en, respectivamente, la parte del dato de entrada y la parte de la máscara de entrada.

El método se pone en práctica, por ejemplo, por una secuencia de instrucciones memorizadas en la entidad electrónica y ejecutarse por un microprocesador de la entidad electrónica.

La entidad electrónica puede ser, en la práctica, una tarjeta de microcircuito, particularmente adaptada a este tipo de operaciones aseguradas.

La invención da a conocer, asimismo, un dispositivo electrónico de procesamiento criptográfico de datos representados bajo la forma digital adecuada para realizar una transformación de un dato de entrada, enmascarado por una

máscara de entrada, en un dato de salida por medio de una tabla de conversión, caracterizado por:

- medios para transferir, para al menos una pluralidad de valores posibles para la máscara de entrada, el valor de salida de la tabla de conversión correspondiente al dato de entrada enmascarado, transformado por aplicación de una operación de desenmascaramiento por medio del valor posible, en una tabla en una posición correspondiente a un valor determinado enmascarado por la máscara de entrada y transformado por aplicación de la operación de desenmascaramiento por medio del valor posible;

- medios para determinar el dato de salida por medio del dato situado en la tabla en la posición correspondiente al valor determinado.

Este dispositivo puede incluir, además, características que corresponden a los modos de realización antes considerados para el método.

La invención da a conocer, por último, un producto de programa informático, que comprende una serie de instrucciones adecuadas, cuando se ejecutan por un microprocesador, para poner en práctica el método anteriormente citado.

Las ventajas, los objetos y las características particulares de este dispositivo electrónico y de este producto de programa informático, al ser similares a las del método igualmente objeto de la presente invención, tal como se expuso sucintamente con anterioridad, no se recuerdan en la presente descripción.

Otras ventajas, objetos y características particulares de la presente invención se harán evidentes a partir de la descripción dada a continuación, realizada con un objeto explicativo y nunca limitativo, con respecto a los dibujos adjuntos, en donde:

- la Figura 1 representa esquemáticamente los elementos principales de una forma de realización posible para una tarjeta de microcircuito;

- la Figura 2 representa la apariencia física general de la tarjeta de microcircuito de la Figura 1;

- la Figura 3 representa, bajo la forma de un logigrama, las etapas esenciales de un cifrado según el algoritmo AES con enmascaramiento;

- la Figura 4 representa, bajo la forma de un logigrama, un primer modo de realización de la invención puesto en práctica en el método representado en la Figura 3;

- la Figura 5 representa, bajo la forma de un logigrama, un segundo modo de realización de la invención puesto en práctica en el método representado en la Figura 3;

- la Figura 6 representa, bajo la forma de un logigrama, un tercer modo de realización de la invención puesto en práctica en el método representado en la Figura 3;

- las Figuras 7 y 8 ilustran un ejemplo de puesta en práctica de las tablas-S utilizadas en los modos de realización representados en las Figuras 4 a 6;

- la Figura 9 representa, bajo la forma de un logigrama, un primer modo de realización para el acceso a una sub-palabra de una palabra binaria, por ejemplo almacenada en la tabla-S de la Figura 8;

- la Figura 10 representa, bajo la forma de un logigrama, un segundo modo de realización para el acceso a una sub-palabra de una palabra binaria, por ejemplo almacenada en la tabla-S de la Figura 8 y

- la Figura 11 representa, bajo la forma de un logigrama, un tercer modo de realización para el acceso a una sub-palabra de una palabra binaria, por ejemplo almacenada en la tabla-S de la Figura 8.

Un ejemplo de entidad electrónica es una tarjeta de microcircuito 10, cuyos principales componentes electrónicos se representan en la Figura 1 y que contiene un microprocesador 2 conectado, de una parte, a una memoria viva (o RAM del inglés *Random Access Memory*-Memoria de Acceso Aleatorio*)* 4 y de otra parte, a una memoria de semiconductores no volátil regrabable 6, por ejemplo una memoria muerta borrable y programable eléctricamente (o EEPROM del inglés *Electrically Erasable Programable Read Only Memory)*. Como variante, la memoria no volátil regrabable de semiconductores 6 podría ser una memoria instantánea (‘flash’).

Las memorias 4, 6 están conectadas al microprocesador 2 por un bus cada una en la Figura 1; como variante, podría tratarse de un bus común.

La tarjeta de microcircuito 10 contiene, además, una interfaz 8 de comunicación con un terminal de usuario aquí

realizado bajo la forma de contactos, de los cuales uno asegura, por ejemplo, un enlace bidireccional con el microprocesador 2. La interfaz 8 permite así el establecimiento de una comunicación bidireccional entre el microprocesador 2 y el terminal de usuario, en donde se insertará la tarjeta de microcircuito 10.

De este modo, en el momento de la inserción de la tarjeta de microcircuito 10 en un terminal de usuario, el microprocesador 2 pondrá en práctica un método de funcionamiento de la tarjeta de microcircuito 10, según un conjunto de instrucciones, almacenadas por ejemplo en una memoria muerta (o ROM del inglés *Read-Only Memory*- Memoria de lectura solamente*)* - no representada – o en la memoria regrabable 6, que define un programa informático. Este método incluye, en general, el intercambio de datos con el terminal de usuario por intermedio de la interfaz 8 y el procesamiento de datos dentro de la tarjeta de microcircuito 10 y, más concretamente, en el interior del microprocesador 2 con la posible utilización de datos almacenados en la memoria regrabable 6 y de datos almacenados temporalmente en la memoria viva 4.

Ejemplos de métodos que ponen en práctica la invención se proporcionan a continuación.

La Figura 2 representa la apariencia física general de la tarjeta de microcircuito 10 realizada con la forma general de un paralelepípedo rectangular de muy pequeño espesor.

La interfaz de comunicación 8 provista de los contactos, antes citados, aparece claramente sobre la cara de la tarjeta de microcircuito 10 visible en la Figura 2, bajo la forma de un rectángulo inscrito en la cara superior de la tarjeta de microcircuito 10.

Se describe ahora ejemplos de la realización de la invención con referencia a un algoritmo criptográfico de tipo AES *("Advanced Encription Standard'* según la terminología anglo-sajona) representado sintéticamente en la Figura 3.

Queda entendido, no obstante, que la invención puede, por ejemplo, aplicarse en el caso de otros algoritmos que implican una función no lineal, tal como el algoritmo DES, con por ejemplo la utilización de bloques de cifrado (también conocidos bajo la terminología *"block cipher").* La máscara o las máscaras aplicadas al dato en la entrada de la función no lineal no son, por otro lado, necesariamente la o las máscaras X aplicadas al inicio del algoritmo, sino que se deducen, en general fácilmente, según los mecanismos de algoritmo, por ejemplo tal como se describe en la solicitud de patente WO 2007/116140 antes citada.

La Figura 3 representa las etapas esenciales del método de cifrado AES de una palabra M dentro de la entidad electrónica.

La palabra M es, en general, una parte de un mensaje a cifrar que tiene, por ejemplo, una longitud de 128 bits. Otras longitudes son naturalmente susceptibles de consideración, tales como las longitudes de 192 bits y de 256 bits que se usan con frecuencia.

El ejemplo aquí descrito utiliza, como entidad electrónica, la tarjeta de microcircuito antes descrita con referencia a las Figuras 1 y 2, pero, por supuesto, se pueden utilizar otros tipos de entidad electrónica, tales como, por ejemplo, un ordenador personal.

A este respecto, la entidad electrónica memoriza, por ejemplo, dentro de la memoria no volátil 6, una clave criptográfica K a partir de la que se derivan sub-claves K_0, ...K_n por medio de un procedimiento de expansión de claves.

La obtención de las sub-claves K_0, ... K_n se puede realizar según técnicas conocidas y que, por lo tanto, no se describirán aquí con detalle. Se podrá referir, por ejemplo, a la solicitud de patente FAR 2 838 262.

Se hace constar, sin embargo, que la invención, descrita a continuación para el cálculo al nivel de las cajas-S (o *"S-box")* implicadas en cada iteración (o ROUND) del algoritmo AES, se podría poner en práctica en el momento de la aplicación de la función no lineal utilizada en el algoritmo de derivación de las sub-claves K_0, ... K_n.

El método de cifrado comienza en la etapa E100 por la recepción, por ejemplo a través de la interfaz 8 de la tarjeta de microcircuito, de la palabra (en general, una parte de mensaje) M a cifrar.

Se procede, entonces, dentro de la entidad electrónica al muestreo de un número aleatorio X utilizado como máscara de la palabra M en el curso de una etapa E102. Aunque se utilice el término de *"número aleatorio",* se trata, por ejemplo, en la práctica, de un número pseudo-aleatorio determinado en el interior del microprocesador 2. De manera general, el número X y cualquier dato, objeto de referencia más adelante como aleatorio, debe ser un número no previsible del exterior de la entidad electrónica.

Se procede, entonces, en el curso de una etapa E104, al enmascaramiento de la palabra M con el fin de obtener una palabra enmascarada M' por combinación de la palabra M y del número aleatorio X, por medio de una operación de O-EXCLUSIVA (que suele denominarse "*XOR*"): $M' = M \oplus X$.

Cuando una simple máscara se utiliza para enmascarar un dato (enmascaramiento de orden 1), la fuga de información sobre el dato enmascarado, quizás dirigido conjuntamente con la fuga de información sobre la máscara, con el fin de encontrar información sobre el dato sin cifrar. Este tipo de ataques, que suelen denominarse ataques de orden 2, se pueden evitar mediante la utilización de una segunda máscara (enmascaramiento de orden 2).

Así, como variante, puede ser dos máscaras X_1 y X_2 que se generan para doblemente enmascarar el mensaje M : M' = (M $\oplus$ X_1)$\oplus$ X_2. Esta configuración proporciona una protección del mensaje M al nivel del segundo orden de fuga. Se describirá, en particular, a continuación, un ejemplo de realización de la invención, que implica este doble enmascaramiento.

