(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 573 644**

(51) Int. CI.:

H04L 9/00 (2006.01)
H04L 9/06 (2006.01)
H04L 9/08 (2006.01)
H04L 9/18 (2006.01)
H04L 29/06 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **28.12.2010 E 10809321 (2)**

(97) Fecha y número de publicación de la concesión europea: **30.03.2016 EP 2520041**

(54) Título: **Procedimiento de generación de tabla de consulta para una caja blanca criptográfica**

(30) Prioridad:

30.12.2009 FR 0959679

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
09.06.2016

(73) Titular/es:

KONINKLIJKE PHILIPS N.V. (100.0%)
High Tech Campus 5
5656 AE Eindhoven, NL

(72) Inventor/es:

BILLET, OLIVIER y
MACARIO-RAT, GILLES

(74) Agente/Representante:

ISERN JARA, Jorge

DESCRIPCIÓN

Procedimiento de generación de tabla de consulta para una caja blanca criptográfica

La presente invención se refiere al ámbito de la criptografía, especialmente aplicada a la distribución de contenidos multimedia.

Con la llegada de lo digital, y gracias a las redes de comunicación, la distribución de contenidos audiovisuales adquiere múltiples formas y nacen nuevos contextos de distribución que no existían hasta entonces.

La distribución masiva de tales contenidos plantea el problema de la protección de estos contenidos: si, ahora, estos son fáciles de distribuir, también es relativamente fácil copiarlos, rápidamente y de forma masiva.

Para proteger estos contenidos, se ha extendido con cierto éxito el uso de técnicas criptográficas. En efecto, la criptografía tiene su aplicación para brindar seguridad a transacciones entre dos o varias entidades, mediante la aplicación de una tarea criptográfica. Entre estas tareas criptográficas se encuentran el cifrado de mensajes, la firma electrónica o la autentificación de mensajes. Un procedimiento de cifrado consiste esencialmente en cifrar un mensaje de manera que solo un destinatario legítimo pueda descifrarlo por medio de un software y de un material criptográfico que haya adquirido de forma legal.

El problema de semejante técnica de cifrado es que el usuario legítimo que posee el software de descifrado puede intentar comprender los engranajes de este software, efectuando operaciones de ingeniería inversa.

Estas operaciones de ingeniería inversa tienen por objeto identificar los algoritmos utilizados en el software, incluso conseguir claves o secretos criptográficos distribuidos al usuario legítimo y utilizados por este software de descifrado.

La identificación de los algoritmos, así como la distribución masiva de las claves o los secretos criptográficos conseguidos, destruyen así los esfuerzos de protección realizados para obtener el procedimiento criptográfico de cifrado.

Con objeto de luchar contra esta amenaza de ingeniería inversa que afecta a los algoritmos criptográficos incorporados en software que consume contenidos multimedia, se ha propuesto un nuevo modelo de ataque criptográfico, denominado ataque en caja blanca, así como una estrategia de protección de este modelo, en el artículo “A White-Box DES Implementation for DRM Applications” [“Una implementación de la Norma de Cifrado de Datos de Caja Blanca para aplicaciones de Gestión de Derechos Digitales”] de Chow y otros. Los algoritmos criptográficos descritos en este artículo están constituidos por aplicaciones afines y aplicaciones no lineales con un muy pequeño número de variables en forma de tablas de memoria. De esta manera, se vuelve mucho más difícil identificar el algoritmo criptográfico utilizado, y es posible ocultar algunas etapas y/o valores durante la ejecución del algoritmo de descifrado.

Especialmente, las variables que transitan entre las distintas tablas, y observables por el usuario, las cuales corresponden a las variables de transición entre las distintas operaciones elementales del algoritmo criptográfico de descifrado, están codificadas mediante una función secreta de codificación. De esta manera, dado que los valores observados no son directamente los valores utilizados por las operaciones elementales criptográficas, sino su versión codificada, la ingeniería inversa se vuelve más difícil.

Sin embargo, esta estrategia de protección ha sido cripto-analizada con éxito, como se menciona en el artículo “Cryptanalysis of a white box AES implementation” [“Cripto-análisis de una implementación de la Norma de Cifrado Avanzada de caja blanca”] de Billet y otros. De esta manera, se ha demostrado que esta estrategia no cumplía los objetivos de seguridad pretendidos, especialmente debido a que la protección de las variables de transición observables por un atacante no era suficiente.

El documento de patente US2005/259814 A1 (Gebotys Catherine H [CA]) publicado el 24 de noviembre de 2005 expone un dispositivo que emplea una tabla de sustitución oculta para contrarrestar los ataques en los canales ocultos.

El documento de patente EP 1 615 098 A2 (Giesecke & Devrient GMBH [DE]) publicado el 11 de enero de 2006 expone una cadena de tablas de ocultación en un circuito de elaboración de datos dotado de seguridad.

El documento de patente EP 1 833 190 A1 (Research in Motion LTD [CA]) publicado el 12 de septiembre de 2007 expone la subdivisión de una tabla de sustitución oculta en un conjunto de sub-tablas.

El documento de patente US2007/14 478 A1 (Komano Yuichi [JP y otros]) publicado el 21 de junio de 2007 expone la dotación de seguridad a una tabla de sustitución sobre la base de una máscara seleccionada aleatoriamente.

Por lo tanto, un objeto de la presente invención consiste en mejorar la situación.

La presente invención propone a tal efecto un procedimiento de generación de una tabla de consulta utilizable en un procedimiento de procesamiento criptográfico que comprende el almacenamiento de una pluralidad de datos de entrada y de datos de salida, estando cada uno de dichos datos de entrada asociado a al menos uno de dichos datos de salida en la tabla, incluyendo el procedimiento, para cada uno de dichos datos de entrada, la obtención de al menos uno de dichos datos de salida mediante la aplicación de una función de codificación a un primer dato subsidiario y a un dato intermedio cifrado dependiente del dato de entrada.

De esta manera, el procedimiento de la invención propone una solución criptográfica más resistente a la ingeniería inversa que la estrategia de caja blanca expuesta anteriormente y permite proteger mejor los referidos algoritmos criptográficos en caja blanca.

El dato intermedio cifrado se obtiene ventajosamente mediante la aplicación de una función criptográfica al dato de entrada, lo que permite proteger esta función criptográfica de una deducción por medio de ingeniería inversa.

En un modo preferido de realización, para cada dato de entrada, se genera una pluralidad de primeros datos subsidiarios distintos y se obtiene una pluralidad de datos de salida mediante la aplicación, para cada uno de dichos primeros datos subsidiarios generados, de una función de codificación a dicho primer dato subsidiario y al dato intermedio cifrado. Esto permite el refuerzo de la protección de la función criptográfica evitando que exista una relación unívoca en la tabla de consulta. Mediante el uso de estos datos intermedios, es posible marcar, en el sentido de 'marca de agua' del término, el procedimiento de descifrado, bien para el transmisor, o bien para el destinatario, o bien para el contenido. De esta manera, si se eligen estos datos subsidiarios de una forma conocida por el dispositivo transmisor que emplea el software de cifrado, es posible marcar de manera personalizada un contenido en función de una identidad que puede ser la asociada al dispositivo transmisor, por ejemplo, la identidad de la persona tenedora de los derechos asociados al contenido, o la identidad asociada a un dispositivo receptor, por ejemplo, la identidad de una persona que ha adquirido derechos sobre un contenido que se le transmite cifrado. Por otra parte, es posible deslizar entre estos datos subsidiarios datos de señuelo que pueden servir, a continuación, para detectar un uso anormal del dispositivo de descifrado. Por ejemplo, estos datos de señuelo pueden utilizarse para detectar que una persona malintencionada intenta hacer ingeniería inversa en el dispositivo de descifrado y para que el dispositivo se coloque entonces en modo defensivo, haciendo imposible la culminación de esta acción de ingeniería inversa. En otro ejemplo, estos datos de señuelo pueden utilizarse para que el dispositivo receptor de descifrado adopte un comportamiento pilotado a distancia desde el dispositivo transmisor. Por lo tanto, el dispositivo receptor puede pilotarse con el fin de revelar su identidad. Esto ofrece una posibilidad de rastreo de traidores.

Ventajosamente, para cada dato de entrada, se genera una pluralidad de segundos datos subsidiarios distintos y se obtiene una pluralidad de datos de entrada codificados mediante la aplicación, para cada uno de dichos segundos datos subsidiarios generados, de una función de codificación a dicho segundo dato subsidiario y al dato de entrada. Esto permite asimismo el refuerzo de la protección de la función criptográfica evitando dar acceso directamente, mediante la tabla de consulta, a los datos de entrada.

Preferiblemente, los primeros datos subsidiarios y/o los segundos datos subsidiarios se generan aleatoriamente, lo que refuerza la protección de la función criptográfica.

En otro modo de realización ventajoso, el dato de entrada se obtiene mediante aplicación de una función preliminar a un dato de entrada pre-codificado, lo que refuerza la protección de la función criptográfica empleada con relación a una técnica de ingeniería inversa.

Preferiblemente, cuando el dato de entrada se obtiene mediante aplicación previa de una función previa de codificación a al menos un dato de entrada inicial, la función preliminar aplicada al dato de entrada es una función de descodificación correspondiente a dicha función previa de codificación. Es entonces posible proteger una sucesión de funciones parciales de cifrado en una caja blanca criptográfica, permitiendo únicamente la observación de variables de transición codificadas.

La presente invención tiene asimismo por objeto un procedimiento de generación de una pluralidad de tablas de consulta, que comprende la generación inicial de una primera tabla de consulta por medio del procedimiento de generación anterior en el que la función criptográfica se aplica directamente a al menos un dato de entrada inicial, seguida por la generación de una sucesión de tablas de consulta por medio del procedimiento de generación anterior.

En un modo de realización preferido del procedimiento de generación de una pluralidad de tablas anterior, este comprende una etapa final de generación de una última tabla de consulta en la que están almacenados una pluralidad de datos de entrada asociados, cada uno, a una pluralidad de datos de salida, obteniéndose cada uno de dichos datos de salida mediante la aplicación de una función criptográfica a un dato intermedio obtenido mediante la aplicación al dato de entrada, asociado a una función de descodificación correspondiente a la función de codificación empleada durante la generación de la última tabla generada en el transcurso del procedimiento anterior.

La invención se refiere además a un procedimiento de cifrado de un dato de entrada en un dato de salida, obteniéndose este dato de salida mediante la aplicación de una función de codificación a un primer dato subsidiario y a un dato intermedio cifrado obtenido mediante la aplicación de una función criptográfica al dato de entrada.

Ventajosamente, el procedimiento de cifrado incluye una etapa previa de almacenamiento de una pluralidad de datos subsidiarios utilizados para generar al menos una tabla de consulta, seleccionándose el primer dato subsidiario aleatoriamente entre dicha pluralidad de datos subsidiarios, lo que permite el descifrado del dato cifrado por medio de la tabla de consulta generada. Los datos así cifrados solo pueden descifrarse eficazmente en un dispositivo de descifrado que incluya una tabla de consulta como la referida.

La presente invención se refiere asimismo a una unidad de almacenamiento que comprende al menos un medio de almacenamiento en el que están almacenadas una primera pluralidad de datos de entrada y una segunda pluralidad de datos de salida. Para cada medio de almacenamiento, cada dato de entrada está asociado a al menos uno de dichos datos de salida, según una tabla de consulta generada por medio del procedimiento de generación anterior.

La presente invención se refiere asimismo a un dispositivo de implementación física de una tabla de consulta criptográfica que comprende una unidad de procesamiento de datos conectada a una unidad de programación, capaz de recibir al menos un medio de almacenamiento, estando la unidad de procesamiento dispuesta para generar, a partir de una pluralidad de datos de entrada, al menos una tabla de consulta criptográfica por medio del procedimiento de generación anterior, y la unidad de programación está dispuesta para almacenar dicha tabla de consulta en el medio de almacenamiento.

La presente invención se refiere además a un dispositivo de descifrado de un dato cifrado que comprende la unidad de almacenamiento anterior y un módulo de procesamiento conectado a cada uno de los medios de almacenamiento de la unidad de almacenamiento, estando dicho módulo de procesamiento dispuesto para leer un primer dato de entrada asociado al dato por descifrar en el medio de almacenamiento en el que está almacenada la última tabla generada, y para leer, sucesivamente en el orden contrario de generación, tablas sucesivamente almacenadas en los medios de almacenamiento, correspondiendo un dato de entrada, asociado al dato de salida, al dato de entrada leído en el medio de almacenamiento leído anteriormente.

El usuario de este dispositivo de descifrado solo puede observar variables de transición codificadas sin poder deducir de ello etapas de descifrado parcial implementadas sucesivamente en los medios de almacenamiento.