Se asignará, a continuación, el símbolo 'prima' a los valores enmascarados (aquí, por ejemplo M'), mientras que las magnitudes que no llevan el signo 'prima' representan las magnitudes correspondientes sin enmascaramiento, es decir tales como las que se hubieran obtenido en el curso del algoritmo AES realizado sin enmascaramiento; estas magnitudes sin enmascaramiento se introducen aquí con fines explicativos, pero no serán usadas por el método aquí descrito, que utiliza su versión enmascarada, salvo, por supuesto, para las magnitudes M y M_n utilizadas, respectivamente, en la entrada y en la salida.

Se presentará, a continuación, este método en diferentes etapas del algoritmo AES utilizando el dato enmascarado M' según las etapas aquí descritas que corresponden a las etapas clásicas del algoritmo AES adaptadas para tener en cuenta el enmascaramiento.

Se procede así, ante todo, a la transformación inicial por medio de la sub-clave K_0 en el curso de una etapa E106 aplicando la clave K_0 al dato por medio de un operador O EXCLUSIVA: $M'_0 = M' \oplus K_0$.

Si se denomina M_0 el resultado de la transformación inicial sin enmascaramiento ($M_0 = M \oplus K_0$), se puede observar que el resultado M'_0 de la etapa E106 se puede escribir $M'_0 = M_0 \oplus X$. Se observa así que el resultado de la etapa E106 corresponde al resultado de la transformación inicial sin enmascaramiento, enmascarado por el valor de la máscara X.

Se procede, después de la transformación inicial, a una etapa E108 de inicialización a 1 de un índice *i* que se refiere, en la serie, a la iteración (o ROUND) interesada.

En la etapa E110, se aplica a la palabra enmascarada M'_{i-1}, obtenida en la etapa precedente (etapa E106 después de la transformación inicial o iteración precedente), una iteración (ROUND) de las etapas E110 y E112, con el fin de obtener una nueva palabra enmascarada M'_1.

Cada iteración se puede diseñar para que el resultado obtenido después de la iteración M'_i, sea igual al resultado M_i después de la iteración *i* en un algoritmo sin enmascaramiento, enmascarado con una máscara X idéntica a la introducida en la etapa E104, es decir, al final de cada iteración, $M'_i = M_i \oplus X$.

Como alternativa, se puede prever que el enmascaramiento se modifique en cada iteración, por ejemplo volviéndole a enmascarar con una nueva máscara Z, según se describe a continuación.

Una vez realizada la iteración o ROUND i, se incrementa el valor del índice *i* en el curso de una etapa E114 y luego, se prueba, en el curso de una etapa E116, la igualdad i = n, en donde n es el número de iteraciones más uno utilizadas en el algoritmo interesado (en general, 10 iteraciones para una palabra de 128 bits).

Si no se ha alcanzado la última iteración (es decir, que no se ha verificado la igualdad i = n), se retorna a la etapa E110 para la puesta en práctica de la iteración siguiente.

Si, por el contrario, se alcanza la última iteración (es decir, cuando se verifica i = n), se procede a la transformación final en el momento de una etapa E118, en cuyo curso se obtiene, por lo tanto, una palabra M'_n a partir de la palabra M'_{n-1} precedentemente obtenida y ello con utilización de la sub-clave K_n (etapa E118).

El resultado de la transformación final M'_n corresponde, así, a la palabra cifrada obtenida gracias al algoritmo AES a partir de la palabra inicial M, enmascarada con la máscara precedentemente definida X (o la última máscara intermedia Z generada si se desea modificar el enmascaramiento en el curso de las iteraciones).

Se procede, entonces, en la etapa E120, al desenmascaramiento de la palabra obtenida en la etapa E118 con el fin de obtener la palabra cifrada M_n : $M_n = M'_n \oplus X$ (o si fuera el caso, con la máscara Z).

La palabra cifrada M_n puede, entonces, emitirse al exterior de la entidad electrónica por medio de la interfaz 8 en el curso de una etapa E122 que termina el método de cifrado por la entidad electrónica de la palabra M.

La presente invención se refiere, más concretamente, a la utilización de las cajas S-box, en el momento de la

iteración de las etapas E110 y E112.

En el dominio de la criptografía, la transformación $M'_{i-1} \rightarrow M'_i$ se puede modelizar por la composición de tres operaciones: una función aditiva de la clave derivada K_i, una función no lineal y una función lineal.

La etapa E110 ilustra la aplicación de la función no lineal mediante la utilización de una caja-S y descrita, con más detalle, haciendo referencia a los modos de realización siguientes.

La etapa E112 corresponde, entonces, a la aplicación de las otras dos funciones, de las que un ejemplo se da a conocer en la solicitud de patente WO 2007/116140 antes citada, en relación con su Figura 4, en particular etapas de desplazamiento de bits de sub-bloques (también denominada etapa de *ShiftRow),* de multiplicación por una matriz (etapa denominada *Mix Column) y* de adición de la clave K_i (etapa *Add Round Key).*

Para la realización de la etapa E110, la tabla-S se guarda en la memoria no volátil 6, tabla también denominada tabla de conversión *(look-up table o* LUT). La tabla de conversión S recibe datos de entrada de dimensión (es decir, de número de bits) *m* y proporciona datos de salida de dimensión *n,* en particular *m* puede ser igual a *n.* La invención se aplica igualmente en el caso en donde estas dos dimensiones son diferentes.

Un primer modo de realización de la etapa E110 se describirá, a continuación, haciendo referencia a la Figura 4.

En esta figura, la etapa E110 se pone en práctica bajo la forma de un sub-programa que recibe, en la entrada, el dato enmascarado M'_{i-1} y la máscara X utilizada para el enmascaramiento (etapa E200).

Se inicializa, entonces, por ejemplo al valor cero, una variable A que corresponde, como se verá a continuación, a un valor posible para la máscara X (etapa E202).

Se entiende por valor posible uno de los valores que es susceptible de tomar la máscara X, de la que se ha constatado que constituía un valor aleatorio.

Se lee, entonces, en la etapa E205, el valor memorizado en la tabla de conversión S en la posición definida por el dato enmascarado M'_{i-1} al que se ha aplicado previamente (con miras a su desenmascaramiento ocasional) la variable A por medio de la operación $M'_{i-1} \oplus A$. Se lee, por lo tanto, en la etapa E205 el valor $S\ M'_{i-1} \oplus A$).

Se hace constar que, en el ejemplo aquí descrito, el algoritmo utilizado para el enmascaramiento, como el utilizado para el desenmascaramiento, consisten en aplicar una operación de *"O- exclusiva'* (o suma booleanas), que se suele emplear, con este objeto, debido a las propiedades involutivas de esta función.

Se procede, siempre en la etapa E205, a la adición de una nueva máscara Z al valor que acaba de leerse y luego se memoriza el valor así obtenido en una tabla T en una posición definida por la suma booleana de la máscara X y de la variable A. La tabla T es también memorizada en una memoria de la entidad electrónica, aquí en general, una memoria viva, a la que el acceso es simple y rápido.

La nueva máscara Z utilizada es, por ejemplo, igualmente recibida, a la entrada, en el momento de la etapa E200. Como variante, esta nueva máscara Z se podría determinar, por ejemplo, mediante muestreo aleatorio, en la etapa de inicialización antes citada E202, en cuyo caso se retorna en la etapa E212 descrita más adelante. Según otra variante ya citada, se podría, de nuevo, utilizar la máscara X en esta etapa.

Cuando se realiza la copia de la salida de la tabla de conversión S, antes definida, en la tabla T en la posición también antes indicada, se incrementa el valor de la variable A (etapa E208).

Se comprueba, entonces, en la etapa E210, si la variable A ha barrido el conjunto de los valores posibles para la máscara X (comparando aquí el valor de A con 2^m puesto que este valor ha sido inicializado a cero durante la etapa E202).

En caso de comparación negativa (es decir, si algunos valores posibles para la máscara X no han sido procesados) se retorna a la etapa E205 para una nueva iteración de esta etapa con el nuevo valor de la variable A.

Por el contrario, si la comparación es positiva, el valor A ha tomado, sucesivamente, todos los valores posibles para la máscara X y se puede, entonces, retornar, en tanto como resultado de la etapa E110, el valor T[0] (etapa E212).

En efecto, la tabla T fue rellenada por las iteraciones sucesivas de la etapa E205, de tal modo que, cuando la variable A haya tomado el valor efectivo de la máscara X, se ha leído en la tabla de conversión S el valor correspondiente a la entrada $M'_{i-1} \oplus X = M'_{i-1}$, o sea el valor $S(M_{i-1})$ que se buscaba obtener, y se ha colocado este valor (después del enmascaramiento por la nueva máscara Z) en la tabla T, en la posición $X \oplus X = 0$.

De este modo, se ha reorganizado la tabla de correspondencia S en la tabla T, de tal forma que el valor de interés (S

(M_{i-1})) se memorice en una posición determinada (aquí la posición correspondiente al valor nulo). Sin embargo, esta reorganización ha incidido en un conjunto de valores (aquí el conjunto de la tabla de conversión); la reorganización efectuada depende, además, de la máscara aleatoria X y será, por lo tanto, diferente en cada ocasión.

El conjunto de estos efectos concurren a mejorar la seguridad de la puesta en práctica de la etapa E110.

Se hace constar que se utiliza aquí el vocablo *"tabla"* T aun cuando la estructura del dato interesado sólo tiene una entrada, como la tabla de conversión S.

Se ha representado en la Figura 5 un segundo modo de realización, susceptible de consideración, para la etapa E110.

Este modo de realización se asemeja al primer modo descrito con referencia a la Figura 4; sin embargo, la reorganización de la tabla de correspondencia S en una tabla T es tal que el valor de interés en la tabla de correspondencia S ($S(M_{i-1})$) no es objeto de recopia (después del enmascaramiento) a una posición predeterminada (la posición que corresponde al valor nulo en la Figura 4), sino a una posición determinada, de manera aleatoria, en cada paso a la etapa E110, como se indica a continuación, gracias a la utilización de un nuevo valor aleatorio Y, lo que mejora también la seguridad.