La presente invención se refiere finalmente a un producto de programa de ordenador grabado en un soporte de almacenamiento para su ejecución por medio de una unidad de procesamiento, que permite el empleo del procedimiento de generación anterior con objeto de obtener al menos una tabla de consulta criptográfica.

Otros detalles y ventajas de la invención se entenderán mejor a partir de ejemplos de empleo cuya descripción se lleva a cabo a continuación con referencia a los dibujos adjuntos, en los cuales:

- la figura 1A ilustra un procedimiento de generación de una tabla de consulta según la presente invención;
- la figura 1B ilustra la tabla de consulta generada por el procedimiento de generación según la presente invención;
- la figura 1C ilustra los procedimientos de cifrado y de descifrado que utilizan la tabla de consulta generada por el procedimiento de generación de la presente invención;
- la figura 2A ilustra un primer modo de realización del procedimiento de generación de la presente invención;
- la figura 2B ilustra la tabla de consulta generada por el primer modo de realización del procedimiento de generación de la presente invención;
- la figura 3 ilustra un segundo modo de realización del procedimiento de generación de la presente invención;
- las figuras 4A a 4C ilustran un procedimiento de generación de una pluralidad de tablas de consulta según la presente invención;
- la figura 5 ilustra un dispositivo de implementación física de una tabla de consulta criptográfica, generada por medio del procedimiento según la presente invención; y
- la figura 6 ilustra un dispositivo de descifrado criptográfico que utiliza al menos una tabla de consulta generada por medio del procedimiento según la presente invención.

A continuación, se hace referencia a la figura 1A que ilustra un procedimiento 100 de generación de una tabla de consulta según la presente invención.

Este procedimiento consiste en almacenar en una tabla de consulta T, para cada dato de entrada X(i) tomado en cierto número m (donde m es un entero superior a 1) de datos de entrada X(1),...,X(i),..., X(m), un número n de datos de salida y(i,j), siendo n un entero superior o igual a 1. La tabla de consulta T así generada contiene entonces n*m pares de datos codificados (X(i),y(ij)) y puede almacenarse en un medio de almacenamiento del tipo de memoria muerta fija. Los datos X(i) e y(i,j) pueden presentarse, por ejemplo, en forma binaria o de cualquier otra forma que permita un almacenamiento sencillo en un medio de almacenamiento clásico.

La figura 1A muestra el cálculo que permite, para un dato X(i) de entrada, obtener los datos de salida y(i,j) correspondientes. Para obtener la tabla de consulta completa, basta con reproducir este cálculo m veces para cada uno de los datos de entrada X(i) posibles, como se indica mediante el bucle recursivo formado por la etapa 109 de comprobación de fin de bucle y la etapa de incremento 111 de la variable i. Una vez efectuado el cálculo de los datos de salida y(i,j), correspondiente a cada dato de entrada X(i) para todos los datos de entrada X(i), la tabla de consulta T puede almacenarse en un medio de almacenamiento durante una etapa de almacenamiento 113.

En particular, para cada uno de los datos X(i) de entrada, se obtiene al menos un dato de salida asociado y(i,j) mediante la aplicación, durante la etapa 107 de codificación, de una función de codificación C con dos datos distintos: la función de codificación C se aplica a un primer dato subsidiario s(j), generado durante la etapa de generación 105, y a un dato intermedio cifrado Y(i) dependiente del dato de entrada X(i).

La presencia de un primer dato subsidiario s(j) en la codificación del dato intermedio Y(i) permite impedir a un usuario encontrar, mediante ingeniería inversa, variables utilizadas durante etapas de cifrado que preceden a la codificación y, por lo tanto, deducir de ello directamente el tipo de función de cifrado utilizada.

El dato intermedio cifrado Y(i) se obtiene ventajosamente mediante la aplicación, durante una etapa de cifrado 103, de una función criptográfica F al dato de entrada X(i). Esta función criptográfica puede corresponder a cualquier tipo de operación como, por ejemplo, una operación aritmética tal como una suma, una multiplicación, la elevación a una potencia o una permutación arbitraria.

Gracias a la aplicación de la función de codificación C al dato intermedio cifrado Y(i) así como al dato subsidiario s(j), la función criptográfica F empleada está “protegida” y no puede ser deducida por parte de un usuario que solo tenga acceso a los datos de entrada X(i) y de salida y(i,j).

En efecto, para poder deducir la función criptográfica F, el usuario necesita conocer tanto el dato al que se aplica directamente esta función F como el dato resultante directamente de la aplicación de esta función F.

De esta manera, aunque el usuario, conociendo el dato de entrada X(i), tiene así acceso al dato al que se aplica directamente la función criptográfica F, no tiene acceso al dato intermedio cifrado Y(i) resultante directamente de la aplicación de esta función F, ya que está oculta por medio de la función de codificación C.

Este primer dato subsidiario s(j) puede generarse, durante una etapa 105 de generación, mediante un proceso aleatorio, lo que refuerza la resistencia del procedimiento frente a un usuario malintencionado. Puede ser ventajoso, desde el punto de vista de la facilidad de implementación técnica, utilizar un proceso seudo-aleatorio para generar este dato subsidiario s(j).

En un modo preferido de realización, los distintos datos subsidiarios s(j) generados para un mismo dato de entrada X(i) se generan durante la etapa 105, de manera que sean distintos unos de otros, con objeto de reforzar la protección de la función de cifrado F. Como variante, este primer dato subsidiario puede generarse durante la etapa 105 en función del dato de entrada X(i), como se explicará más adelante.

Estas operaciones de generación de un primer dato subsidiario s(j) y de codificación de un dato de salida correspondiente y(i,j), tal que $y(i,j)=C(Y(i),s(j))$, se repiten preferiblemente n veces para cada dato de entrada X(i), de manera de obtener n datos de salida y(i,j), con j yendo de 1 a n, para cada dato de entrada X(i). Esto se simboliza en la figura 1A mediante el bucle recursivo formado por la etapa 108 de comprobación de fin de bucle y la etapa de incremento 110 de la variable j.

Una vez terminados ambos bucles recursivos, se dispone de un conjunto de pares $\{(X(i),y(i,j))\}$ $1 \leq i \leq m$, $1 \leq j \leq n$, que es posible almacenar, durante la etapa de almacenamiento 113, en forma de una tabla de consulta T en un medio de almacenamiento adecuado.

La tabla de consulta generada mediante el procedimiento de generación anterior se ilustra en la figura 1B.

En esta figura 1B, la tabla de consulta T se presenta en forma de 2 columnas y n*m líneas, incluyendo la primera columna los datos de entrada X(i) e incluyendo la segunda columna los datos de salida codificados y(i,j), asociados a estos datos de entrada X(i). Estos datos pueden clasificarse, por ejemplo, mediante el incremento de la primera variable i, lo que permite agrupar los pares de datos (X(i),y(i,j)) de forma creciente en función del dato de entrada X(i) utilizado para generar los datos de salida y(i,j), como se indica en la figura 1B.

En esta figura 1B, se observa bien que a un mismo dato de entrada X(i) le corresponde cierto número de datos de salida y(i,j), generados en función de datos subsidiarios s(j). El uso de una tabla de consulta T, en un dispositivo de usuario, permite encontrar directamente, a partir de un dato de salida y(i,j) transmitido por un servidor de distribución, un dato de entrada X(i) descifrado sin proporcionar información alguna sobre la función de cifrado empleada, ni dejar la posibilidad al usuario de obtener ninguna.

Esta tabla de consulta T constituye una caja negra en el sentido en que, para un dato de salida y(i,j) específico proporcionado por el usuario, esta tabla T solo devuelve un único dato de entrada X(i) correspondiente y, recíprocamente, sin dar información alguna sobre la manera en que se ha calculado el dato de salida y(i,j).

En una variante, si se consigue que los valores de X(i) sean los valores comprendidos entre 1 y m, se puede omitir la primera columna ya que, implícitamente, se puede deducir el valor de X(i) del número de línea.

La figura 1C ilustra los procedimientos de cifrado y de descifrado que utilizan la tabla de consulta generada por el primer modo de realización del procedimiento de generación de la presente invención.

Un dato X(i) a transmitir, por ejemplo un bloque de datos binarios de una película a transmitir desde un servidor de distribución hacia uno o varios usuarios que disponen de un dispositivo de descifrado provisto de una tabla de consulta T generada gracias al procedimiento de generación anterior, se cifra en primer lugar mediante el procedimiento 210 de cifrado, en un dato de salida y(i,j).

Para ello, se aplica primero la función de cifrado F, empleada durante la generación de la tabla de consulta T, al dato X(i), durante una primera etapa de cifrado 213, con objeto de obtener un dato de entrada cifrado Y(i).

A continuación, se genera un dato subsidiario s(i) correspondiente a uno de los datos subsidiarios s(j) empleados durante la generación de la tabla de consulta T, durante una etapa de generación 215. Esta etapa de generación 215 puede implementarse, por ejemplo, mediante la selección aleatoria de un dato subsidiario entre el conjunto almacenado, durante una etapa de almacenamiento 205, de los datos subsidiarios s(j) utilizados durante la generación de la tabla de consulta T durante una etapa de generación 200 similar a la etapa 100 anterior.

Se aplica finalmente la función de codificación C, empleada durante la generación de la tabla de consulta T, al dato de entrada cifrado Y(i) y al dato subsidiario s(j) generado, durante una etapa de codificación 217, con objeto de obtener un dato de salida y(i,j).

Gracias al empleo de las funciones de cifrado F y de codificación C empleadas durante la generación de la tabla de consulta T, y gracias a la generación de un dato subsidiario s(j) utilizado durante la generación de esta misma tabla de consulta T, el dato de salida y(i,j) obtenido corresponde a uno de los datos de salida almacenados en esta tabla T.

El dato de salida y(i,j) obtenido se transmite entonces hacia el(los) usuario(s) afectado(s) durante una etapa 220 de transmisión, que puede efectuarse con la ayuda de medios habituales de transmisión por cable, o inalámbricos.

Una vez recibido por el dispositivo de un usuario, este dato de salida y(i,j) se descifra en el transcurso de una etapa de descifrado 230. Esta etapa de descifrado 230 consiste en utilizar la tabla de consulta T generada según el procedimiento anterior, con las mismas funciones F y C que las empleadas durante la preparación del dato y(i,j), y en buscar en esta tabla T el dato de entrada X(i) asociado al dato de salida y(i,j) recibido. Este dato de entrada X(i) corresponde entonces al dato X(i) descifrado y la transmisión cifrada del dato X(i) queda entonces debidamente efectuada.

A continuación, se hace referencia a la figura 2A, que ilustra un primer modo de realización del procedimiento de generación de la presente invención.

Las etapas del procedimiento 100’ ilustrado en esta figura corresponden a las etapas del procedimiento 100 ilustrado en la figura 1A, con la diferencia de que la función de codificación Cy se emplea para codificar el dato de entrada cifrado Y(i) en función de cada primer dato subsidiario s(j), de manera que y(ij)=Cy(Y(i),s(j)).

La etapa de generación 105’ incluye además la generación, para cada valor de la variable j incrementado en función de la variable i, de un segundo dato subsidiario r(j). Entonces, es posible obtener un dato de entrada codificado x(i,j) a partir del dato de entrada X(i) y de este segundo dato subsidiario r(j).

Al igual que para el primer dato subsidiario s(j), este segundo dato subsidiario r(j) puede generarse mediante un proceso aleatorio o, más ventajosamente desde el punto de vista de la facilidad de implementación técnica, mediante un proceso seudo-aleatorio. En un modo preferido de realización, los distintos segundos datos subsidiarios r(j) generados para un mismo dato de entrada X(i) se generan, durante la etapa 105’, de manera que sean distintos unos de otros. Finalmente, como variante, este segundo dato subsidiario r(j) se genera en función del dato de entrada X(i), como se explicará más adelante.

Una segunda función de codificación Cx se aplica entonces, durante la etapa de codificación 107', al dato de entrada X(i) y al segundo dato subsidiario r(j) generado durante la etapa 105', con objeto de obtener un dato de entrada codificado x(i,j) que verifique x(i,j)=Cx(X(i),r(j)).

Por lo tanto, para cada par de variables (i,j), se obtienen un dato de entrada codificado x(i,j) y un dato de salida y(i,j) al término de la etapa de codificación 107'. Para un mismo dato de entrada X(i), se calculan así n pares de datos (x(i,j),y(i,j)), con j yendo de 1 a n.

Una vez terminados los dos bucles recursivos, se dispone de un conjunto de pares {(x(i,j),y(i,j))} $1 \leq i \leq m$, $1 \leq i \leq n$, que es posible almacenar, durante la etapa de almacenamiento 113', en forma de una tabla de consulta T' en un medio de almacenamiento adecuado.