El método de la Figura 5 comienza por la recepción, a la entrada, del dato enmascarado M'_{i-1} y de la máscara X (etapa E300) como para el primer modo de realización descrito con referencia a la Figura 4.

Se procede, entonces, a la inicialización de la variable A, por ejemplo al valor nulo, pero también, en este caso, a la de una variable Y por muestreo aleatorio en un número de bits m igual al de la máscara X (etapa E302).

Se calcula, entonces, en la etapa E304, para una simplificación de los cálculos, un valor intermedio Y' igual al enmascaramiento del valor aleatorio Y que se acaba de determinar por la máscara X recibida a la entrada: $Y' = Y \oplus X$.

Como para el primer modo de realización, se procede, entonces, a un bucle, que permite dar a la variable A todos los valores, susceptibles de consideración, para la máscara X.

Se inicia este bucle en la etapa E306, en donde se lee, ante todo, en la tabla de conversión S el valor de salida correspondiente, en la entrada, al dato enmascarado M'_{i-1} transformado por aplicación de la variable A por medio de la operación de desenmascaramiento. Se lee, así, en la etapa E306 el valor $S(M'_{i-1} \oplus A)$, valor que se enmascara con la ayuda de una nueva máscara Z (que se puede determinar según las diferentes posibilidades ya consideradas para el primer modo de realización).

El valor leído y enmascarado se recopia, a continuación, en una tabla T en una posición definida por aplicación al valor intermedio Y' de la variable A por medio, asimismo, de la operación de desenmascaramiento, es decir, en la posición $A \oplus Y'$.

Se puede, entonces, pasar a la etapa E308 en donde se incrementa el valor de A con el fin de comprobar, en la etapa E310, si se ha alcanzado el valor máximo 2^m: en caso negativo, se realiza un bucle en la etapa E306; en caso afirmativo, se termina el procesamiento del conjunto de los valores posibles para el valor A y se puede por lo tanto, reenviar, a la salida, en la etapa E312 el valor objeto de recopia en la tabla en la posición Y.

En efecto, cuando la copia de la etapa E306 ha considerado la posición Y, ello significa que la variable A ha permitido efectivamente desenmascarar el enmascaramiento de Y realizado por medio de la máscara X (etapa E304) y por lo tanto, este mismo valor A ha permitido desenmascarar el dato enmascarado M'_{i-1} en el dato M_{i-1} del que se desea leer la salida en la tabla de conversión S.

Se ha representado en la Figura 6 un tercer modo de realización, susceptible de consideración, para la etapa E110 de la Figura 3.

Este modo de realización corresponde al caso en el que el dato M_{i-1} es enmascarado al segundo orden, es decir, mediante la utilización de dos máscaras X_1 y X_2 y en donde el valor enmascarado M'_{i-1} vale, por lo tanto, $M_{i-1} \oplus X_1 \oplus X_2$.

En este caso, se recibe, en la entrada, en la etapa E400, el dato enmascarado M'_{i-1} así como las dos máscaras X_1 y X_2.

Según este modo de realización, se procede, ante todo, en la etapa E402 al muestreo aleatorio de un valor X_3, valor que consiste en un número del mismo número de bits m que las máscaras X_1 y X_2 (de aquí la notación rand(m)).

Se procede, entonces, a una etapa de inicialización E404 en donde se pone al valor nulo una variable A. Se puede,

además, calcular, en esta etapa E404, con miras a una simplificación de los cálculos, un valor X' igual al valor aleatorio precedentemente determinado X_3 enmascarado sucesivamente por las máscaras X_1 y X_2, o sea $X' = (X_3 \oplus X_1) \oplus X_2$.

En esta etapa, se procede de realizar las operaciones de O-exclusiva (XOR) en el orden de los paréntesis, a saber, primero $X_3 \oplus X_1$ y luego, la adición del resultado con X_2. En efecto, respetando este orden, se garantiza una protección contra los ataques de orden 2, puesto que no se manipula directamente el valor $X_1 \oplus X_2$ que podría entonces asemejarse a un simple enmascaramiento (al primer orden) del dato X_3.

Una vez realizada esta etapa de inicialización, se procede a la etapa E406, en donde se determina un valor intermedio A' por aplicación de la variable A al valor X' con miras a su desenmascaramiento, es decir, aquí por la operación $A \oplus X'$ (puesto que, como se constató, a propósito del primer modo de realización, la operación de enmascaramiento y de desenmascaramiento se realiza, en este caso, por la única y misma operación de suma booleana).

Se procede, entonces, a la etapa E408 en donde se comienza por leer, en la tabla de conversión S, el valor de salida asociado al valor de entrada $M'_{i-1} \oplus A$, es decir, el valor de salida correspondiente al dato enmascarado M'_{i-1} transformado por aplicación de la variable A por medio de la operación de desenmascaramiento.

El valor de salida así obtenido S $M'_{i-1} \oplus A$) es, entonces, sucesivamente enmascarado por dos máscaras Z_1 y Z_2 con el fin de obtener un enmascaramiento del segundo orden.

Como para el primer modo de realización, las nuevas máscaras Z_1 y Z_2 se pueden pasar al argumento (es decir, en la entrada) en la etapa E400 o, como variante, determinarse en la etapa de inicialización E404 (en cuyo caso, se retornan igualmente a la etapa E414 descrita a continuación).

Con el fin de evitar fugas al segundo orden susceptibles de detectarse, se procede, como se indica sucesivamente al enmascaramiento con la ayuda del valor Z_1 y luego, en una segunda fase, al enmascaramiento con la ayuda del valor Z_2.

El valor obtenido, en la salida, después del enmascaramiento es, por último, escrito en el curso de esta misma etapa E408 en una tabla T en una posición definida por el valor intermedio A' (del que se recuerda que resulta de la aplicación al valor aleatorio determinado X_3 sucesivamente de un enmascaramiento por las máscaras X_1, X_2, después de la aplicación de la operación de desenmascaramiento con la variable A en la etapa E406).

Se procede, a continuación, a la etapa E410 en donde se incrementa la variable A y luego, se comprueba, en la etapa E412, si el valor A ha alcanzado el valor máximo posible para las máscaras (aquí 2^m).

En caso de respuesta negativa, se retorna a la etapa E406 con el fin de reiterar las etapas E406 y E408 con todos los valores, susceptibles de consideración, para A.

Por último, en caso de respuesta positiva, se reenvía a la etapa E414 el valor $T[X_3]$ correspondiente al dato memorizado en la tabla T en la posición definida por el valor aleatorio X_3 muestreado en la etapa E402.

En efecto, la tabla T fue rellenada en su posición X_3 en la iteración de la etapa E408 correspondiente a un valor A que ha permitido el desenmascaramiento efectivo del enmascaramiento de X_3 por las máscaras X_1 y X_2 en la etapa E404: es, por lo tanto, la iteración para la que el valor A ha permitido, asimismo, desenmascarar el valor M'_{i-1} en el valor M_{i-1} de la que se buscaba justamente conocer la salida en la tabla de conversión S.

Este modo de realización permite así obtener las ventajas antes citadas para los modos de realización descritos en las Figuras 4 y 5 dentro del marco de un enmascaramiento del segundo orden.

Cualquiera que sea el modo de realización anteriormente considerado, la arquitectura material de los equipos electrónicos, tales como el microcircuito 10, impone, a veces, la utilización de palabras-máquina en un número de bits determinados, por ejemplo 8 bits, 16 bits o 32 bits. Así, cuando los datos de salida sean palabras de un número inferior de bits, por ejemplo 4 bits, se memoriza varias palabras de salida S(M) en una sola palabra máquina de la tabla de conversión.

Las Figuras 7 y 8 ilustran esta realización práctica. En la Figura 7, se observa la tabla S anteriormente utilizada, que presenta 2^m palabras de salida de longitud *n*. En la Figura 8, la realización práctica memoriza las palabras de salida de longitud n, aquí 2 bits, en palabras máquina de longitud $2^w n$, en este caso 16 bits (w es entonces igual a 3). El almacenamiento del conjunto de las palabras de salida S(M) sólo requiere, entonces, la utilización de 2^{m-w} palabras máquina, que almacenan cada una, respectivamente, 2^w palabras de salida S(M) (asimismo denominadas por la serie de "sub-palabras"). Se obtiene, así, un almacenamiento en memoria eficaz. La tabla así formada comprende 2^{m-w} palabras de longitud $2^w n$.

Las palabras de salida se memorizan de manera que se acceda a la palabra S(M) seleccionando una palabra-máquina $S_{máquina}(M_H)$ a partir de los m-w bits más significativos de M (indicados para la serie M_H) y recuperando la sub-palabra de la palabra seleccionada a partir de los w bits menos significativos de M (en adelante, denominados M_L). En particular $S_{máquina}(M_H) = \{S(M_H, 0), S(M_H,1),..., S(MH,2^w-1)\}$.

Otra convención, que descompone la palabra M, puede igualmente considerarse consistiendo, por ejemplo, en seleccionar la palabra máquina a partir de los bits centrales de M y seleccionar la sub-palabra a partir de bits extremos de M (por ejemplo, los dos bits de peso fuerte y el bit de peso débil).

Considerando lo que precede, se puede leer la palabra máquina $S_{máquina}(M_H)$, de manera asegurada, aplicando los algoritmos, antes citado en relación con las Figuras 4 a 6, a las partes de peso fuerte de M' y de su máscara X.

Se recupera así $S_{máquina}(M_H)$ ocasionalmente enmascarado, en función de $M'_{i-1,H}$, A_H o X_H (ocasionalmente $X_{1,H}$ y $X_{2,H}$ en caso del doble enmascaramiento) (representando el índice H los m-w bits más significativos).

Sin enmascaramiento de salida, se recupera $S_{máquina}(M_H)$.

Con un enmascaramiento de salida simple con la ayuda de la máscara Z, se recupera $S_{máquina}(M_H) \oplus Z$ (Z de longitud m-w).

Con un enmascaramiento de salida doble con la ayuda de las máscaras Z_1 y Z_2, se recupera $S_{máquina}(M_H) \oplus Z_1 \oplus Z_2$ (Z_1 y Z_2 de longitud m-w).

Asimismo, órdenes de enmascaramiento superiores pueden considerarse igualmente sin complicar los mecanismos descritos a continuación.

Por otro lado, se dispone de $M'_{i-1,L}$, X_L (posiblemente $X_{1,L}$ y $X_{2,L}$ en caso del doble enmascaramiento) que corresponden a los bits de identificación de la palabra de salida en el interior de la palabra máquina $S_{máquina}(M_H)$, posiblemente enmascarada, leída en la tabla T según los algoritmos anteriores.

Se describe ahora, haciendo referencia a las Figuras 9 a 11, diferentes mecanismos que permiten extraer eficazmente, y ocasionalmente con toda seguridad, habida cuenta de los grados de enmascaramiento, la sub-palabra (por lo tanto la palabra de salida) de índice $M_{i-1,L}$ de la palabra máquina $S_{máquina}(M_H)$ posiblemente enmascarada en salida, a partir del índice enmascarado $M'_{i-1,L}$ y de las máscaras A_L o X_L y posiblemente $X_{1,L}$ y $X_{2,L}$.

Para simplificar la notación, se indica, además, como U (o U' si está enmascarada) la palabra-máquina, j' el índice enmascarado, r_j (r_j1 y r_j2) las máscaras del índice, r_u (r_u1 y r_u2) las máscaras de salidas que enmascaran, ocasionalmente, la palabra-máquina U'. Se busca, además, extraer la sub-palabra U(j) a partir de j' y de las máscaras de j' (o extraer U'(j) y las máscaras de salidas correspondientes indicadas s_u (s_u1 y s_u2)), es decir, sin manipular j por razones de seguridad.

Estos diferentes mecanismos se refieren, en general, al acceso a una sub-palabra de índice j en un palabra binaria (U, U') formada por 2^w sub-palabras $\{U(0),..., U(2^w-1)\}$ a partir del índice binario j' enmascarado por un máscara binaria correspondiente r_j, que comprende:

i) una etapa de separación de dicha palabra (U, U') en dos mitades ($H_0(U)$, $H_1(U)$, U_H, U_L), respectivamente, de los bits más significativos y de los bits menos significativos,

ii) al menos una etapa de selección de una mitad de dicha palabra (U, U') en función de los valores de los bits del mismo índice en, respectivamente, el índice j' y la máscara r_j.

En efecto, se observa que según el valor del bit de peso fuerte de j, la sub-palabra a la que se quiere acceder está en la parte izquierda (es decir, es de peso fuerte) o bien, en la parte derecha (es decir, de peso débil) de U. Así, combinando la utilización de los bits respectivos del índice enmascarado y de su máscara (es decir, de los bits que contribuyen a obtener el bit correspondiente del índice j no enmascarado), se selecciona eficazmente la parte de la palabra de interés, sin manipular el índice j no enmascarado.

En particular, se elegirá sub-palabras de la misma longitud.

Se ilustra una primera realización de estos mecanismos con la ayuda de la palabra U=1010011101101101 compuesta de 8 sub-palabras de índice que varía de 000 (sub-palabra de peso fuerte – a la izquierda) a 111 (sub-palabra de peso débil – a la derecha), haciendo referencia a la Figura 9. Más concretamente, se desea acceder a la sub-palabra cuyo índice enmascarado j' vale 101 y su máscara r_j 111 (se recuerda que $j'=j \oplus r_j$). Se observa que $j=101 \oplus 111=010$, por lo tanto se desea acceder a la sub-palabra 01 compuesta de los 5º y 6º bits de U, partiendo desde la izquierda.

Esta realización pone en práctica las etapas algorítmicas siguientes:

para k=0 a w-1
$(R_0, R_1) \leftarrow U$
$swap_{j'[k]}(R_0, R_1)$
$swap_{rj[k]}(R_0, R_1)$
$U \leftarrow R_0$
retorno R_0

en donde R_0 y R_1 son dos registros de longitud al menos igual a la semi-longitud de U, en este caso al menos 8 bits, y $swap_b(R_0, R_1)$ es una función que modifica por inversión el contenido de los dos registros, cuando b=0:

$$swap_b(R_0, R_1) = \begin{cases} NOP & si \quad b = 0 \\ swap(R_0, R_1) & si \quad b = 1 \end{cases}.$$

Una tal función $swap_b$ se puede poner en práctica por las etapas siguientes:

$tmp \leftarrow R_1$
$R_b \leftarrow R_0$
$R_{\overline{b}} \leftarrow tmp$

Como alternativa, $swap_b$ puede ponerse en práctica con la ayuda de la función Rotate(R,x) que efectúa una rotación de R en x bits a la derecha o a la izquierda, mediante $swap_b(R_0,R_1)$=Rotate((R_0,R_1), I+b.I), en donde *I* es la longitud de los registros. En este caso, se utilizará un solo registro R compuesto de las dos partes iguales R_0 y R_1, con el fin de que el desfase de bits desplace potencialmente los bits desde una parte del registro R a la otra.

Como alternativa a la utilización de dos registros, es posible utilizar un registro doble (longitud 2I, en este caso, al menos 16 bits) constituido por dos partes.

En la etapa E600, se inicializa un contador *k* a 0.

En la etapa E602, se asigna a los dos registros R_0 y R_1, respectivamente U_H (10100111) y U_L (01101101).

En la etapa E604, se aplica la función $swap_b$ en función del bit más significativo del índice j'={j'[0], j'[1], ..., j'[w-1]}:

j'[k=0]=1.

En este caso, se invierte entonces los dos registros: R_0 = 01101101 y R_1 = 10100111.

En la etapa E606, se aplica, de nuevo, la función $swap_b$ en función del bit del mismo índice k, por lo tanto el más significativo, de la máscara r_j: r_j[k=0]=1.

En este caso, se invierte entonces los dos registros: R_0 = 10100111 y R_1 = 01101101.

En la etapa E608, se sustituye el valor de U por el contenido del registro R_0.

En la etapa E610, se incrementa el valor de iteración k: k=1.

En la etapa E612, se compara k con w (aquí w=3). Como k<3, se retorna a la etapa E602 distribuyendo el nuevo valor de U en los dos registros: R_0 = 1010 y R_1 = 0111.

Como j'[k=1]=0 y r_j[k=1]=1, se efectúa una sola permutación de los dos registros, en el momento de la etapa E606: R_0 = 0111 y R_1 = 1010.

Se conserva así en la etapa E608, U=0111.

En la etapa E612, k=2; por lo tanto, se retorna a la etapa E602.

La iteración k=2 lleva a R_0 = 01 y R_1 = 11 puesto que dos permutaciones se han efectuado en las etapas E604 y E606.

En la etapa E612, k=3=w, por lo tanto se pasa a la etapa E614 mediante la cuál se retorna a U=01. Se obtiene, así, la sub-palabra prevista.

Se observa que, en cada iteración, la magnitud de U disminuye a la mitad y se converge así por dicotomía hacia la sub-palabra deseada.

La seguridad de este mecanismo está garantizada por la aplicación de las funciones de permutación (inversión de los dos registros) a todas las iteraciones (con parámetros diferentes) aun cuando no se desarrolle ninguna permutación.

De una forma general, la operación de permutación *(swap),* tal como se consideró anteriormente confiere, cuando se utiliza, de forma condicional, un grado de seguridad elevado cuando se trata de identificar (por ejemplo, para selección o aislamiento) un elemento entre dos. A este respecto, podrá considerarse la posibilidad de una protección de este método independiente de las demás enseñanzas descritas en la presente solicitud de patente.

Se puede resaltar que al utilizar una indexación de j' (y de sus máscaras) ya no de izquierda a derecha sino de derecha a izquierda, ha lugar a proceder a una permutación (swap) suplementaria en el interior de cada una de las iteraciones k.

Con el fin de proporcionar una protección de orden 1 de enmascaramiento de los datos, el modo de realización antes descrito puede extenderse al dato U' enmascarado con la ayuda de una máscara aleatoria r_u. En este caso, se busca determinar no solamente la sub-palabra de U' (como se describió anteriormente) que conviene sino también la sub-palabra de r_u correspondiente.

El acceso a la sub-palabra $r_u(j)$ a partir de la palabra máquina r_u es similar al mecanismo anterior.

En cada una de las iteraciones k, se realiza, inmediatamente después de la etapa E608, las etapas E602', E604', E606' y E608' aplicadas a la máscara r_u, similares a las de número correspondiente E602, E604, E606 y E608.

El algoritmo antes propuesto, se hace entonces:

para k=0 a w-1
$(R_0, R_1) \leftarrow U'$
$swap_{j'[k]}(R_0, R_1)$
$swap_{r_j[k]}(R_0, R_1)$
$U' \leftarrow R_0$
$(R_0, R_1) \leftarrow r_u$
$swap_{j'[k]}(R_0, R_1)$
$swap_{r_j[k]}(R_0, R_1)$
$r_u \leftarrow R_0$
retorno (U', r_u)

Se obtiene al final del algoritmo (etapa E614) los valores U' y r_u correspondientes a las sub-palabras buscadas, tales que $U' = U \oplus r_u$, en donde U es el valor de salida (de la caja S) no enmascarada.

Se describe, ahora, con referencia a la Figura 10, una realización más asegurada que pone en práctica un doble enmascaramiento de los valores, tanto para el índice j' (máscaras r_j1 y r_j2) como para la palabra U' (máscaras r_u1 y r_u2), por ejemplo resultantes de la etapa E414 anterior aplicada a la determinación de $S_{máquina}(M_H)$. Estos dobles enmascaramientos se pueden aplicar, por supuesto, independientemente.