La figura 2B ilustra la tabla de consulta generada mediante dicho primer modo de realización del procedimiento de generación de la presente invención.

En esta figura 2B, la tabla de consulta T' se presenta siempre en forma de 2 columnas y n*m líneas, incluyendo esta vez la primera columna los datos de entrada codificados x(i,j), mientras que la segunda columna incluye siempre los datos de salida y(i,j) asociados respectivamente a los datos de entrada codificados x(i,j).

Estos datos pueden clasificarse, por ejemplo, mediante incremento de la primera variable i, seguido por el incremento de la segunda variable j, lo que permite agrupar los pares de datos (x(ij),y(ij)) en función del dato de entrada X(i) utilizado para generarlos, como se muestra en la figura 2B.

De manera preferente, los datos se clasifican sin orden aparente con relación a i y j, de manera que un observador de la tabla no pueda agrupar los pares de datos (x(i,j),y(i,j)) en función del dato de entrada X(i) utilizado para generarlos. En un modo preferido de realización, los datos se clasifican por valores crecientes, o elegidos según un orden arbitrario, lo que facilita, más adelante, su búsqueda en la tabla.

La tabla de consulta T' así generada consiste por lo tanto en un conjunto de n*m pares de datos codificados (x(i,j),y(i,j)) y puede así presentarse en forma de una tabla de 2 columnas y n*m líneas, conteniendo cada línea de esta tabla un par de datos x(i,j) e y(i,j). Esta tabla de consulta puede almacenarse en un medio de almacenamiento del tipo de memoria muerta fija.

Se ve claramente, en esta figura 2B, que para cada dato de entrada X(i), existe cierto número de datos de salida y(i,j) asociados a cierto número de datos de entrada codificados x(i,j). Este modo de realización es ventajoso en el sentido de que la tabla T' generada ya no contiene directamente dato alguno de entrada X(i), sino solo datos codificados x(i,j) correspondientes a este dato de entrada.

Un usuario que efectúa ingeniería inversa en una tabla de consulta T' ya no tiene siquiera acceso a los datos de entrada X(i) a los que se aplica directamente la función de cifrado F. Este otro modo de realización es más resistente a un intento de deducción fraudulenta en la medida en que un usuario que tenga acceso a los datos de entrada codificados x(i,j) y de salida y(i,j) no tiene conocimiento directo ni del dato de entrada X(i), ni del dato cifrado Y(i). Por lo que le es tanto más difícil deducir la función criptográfica F utilizada.

A continuación, se hace referencia a la figura 3, que ilustra un segundo modo de realización del procedimiento de generación de la presente invención.

Las etapas del procedimiento 100'' ilustrado en esta figura corresponden a las etapas del procedimiento 100 ilustrado en la figura 1A, con la diferencia de que el dato x(i) al que se aplica el procedimiento es un dato pre-codificado, obtenido a partir del dato de entrada X(i) por medio de una función de codificación previa.

En tal caso, es ventajoso aplicar previamente otra función preliminar D al dato pre-codificado x(i), durante una etapa preliminar 101". Esta otra función preliminar D es preferiblemente una función que permite la descodificación de la función de codificación previa empleada para obtener este dato x(i) a partir del dato de entrada X(i); con el fin de que el cifrado de la etapa 103'' pueda afectar perfectamente al dato de entrada X(i) en sí y no a su forma pre-codificada.

Este otro modo de realización es resistente a un intento de deducción fraudulenta en la medida en que un usuario con acceso a los datos de entrada pre-codificados x(i) y de salida y(i,j) no tiene conocimiento directo ni del dato de entrada X(i), ni del dato cifrado Y(i). Le es tanto más difícil deducir la función criptográfica F utilizada.

Este otro modo de realización es especialmente ventajoso cuando el dato de entrada pre-codificado x(i) se obtiene a su vez mediante aplicación previa de otra función previa de codificación C_0 a un dato de entrada inicial. En tal caso, al elegir preferiblemente como función preliminar utilizada en la etapa 101 la función de descodificación D_0 correspondiente a esta función previa de codificación C_0, es posible encadenar una sucesión de operaciones criptográficas al mismo tiempo que solo quedan visibles datos codificados.

Esto permite implementar una función de cifrado global en caja blanca criptográfica, la cual está compuesta por una sucesión de N operaciones criptográficas parciales, cada una representada por una tabla de consulta T_k generada según la presente invención, y utilizando variables de transiciones codificadas que impiden al usuario deducir individualmente cada operación criptográfica y, por lo tanto, la función de cifrado global.

Esta función de cifrado global puede utilizar, por lo tanto, una sucesión de n tablas de consulta $T_1,...,T_k,...,T_N$ generadas según la presente invención, de tal manera que el dato de salida asociado a un dato de entrada inicial en la primera tabla T_1 constituye un dato de entrada para la segunda tabla T_2, que proporciona un dato de salida asociado para la tercera tabla T_3, y así sucesivamente hasta obtener, con la enésima tabla T_N, un dato de salida que corresponde finalmente a la aplicación de la función de cifrado al dato de entrada inicial.

Estas tablas de consulta $\{T_k\}_{1\leq k\leq N}$ pueden entonces utilizarse durante el descifrado, en una caja blanca criptográfica que incluye estas N tablas.

De este modo, para descifrar un dato cifrado por medio de la referida función de cifrado global, conviene encontrar, en la última tabla de consulta T_N, el dato de entrada asociado al dato de salida equivalente al dato por descifrar. Este dato de entrada en la última tabla T_N constituye un dato de salida para la penúltima tabla T_{N-1}, la cual proporciona un dato de entrada correspondiente para la tabla T_{N-1}, y así sucesivamente hasta obtener, con la primera tabla T_1, un dato de entrada que corresponde finalmente al descifrado del dato cifrado.

A continuación, se hace referencia a las figuras 4A-4C que ilustran un procedimiento de generación de una pluralidad de n tablas sucesivas de consulta $\{T_k\}_{1\leq k\leq N}$.

La figura 4A ilustra la etapa inicial 100_1 de generación de la primera tabla T_1.

Para cada dato de entrada inicial X(i) considerado, se aplica una función criptográfica F_1 a este dato de entrada inicial $X_1(i)$, durante una primera etapa de cifrado parcial 103_1, con objeto de obtener un dato intermedio cifrado inicial $Y_1(i)$.

Un dato subsidiario inicial $s_1(j)$ se genera asimismo durante una etapa de generación 105_1, por ejemplo, de manera seudo-aleatoria.

Una etapa 107_1 de codificación se efectúa entonces sobre el dato intermedio cifrado $Y_1(i)$, así como sobre el dato subsidiario inicial $s_1(j)$, por medio de una función de codificación C_1, con objeto de obtener un dato de salida $y_1(i,j)$. Esta operación se repite n veces, para datos subsidiarios iniciales $s_1(j)$, con j yendo de 1 a n, para cada dato de entrada inicial X(i).

Se calcula por lo tanto un conjunto de n*m pares de datos de entrada iniciales y de datos de salida asociados $\{(X(i), y_1(i,j))\}_{1\leq i\leq m,\ 1\leq j\leq n}$, y se forma la primera tabla de consulta T_1, por ejemplo, en forma de una matriz 2*(m*n) almacenada en un medio de almacenamiento adecuado.

Tras esta etapa inicial 100_1, el procedimiento de generación comprende una sucesión de etapas 100_k de generación de una tabla de consulta T_k, con k yendo de 2 a N.

La figura 4B ilustra la etapa 100_k según una etapa de generación anterior 100_{k-1}. Los datos de entrada utilizados en esta etapa 100_k son ciertos, incluso todos los datos de salida $\{y_{k-1}(i)\}_{1\leq i\leq m}$ obtenidos durante la etapa anterior 100_{k-1}. Para cada dato de entrada $y_{k-1}(i)$, que corresponde a una variable de transición entre la tabla T_{k-1} y la tabla T_k, se efectúan las siguientes etapas:

- El dato intermedio cifrado inicial $Y_{k-1}(i)$, obtenido al término de la etapa de cifrado 103_{k-1} de la etapa de generación anterior 100_{k-1} , se recupera primero mediante aplicación, durante la etapa de descodificación 101_k, de la función de descodificación D_{k-1} correspondiente a la función de codificación C_{k-1} utilizada durante la etapa de codificación 105_{k-1} de la etapa de generación anterior 100_{k-1}.

- Una función criptográfica F_k se aplica entonces, durante la etapa de cifrado parcial 103_k, a este dato intermedio cifrado inicial $Y_{k-1}(i)$ con el fin de obtener un nuevo dato intermedio cifrado $Y_k(i)$.

Este nuevo dato intermedio cifrado $Y_k(i)$ corresponde al cifrado del dato de entrada inicial X(i) por la composición de las funciones criptográficas $F_{0...}F_k$. Esta composición se efectúa sin dar un acceso cualquiera a un usuario, ya que este solo puede observar las variables transitorias $y_k(i)$, codificadas a su vez especialmente en función de un dato subsidiario que puede ser seudo-aleatorio. La composición de las funciones criptográficas $F_{0...}F_k$ queda así protegida.

Se genera asimismo un nuevo dato subsidiario $s_k(j)$ durante una etapa de generación 105_k, por ejemplo, de manera seudo-aleatoria. Sin embargo, en esta etapa de generación 105_k, este nuevo dato subsidiario $s_k(j)$ puede asimismo

depender del dato de entrada $y_{k-1}(i)$, en particular, del dato subsidiario anteriormente generado $s_{k-1}(j')$ durante la etapa de generación anterior 100_{k-1}, que se recupera gracias a la etapa de descodificación 101_k.

En un modo de realización ventajoso, el nuevo dato subsidiario $s_k(j)$ se calcula a partir de este antiguo dato subsidiario $s_{k-1}(j')$, por ejemplo, mediante la aplicación de una función seudo-aleatoria o de una función que garantiza que el nuevo dato subsidiario es distinto del antiguo dato subsidiario, con objeto de facilitar la generación del nuevo dato subsidiario $s_k(j)$.

Una vez generado el nuevo dato subsidiario $s_k(j)$, se utiliza una nueva función de codificación C_k, durante la etapa 107_k, para codificar el nuevo dato intermedio cifrado $Y_k(i)$ y el nuevo dato subsidiario $s_k(j)$, con el fin de obtener un nuevo dato de salida $y_k(i,j)$.

Por lo tanto, se calcula un conjunto de n*m pares de datos de entrada iniciales y de datos de salida asociados $\{(y_{k-1}(i),y_k(i,j))\}_{1\leq i\leq m,\ 1\leq j\leq n}$ y se forma la tabla de consulta T_k, siempre, por ejemplo, en forma de una matriz 2*(m*n) almacenada en un medio de almacenamiento adecuado.

La etapa de generación 100_k se repite entonces N-2 veces hasta una etapa 100_{N-1}, de manera similar a la etapa 100_k, para obtener las tablas de consulta T_2 a T_{N-1}. Cada tabla T_k entre estas está constituida por un conjunto de m pares de datos de entrada y de datos de salida asociados $\{(y_{k-1}(i),y_k(i,j))\}_{1\leq i\leq m,\ 1\leq i\leq m}$ calculado de la manera anterior.

Al término del cálculo de la tabla T_{N-1}, los datos de salida $y_{N-1}(i)$ están en forma codificada, por medio de la última función de codificación C_{N-1} empleada. Es entonces posible, en un modo de realización, generar una última tabla de consulta T_N correspondiente a la descodificación de cada dato $y_{N-1}(i)$ en un dato $y_N(i)$, por medio de la función de descodificación D_{N-1} correspondiente a la función de codificación C_{N-1}.

Mientras tanto, los datos descodificados obtenidos equivalen entonces a los últimos datos intermedios cifrados $Y_{N-1}(i)$ resultantes directamente de la función criptográfica F_{N-1}, lo que deja la posibilidad al usuario de acceder a esta última función criptográfica.

Otra variante consiste en generar la última tabla de consulta T_N de la manera ilustrada en la figura 4C, durante una etapa final 100_N de generación de tabla.

Esta etapa final 100_N comprende, para cada dato de entrada $y_{N-1}(i)$, una primera etapa 101_N de descodificación de este dato con el fin de obtener un nuevo dato intermedio cifrado $Y_{N-1}(i)$.

Se efectúa a continuación una última etapa 103_N de cifrado parcial, por medio de la función criptográfica F_N, para obtener un dato de salida final $Y_N(i)$. De esta forma, se calcula un conjunto de m pares de datos de entrada y de salida final asociada $\{(y_{N-1}(i), Y_N(i))\}_{1\leq i\leq m}$ y se forma la última tabla de consulta T_N siempre, por ejemplo, en forma de una matriz m*2 almacenada en un medio de almacenamiento adecuado.