Esta realización difiere, de la representada en la Figura 9, en sustancia por la aplicación de la función swap tres veces en función de bits correspondientes en el índice j' y en cada una de las dos máscaras r_j1 y r_j2 (se entiende, por lo tanto, que la invención puede extenderse, sin dificultad, a órdenes superiores de enmascaramiento efectuando un

número de permutaciones igual a 1 + número de máscaras de j) y mediante la determinación de las sub-palabras en la palabra U' y en cada una de las máscaras r_u1 y r_u2 de U. Para una homogeneidad de seguridad, conviene elegir un mismo grado de enmascaramiento del índice j y de la palabra binaria U.

Las etapas E700 a E704 no difieren de las etapas E600 a E604.

Puesto que se tiene dos máscaras de índice r_j1 y r_j2, se efectúa dos permutaciones condicionales a las etapas E706 y E708, indexadas respectivamente en los dos bits correspondientes de las dos máscaras $r_j1[k]$ y $r_j2[k]$.

La etapa E710 es la misma que la etapa E608 anterior.

En la misma iteración, se calcula, de forma similar, las dos partes de máscaras r_u1 y r_u2 que corresponden a las etapas E702' a E710' y E702" a E710".

En la etapa E712, se incrementa k de modo que se forme w iteraciones (en comparación con la etapa E714).

En la etapa E716, se retorna U', r_u1 y r_u2 que representan las sub-palabras deseadas (por lo tanto, valores de salida de la tabla S) de las palabras iniciales. Estos tres valores verifican: $U' = U \oplus r_u1 \oplus r_u2$.

Con el fin de aumentar todavía más la seguridad, en particular evitando utilizar las variables intermedias U' y U'(j), que constituyen un debilitamiento potencial debido a la dependencia con el índice j, se propone el mecanismo siguiente haciendo referencia a la Figura 11.

Aunque descrito con la ayuda de un doble enmascaramiento, el mecanismo se aplica a un simple enmascaramiento, incluso en la ausencia de enmascaramiento de la palabra U.

Se define, ante todo, las dos funciones $H_0(y)$ y $H_1(y)$ que retornan, respectivamente, la mitad de los bits más significativos de *y* así como la mitad de los bits menos significativos de y. Estas dos funciones son fácilmente realizables con la ayuda de la función $swap_b$ antes citada, por ejemplo como sigue:

$$H_b(y): \quad (R_0, R_1) \leftarrow y$$
$$\text{swap}_b(R_0, R_1)$$
$$\text{retorno } R_0$$

El mecanismo de acceso a la sub-palabra de índice *j* se puede poner en práctica con la ayuda de las instrucciones siguientes:

para k=0 a w-1

$$(U', r_u1, r_u2) \leftarrow \text{Select}\,(2^{w-1-k}n, (U', r_u1, r_u2), (j'[w-k], r_j1[w-k], r_j2[w-k]))$$

retorno (U', r_u1, r_u2)

en donde la función Select recibe en parámetro una dimensión long ($2^{w-1-k}n$ que representa la magnitud de una mitad de palabra U', r_u1 y r_u2 que se desea recuperar al final de la iteración considerada), un primer 3-upleto de una palabra enmascarada y dos máscaras asociadas y un segundo 3-upleto de un bit enmascarado (aquí el bit de índice w-k del índice j) y dos máscaras asociadas y retorna un 3-upleto (U', r_u1, r_u2) que verifica $U' \oplus r_u1 \oplus r_u2 = H_{j[w-k]}(U)$.

Se constata así que iterando esta función para cada uno de los bits que componen el índice j, se aísla sucesivamente las mitades de la palabra correspondiente a los diferentes bits del índice j para llegar a la sub-palabra de U' de índice j acompañada de las sub-palabras de las máscaras correspondientes.

Se constata, asimismo, que la aplicación de este mecanismo a un enmascaramiento simple hace intervenir 2-upletos y no más de 3-upletos (extensible también a q máscaras y q-upletos).

En la etapa E800, se inicializa un valor de iteración k a 0.

Las etapas E802 a E820 ilustran un ejemplo de realización de la función Select.

En la etapa E802, se genera dos máscaras aleatorias de longitud long=$2^{w-1-k}n$.

En la etapa E804, se genera un booleano *b* aleatorio.

En la etapa E806, se calcula un bit enmascarado a partir del booleano *b y* de los bits de índice *k* (partiendo del bit de mayor peso al bit de menor peso, a medida de las iteraciones) en, respectivamente, las dos máscaras del índice j': b'

$\leftarrow (r_j1[w\text{-}k] \oplus b) \oplus r_j2[w\text{-}k]$.

En esta etapa, se procede a realizar las operaciones XOR en el orden de los paréntesis, a saber, primero $r_j1\,[w\text{-}k] \oplus b$ y luego, la adición del resultado con el booleano $r_j2[w\text{-}k]$. En efecto, respetando este orden, se garantiza una protección contra los ataques de orden 2, puesto que no se manipula directamente el valor $r_j1[w\text{-}k] \oplus r_j2[w\text{-}k]$, que podría considerarse como una simple máscara sólo protegen el bit $j[w\text{-}k]$ por un enmascaramiento de orden 1.

Para las etapas siguientes E808 a E818 (que pueden invertirse entre sí al ser independientes), se utiliza tres pares de registros direccionables, en adelante indicados como A_0,A_1 (asignados a la palabra U), B_0,B_1 (asignados a la primera máscara de U), C_0 y C_1 (asignados a la segunda máscara de U).

Se asigna sucesivamente:

- al registro A indexado por b' calculado en la etapa E806, la parte $H_{j[w-k]}(U')$ enmascarada por un (t_1) de las máscaras aleatorias generadas en la etapa E802:

$$A_{b'} \leftarrow H_{j'[w-k]}(U') \oplus t_1$$

- al otro registro A, la otra parte de U' enmascarada por la misma máscara.

Se procede, del mismo modo, para las dos máscaras r_u1 y r_u2, que utilizan, respectivamente, la otra máscara t_2 y la combinación de las dos máscaras t_1 y t_2, y se almacenan por pares, respectivamente, en B_0, B_1, C_0 y C_1.

En esta fase, se podría demostrar que para el índice *'w-k'* anterior, cualesquiera que sean los valores de las dos máscaras $r_j1[w\text{-}k]$ y $r_j2[w\text{-}k]$, se tiene:

$$(A_b, B_b, C_b) = (H_{j[w-k]}(U') \oplus t_1,\ H_{j[w-k]}(r_u1) \oplus t_2,\ H_{j[w-k]}(r_u2) \oplus t_1 \oplus t_2)$$

$$(A_{\bar{b}}, B_{\bar{b}}, C_{\bar{b}}) = (H_{\overline{j[w-k]}}(U') \oplus t_1,\ H_{\overline{j[w-k]}}(r_u1) \oplus t_2,\ H_{\overline{j[w-k]}}(r_u2) \oplus t_1 \oplus t_2)$$

y los valores almacenados en los registros verifican:

$$A_b \oplus B_b \oplus C_b = H_{j[w-k]}(U).$$

Debido a la independencia de estos valores con respecto a los de r_j1 y r_j2, se obtiene una independencia con respecto al índice j y por lo tanto, una mayor seguridad.

Así, en la etapa E820, se sustituye con (A_b, B_b, C_b) a los valores precedentes (U', r_u1, r_u2).

A continuación, se incrementa *k*, en la etapa E822.

Se compara *k* y *w* para determinar si se pone fin a las iteraciones en la etapa E824 (similar a las etapas E612 y E714 anteriores, con retorno a la etapa E802 para una nueva iteración).

Al final de las iteraciones (etapa E826), se retorna (U', r_u1, r_u2) que corresponden respectivamente a la sub-palabra enmascarada buscada y a las dos sub-palabras de máscaras, que permiten desenmascarar la sub-palabra U' obtenida: $U' \oplus r_u1 \oplus r_u2 = U$.

La utilización de los parámetros aleatorios t_1,t_2 y b no es indispensable. Proporciona, no obstante, una garantía de seguridad del algoritmo porque permite, de una parte, proteger los valores manipulados por un doble enmascaramiento (t_1 y t_2, que se puede reducir a un simple enmascaramiento, si fuere necesario) y de otra parte, atribuir el resultado deseado arbitrariamente en uno u otro de los dos registros indexados en b. En la ausencia de b, se retorna a la etapa E826 A_0, B_0 y C_0. En la ausencia de t_1 y t_2, el parámetro *long* no es necesario en la función Select antes indicada.

En los mecanismos anteriores, con referencia a las Figuras 9 a 11, los valores j'[k], r_j[k], r_j1[k] y r_j2[k] desempeñan funciones simétricas, de tal modo que es admisible invertir sus posiciones en estos mecanismos, por ejemplo el valor b' en la etapa E806 puede utilizar j' en lugar de la máscara r_j1 y las funciones H de las etapas E808 y E814 son entonces indexadas por la máscara r_j1.

Gracias a estos mecanismos, se accede, de forma garantizada, a la sub-palabra de una palabra en particular enmascarada, compuesta de una pluralidad de sub-palabras, con la ayuda de un índice igualmente enmascarado.

Los ejemplos que preceden sólo son modos de realización de la invención que no tienen carácter limitativo alguno.

REIVINDICACIONES

1. Método de procesamiento criptográfico de datos representados bajo forma digital, poniéndose dicho método en práctica por una entidad electrónica (10) y que comprende una transformación (E110) de un dato de entrada (M'_{i-1}), enmascarado por una máscara de entrada (X; $X_1 \oplus X_2$), en un dato de salida (M'_i), poniendo en práctica dicha transformación una tabla de conversión (S), caracterizado por las etapas siguientes :

- para al menos una pluralidad de valores posibles (A) para la máscara de entrada, la transferencia del valor de salida de la tabla de conversión (S) que corresponde al dato de entrada enmascarado (M'_{i-1}) transformado por aplicación de una operación de desenmascaramiento por medio del valor posible (A), en una tabla (T) en una posición correspondiente a un valor determinado (0; Y; X_3) enmascarado por la máscara de entrada y transformado por aplicación de la operación de desenmascaramiento por medio del valor posible;

- determinación del dato de salida por medio del dato situado en la tabla (T) en la posición correspondiente al valor determinado (0; Y; X_3).

2. Dispositivo electrónico de procesamiento criptográfico de datos representados bajo forma digital, apto para realizar una transformación de un dato de entrada, enmascarado por una máscara de entrada, en un dato de salida por medio de una tabla de conversión, caracterizado por:

- medios para transferir, para al menos una pluralidad de valores posibles para la máscara de entrada, el valor de salida de la tabla de conversión correspondiente al dato de entrada enmascarado, transformado por aplicación de una operación de desenmascaramiento, por medio del valor posible, en una tabla en una posición correspondiente a un valor determinado enmascarado por la máscara de entrada y transformado por aplicación de la operación de desenmascaramiento por medio del valor posible;

- medios para determinar el dato de salida por medio del dato situado en la tabla en la posición correspondiente al valor determinado.

3. Dispositivo electrónico según la reivindicación 2, en donde la tabla de conversión define una función no lineal (F).

4. Dispositivo electrónico según la reivindicación 2 o 3, en donde la máscara de entrada es una máscara del primer orden.

5. Dispositivo electrónico según la reivindicación 2 o 3, en donde la máscara de entrada es un máscara del segundo orden.

6. Dispositivo electrónico según la reivindicación 5, que comprende:

- medios de enmascaramiento del valor determinado por un primer elemento de máscara (X_1);

- medios de enmascaramiento del valor determinado por un segundo elemento de máscara (X_2).

7. Dispositivo electrónico según una de las reivindicaciones 2 a 6, en donde los medios para transferir comprenden medios de enmascaramiento del valor transferido por una máscara de salida (Z; Z_1, Z_2).

8. Dispositivo electrónico según la reivindicación 7, en donde la máscara de salida es una máscara del segundo orden.

9. Dispositivo electrónico según la reivindicación 8, en donde los medios de enmascaramiento comprenden:

- primeros sub-medios de enmascaramiento por un primer valor aleatorio (Z_1);

- segundos sub-medios de enmascaramiento por un segundo valor aleatorio (Z_2).

10. Dispositivo electrónico según una de las reivindicaciones 2 a 9, en donde los medios para transferir actúan para el conjunto de los valores posibles para la máscara de entrada.

11. Dispositivo electrónico según una de las reivindicaciones 2 a 10, en donde el valor determinado es un valor predeterminado.

12. Dispositivo electrónico según una de las reivindicaciones 2 a 10, que comprende medios para obtener el valor determinado por muestreo aleatorio.

13. Dispositivo electrónico según una de las reivindicaciones 2 a 12, en donde la o las máscaras (X; X_1, X_2; Z; Z_1, Z_2) son máscaras aditivas.

14. Dispositivo electrónico según la reivindicación precedente, en donde la o las máscaras son máscaras booleanas.

15. Dispositivo según una de las reivindicaciones 2 a 14, en donde dicho dato situado en la tabla es una palabra que comprende una pluralidad de sub-palabras de salida y en donde los medios de determinación del dato de salida comprenden medios de acceso al dato de salida, entre las sub-palabras, por medio de una parte del dato de entrada (M'_L) y de una parte de la máscara de entrada.

16. Dispositivo según la reivindicación 15, en donde los medios de acceso al dato de salida en dicha palabra comprenden:

i) medios de separación de dicha palabra (U, U') en dos mitades (H_0 (U), H_1 (U), U_H, U_L), respectivamente, de los bits más significativos y de los bits menos significativos,

ii) medios de selección de una de dichas mitades de dicha palabra (U, U') en función de los valores de los bits de un mismo índice en, respectivamente, la parte del dato de entrada (M'_L, j') y la parte de la máscara de entrada (r_j, r_j1, r_j2).

17. Tarjeta de microcircuito que comprende un dispositivo electrónico según una de las reivindicaciones 2 a 16.

18. Producto de programa informático, que comprende una serie de instrucciones adaptadas, cuando se ejecutan por un microprocesador, para poner en práctica un método según la reivindicación 1.

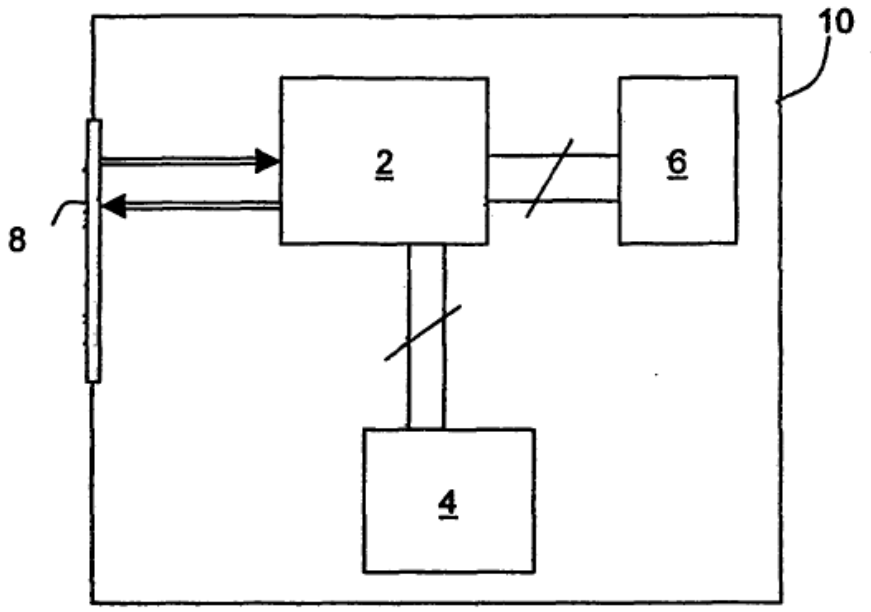

Figura 1

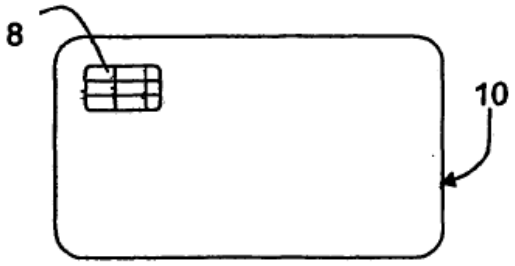

Figura 2

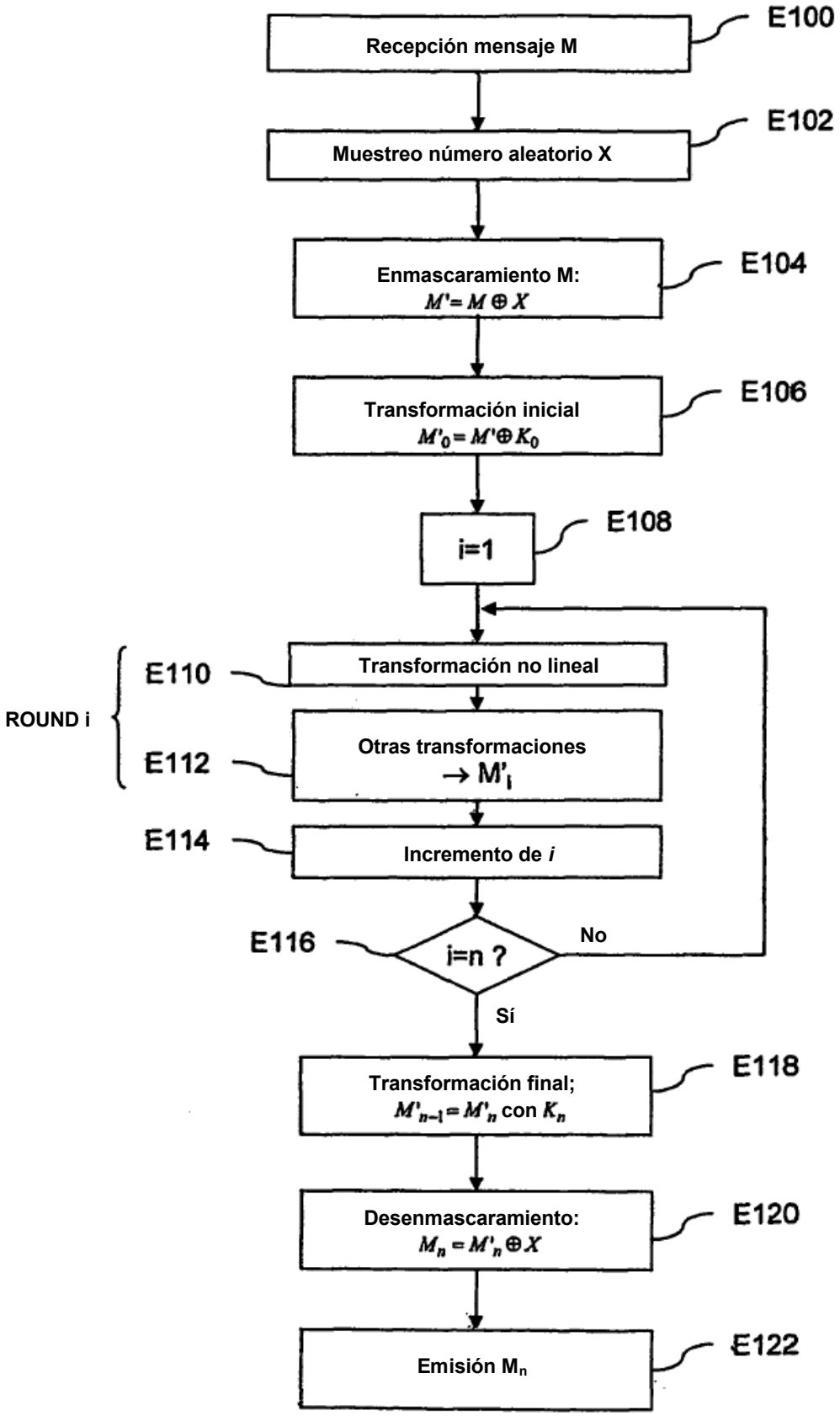
E100
Recepción mensaje M
E102
Muestreo número aleatorio X
E104
Enmascaramiento M:
$M' = M \oplus X$
E106
Transformación inicial
$M'_0 = M' \oplus K_0$
E108
i=1
ROUND i
E110
Transformación no lineal
E112
Otras transformaciones
$\rightarrow M'_i$
E114
Incremento de *i*
E116
i=n ?
No
Sí
E118
Transformación final;
$M'_{n-1} = M'_n$ con K_n
E120
Desenmascaramiento:
$M_n = M'_n \oplus X$
E122
Emisión M_n