Este dato de salida final $Y_N(i)$ corresponde entonces al cifrado global del dato de entrada inicial X(i) por medio de la composición de las funciones criptográficas $F_1,...,F_N$.

A continuación, se hace referencia a la figura 5, que ilustra un dispositivo 300 de implementación física de una tabla de consulta criptográfica según se ha descrito anteriormente.

Este dispositivo 300 comprende medios de entrada 310, como medios de conexión para la transferencia de datos, conectados a una unidad de procesamiento de datos 320. Esta unidad de procesamiento de datos puede ser un procesador, un microprocesador o cualquier tipo de componente capaz de efectuar cálculos sobre datos.

El dispositivo 300 comprende asimismo una unidad de programación 330 capaz de recibir al menos un medio de almacenamiento 340, y conectada a la unidad de procesamiento de datos.

En particular, el medio de almacenamiento 340 a recibir puede consistir en una memoria muerta, del tipo ROM (Read Only Memory), en la que solo es posible escribir datos una sola vez y que, por lo tanto, funciona solo en lectura. En tal caso, la unidad de programación 330 es una unidad capaz de programar esta memoria muerta.

La unidad de procesamiento 320 está, por una parte, dispuesta para recibir un conjunto de m datos de entrada X(i) y para generar, a partir de cada uno de estos datos de entrada X(i), datos de salida y(i,j) por medio de cálculo que implique, por ejemplo, una única función criptográfica F.

El conjunto de los n*m pares de datos de entrada y de salida así obtenidos, $\{(X(i),y(i,j))\}_{1\leq i\leq m,\ 1\leq i\leq m}$ forma una tabla de consulta T correspondiente a la función criptográfica F. La unidad de programación 330 está entonces dispuesta para almacenar esta tabla de consulta en el medio de almacenamiento 340.

En el caso en que la función criptográfica F está compuesta por N funciones criptográficas parciales F_k, la unidad de programación 330 puede almacenar las N tablas de consulta $T_1,...,T_N$ correspondientes a cada función F_k , unas tras otras en N medios de almacenamiento $340_1,...,340_N$, colocándose cada medio de almacenamiento 340_k por turno en la unidad de programación 330 para ser programado por medio de la tabla de consulta T_k que le corresponde. Este método de programación proporciona entonces N medios de almacenamiento $340_1,...,340_N$ separados que comprenden, cada uno, una tabla de consulta T_k asociada a una de las funciones criptográficas parciales F_k.

Una variante consiste en programar simultáneamente las N tablas de consulta $T_1,...,T_N$. Esto puede realizarse con una unidad de programación 330 dispuesta para recibir una unidad de almacenamiento 340 que comprende N medios de almacenamiento $340_1,...,340_N$. La unidad de programación 330 puede entonces programar cada uno de los medios de almacenamiento 340_k de la unidad de almacenamiento con la tabla de consulta T_k que le corresponde, y esta programación puede efectuarse en paralelo. Este procedimiento proporciona por lo tanto una unidad de almacenamiento 340 en la que están almacenadas todas las tablas de consulta $T_1,...,T_N$. Por lo tanto, la unidad de almacenamiento 340 está asociada a la función criptográfica global F. Este procedimiento presenta la ventaja de proporcionar una unidad de almacenamiento potencialmente más compacta y de permitir una implementación física más rápida, mediante el uso de cálculos en paralelo.

A continuación, se hace referencia a la figura 6, que ilustra un dispositivo de descifrado criptográfico 400 que emplea al menos una tabla de consulta generada por medio del procedimiento según la presente invención.

Este dispositivo de descifrado 400 puede utilizarse en cualquier sistema que reciba datos multimedia cifrados, como Set Top Box (Equipos de Sobremesa) que reciban programas cifrados como películas, lectores de soportes físicos de audio, como CD o discos ópticos, lectores de soportes físicos de vídeo como DVD, incluso módulos de seguridad del tipo dongle, o tarjeta con microprocesador, por ejemplo, capaces de descifrar un mensaje recibido cifrado.

El dispositivo de descifrado 400 comprende, por una parte, medios de recepción de datos 410 capaces de recibir un dato cifrado Y, como, por ejemplo, medios de conexión para la transferencia de datos, por cable o de forma inalámbrica.

Para poder descifrar este dato cifrado Y, el dispositivo de descifrado comprende una unidad de almacenamiento 440 que comprende medios de almacenamiento $440_1,...,440_N$ en los que se han almacenado respectivamente N tablas de consulta $T_1...,T_N$ generadas mediante el procedimiento de generación ilustrado anteriormente, en el que se han empleado las funciones F_k.

Esta unidad de almacenamiento 440 ha podido fabricarse por medio del dispositivo 300 de implementación física descrito anteriormente y almacenarse previamente en el dispositivo 400 antes de comercializarse. Esta unidad de almacenamiento 440 puede también programarse a distancia, mediante la descarga de las distintas tablas de consulta $T_1,...,T_N$ en una unidad de almacenamiento que comprende, o que puede dividirse, en N medios de almacenamiento $440_1,...,440_N$.

El dispositivo 400 comprende asimismo una unidad de procesamiento 420 capaz de recibir el dato a descifrar Y, y de utilizar sucesivamente los distintos medios de almacenamiento $440_1,...,440_N$ para descifrar Y. Para ello, la unidad de procesamiento 420 va a efectuar el proceso inverso del efectuado para deducir el dato Y del dato inicial X, durante el cifrado. Más concretamente:

- la unidad de procesamiento 420 accede, durante una primera etapa inicial parcial de descifrado, al último medio de almacenamiento 440_N generado con objeto de encontrar, en la tabla de consulta T_N correspondiente a la última función F_N que compone la función F, el dato de entrada Y_{N-1} correspondiente al dato por descifrar Y tomado como dato de salida en esta tabla T_N; y

- la unidad de procesamiento 420 accede, a continuación, sucesivamente en orden decreciente, a cada uno de los medios de almacenamiento 440_k generados con el fin de encontrar, en la tabla de consulta T_k correspondiente a la función F_k que compone parcialmente F, el dato de entrada Y_{k-1} correspondiente al dato Y_k tomado como dato de salida en esta tabla T_k, habiéndose obtenido este dato Y_k de esta manera, y así sucesivamente hasta el segundo medio de almacenamiento 440_2 en el que se almacena la segunda tabla de consulta T_2 correspondiente a la segunda función parcial F_2;

- la unidad de procesamiento 420 accede entonces, para terminar, al primer medio de almacenamiento 440_1 en el que está almacenada la primera tabla de consulta T_1. Utilizando el último dato de entrada obtenido Y_1 a partir de la tabla anterior T_2 como dato de salida de la tabla T_1, el dato de entrada asociado va a ser el dato inicial X.