Figura 3

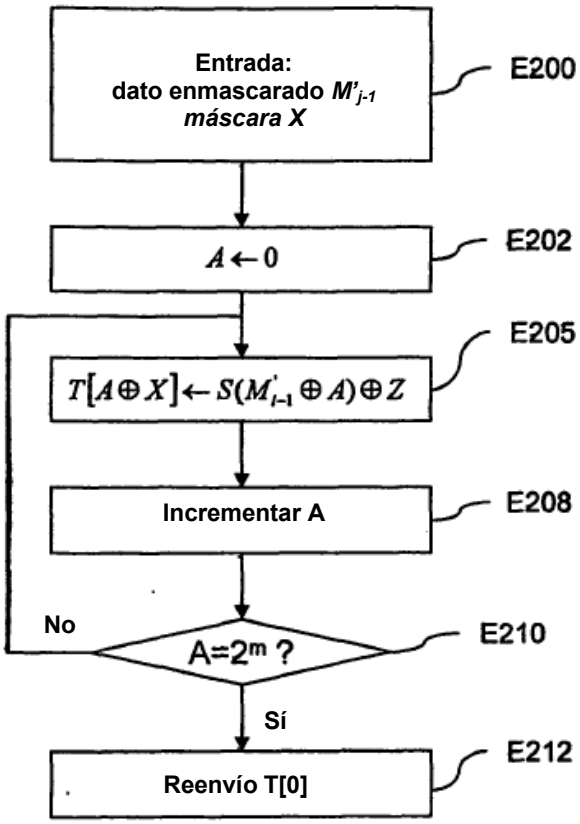

Figura 4

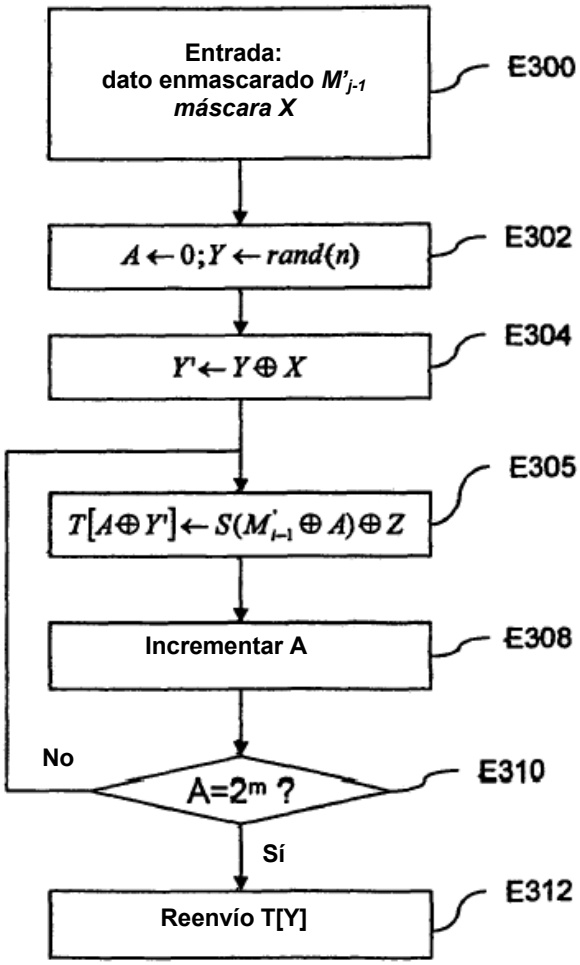

Figura 5

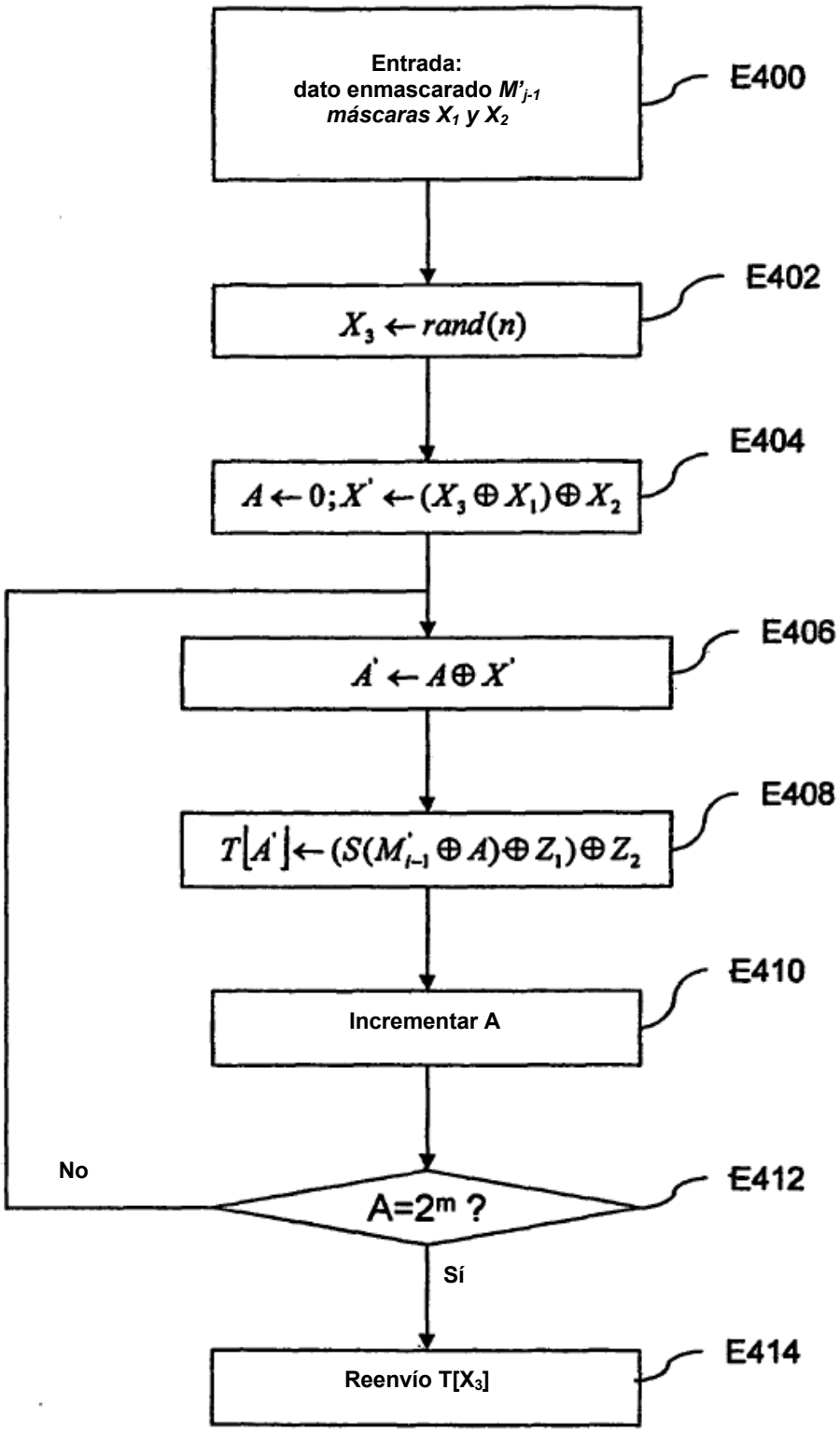
Entrada:
dato enmascarado M'j-1
máscaras X1 y X2
E400
X3 ← rand(n)
E402
A ← 0; X' ← (X3 ⊕ X1) ⊕ X2
E404
A' ← A ⊕ X'
E406
T[A'] ← (S(M'i-1 ⊕ A) ⊕ Z1) ⊕ Z2
E408
Incrementar A
E410
No
A=2m ?
E412
Sí
Reenvío T[X3]
E414

Figura 6

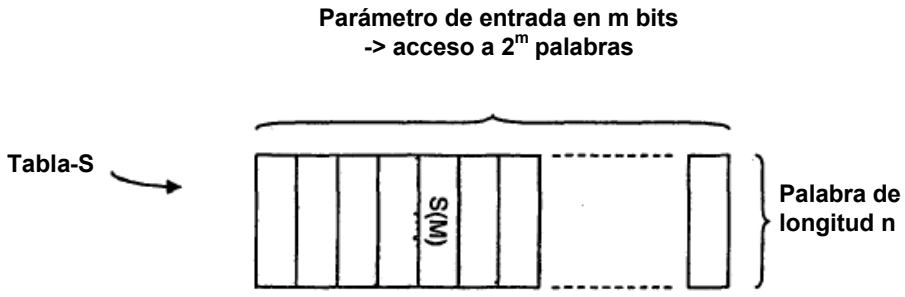

Figura 7

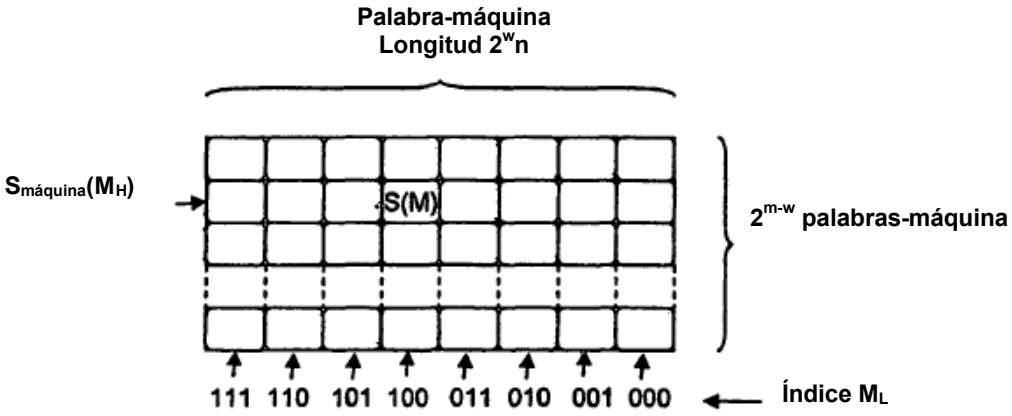

Figura 8

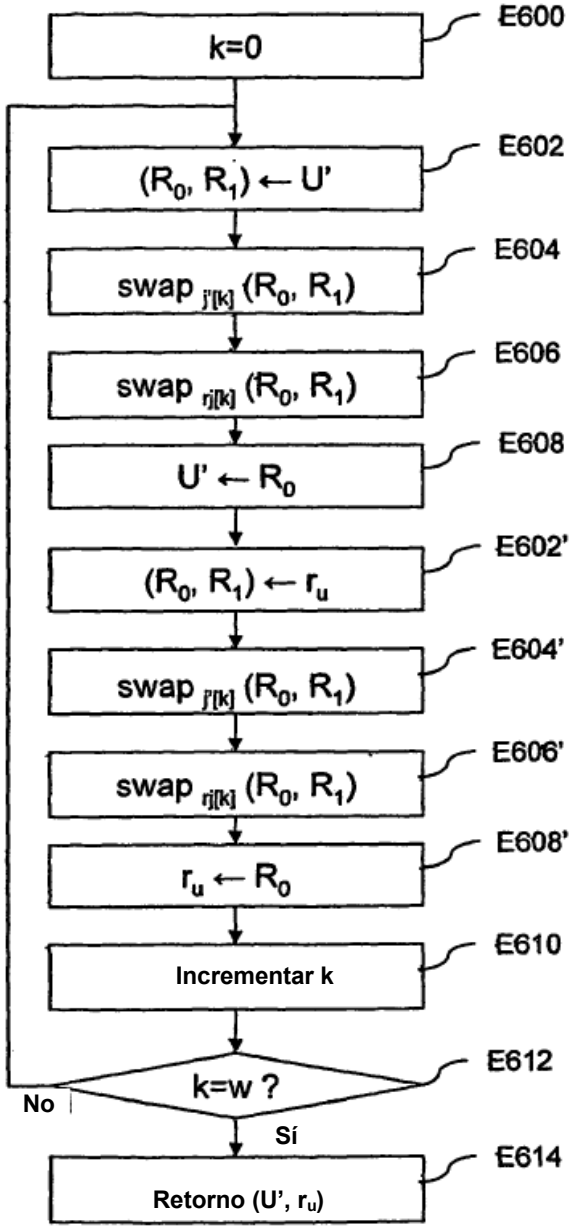

Figura 9

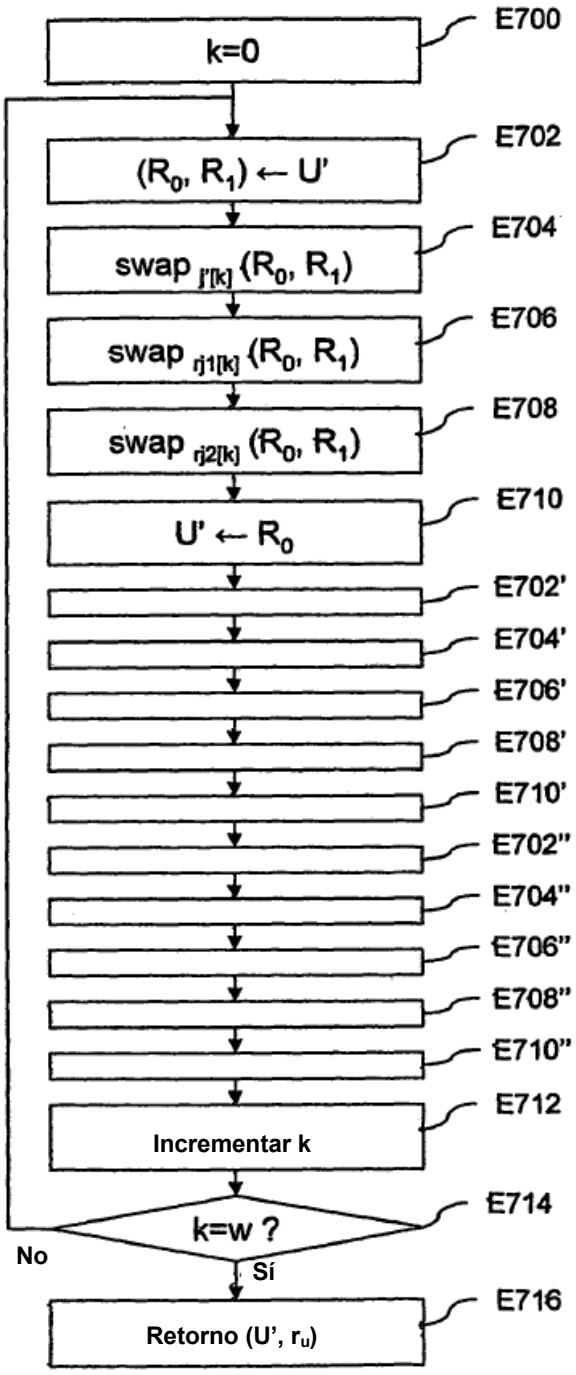

Figura 10

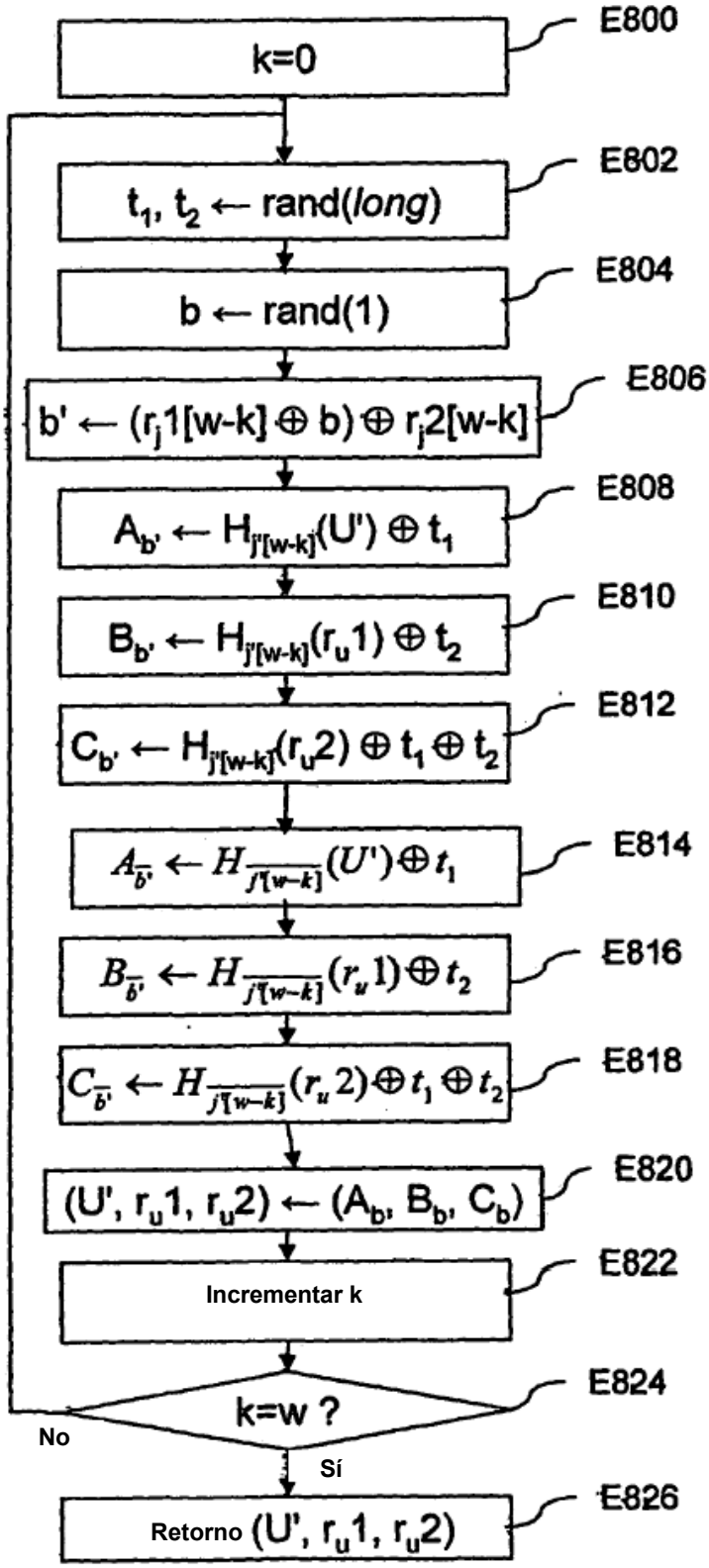
E800
k=0
E802
t1, t2 ← rand(long)
E804
b ← rand(1)
E806
b' ← (rj1[w-k] ⊕ b) ⊕ rj2[w-k]
E808
Ab' ← Hj'[w-k](U') ⊕ t1
E810
Bb' ← Hj'[w-k](ru1) ⊕ t2
E812
Cb' ← Hj'[w-k](ru2) ⊕ t1 ⊕ t2
E814
A b̄' ← H j'[w-k] (U') ⊕ t1
E816
B b̄' ← H j'[w-k] (ru1) ⊕ t2
E818
C b̄' ← H j'[w-k] (ru2) ⊕ t1 ⊕ t2
E820
(U', ru1, ru2) ← (Ab, Bb, Cb)
E822
Incrementar k
E824
k=w ?
No
Sí
E826
Retorno (U', ru1, ru2)

Figura 11