Estas etapas sucesivas equivalen a encontrar el dato inicial X mediante la aplicación de la fórmula $X = F^{-1}(Y) = F_n^{-1} \circ \ldots \circ F_1^{-1}(Y)$.

Una vez encontrado el dato inicial X, puede ser leído, escuchado, visionado o transmitido con la ayuda de los medios de salida 430.

El dispositivo de descifrado 400 constituye una caja blanca en el sentido criptográfico del término, es decir, que su usuario puede ciertamente tener acceso eventualmente a las variables transitorias utilizadas entre cada medio de almacenamiento 440_k (por ejemplo, vigilando los datos que transitan en el bus que une el procesador 420 y la unidad de almacenamiento 440), pero no puede, en modo alguno, deducir de ellas información sobre las funciones criptográficas parciales F_k utilizadas para cifrar el dato Y.

Estas funciones F_k, sin embargo, virtualmente presentes en el dispositivo 400 por medio de las tablas de consulta T_k, son inaccesibles para el usuario y, por lo tanto, están protegidas del mismo.

Las etapas del procedimiento de generación de tablas de consulta descrito anteriormente pueden implementarse por medio de un programa de ordenador, utilizado por ejemplo en la unidad de procesamiento 320 descrita anteriormente. En consecuencia, la invención se refiere asimismo a un programa susceptible de ser ejecutado por un ordenador o por un procesador de datos, incluyendo este programa instrucciones para ordenar la ejecución de las etapas de un procedimiento de generación como el mencionado anteriormente.

Este programa puede utilizar cualquier lenguaje de programación, y encontrarse en forma de un código fuente, código objeto o código intermedio entre código fuente y código objeto, como en una forma en parte compilada o en cualquier otra forma deseable.

La invención se refiere asimismo a un soporte de información legible por un ordenador o un procesador de datos, y que comprende instrucciones de un programa como el mencionado anteriormente.

El soporte de información puede ser cualquier entidad o dispositivo capaz de almacenar el programa. Por ejemplo, el soporte puede comprender un medio de almacenamiento, como una ROM, por ejemplo, un CD-ROM o una ROM de circuito micro-electrónico, o también un medio de grabación magnético, por ejemplo, un disquete o un disco duro.

Por otra parte, el soporte de información puede ser un soporte transmisible, como una señal eléctrica u óptica, que puede enviarse por medio de un cable eléctrico u óptico, por radio o por otros medios. El programa según la invención puede, especialmente, descargarse de una red del tipo de Internet.

Alternativamente, el soporte de información puede ser un circuito integrado en el que se incorpora el programa, con el circuito adaptado para ejecutar o para ser utilizado en la ejecución del procedimiento en cuestión.

Por supuesto, la invención no se limita a los ejemplos de realización anteriores descritos y representados, a partir de los cuales se podrán prever otros modos y otras formas de realización, sin por ello salir del ámbito de la invención.

De este modo, se ha descrito, en las figuras 4A-4C anteriores, una invocación de tablas sucesivas de consulta T_k. Sin embargo, la invención no se limita a dicha utilización “lineal” de estas tablas, sino que se aplica asimismo a una utilización en red mallada donde la invocación se efectúa secuencialmente y paralelamente.

Además, el dato de entrada X(i) de una tabla T puede estar constituido por uno o varios datos de salida de tablas anteriores, por ejemplo, mediante concatenación de estos datos de salida, correspondiéndose entonces entre ellas las codificaciones afectadas.

Por lo tanto, en el ejemplo de la combinación de salidas de dos tablas T_1 y T_2 para la puesta en entrada de una tabla T_3, se obtienen a la salida de T_1 y T_2 ,respectivamente, los datos de salida y_1 e y_2 de manera que $y_1=C_1(Y_1,s_1)$ e $y_2=C_2(Y_2,s_2)$, a partir de los respectivos datos de entrada Y_1,y_2 y de los respectivos datos subsidiarios s_1,s_2.

La tabla T_3 toma entonces en entrada el dato Y_3 de manera que $Y_3=Y_1||Y_2$, donde || es el símbolo de la concatenación, lo que es posible con un dato subsidiario $s_3= s_1||s_2$ y una función de codificación C_3 como $C_3(Y_3,s_3)= C_1(Y_1,s_1) || C_2(Y_2,s_2)$.

REIVINDICACIONES

1. Procedimiento de generación de una tabla de consulta utilizable en un procedimiento de procesamiento criptográfico que comprende el almacenamiento (113) de una pluralidad de datos de entrada (X(i)) y de datos de salida (y(i,j)), donde cada uno de dichos datos de entrada está asociado a una pluralidad de dichos datos de salida en la tabla y, para cada uno de dichos datos de entrada (X(i)), se genera (105) una pluralidad de primeros datos subsidiarios distintos (s(j)) y la pluralidad de datos de salida (y(i,j)) se obtiene mediante aplicación (107), para cada uno de dichos primeros datos subsidiarios generados, de una función de codificación a dicho primer dato subsidiario (s(j)) y a un dato intermedio cifrado (Y(i)) dependiente del dato de entrada (X(i)).

2. Procedimiento de generación de una tabla de consulta según la reivindicación 1, donde el dato intermedio cifrado (Y(i)) se obtiene mediante la aplicación (103) de una función criptográfica (F) al dato de entrada (X(i)).

3. Procedimiento de generación de una tabla de consulta según la reivindicación 2, en el que, para cada dato de entrada (X(i)), se genera (105') una pluralidad de segundos datos subsidiarios distintos (r(j)) y se obtiene una pluralidad de datos de entrada codificados (x(i,j)) mediante la aplicación (107'), para cada uno de dichos segundos datos subsidiarios generados, de una función de codificación a dicho segundo dato subsidiario (r(j)) y al dato de entrada (X(i)).

4. Procedimiento de generación de una tabla de consulta según la reivindicación 3, en el que los primeros datos subsidiarios (s(j)) y/o los segundos datos subsidiarios (r(j)) se generan (105) aleatoriamente.

5. Procedimiento de generación de una tabla de consulta según una de las reivindicaciones 1 a 4, donde el dato de entrada (X(i)) se obtiene mediante la aplicación (101) de una función preliminar (D) a un dato de entrada pre-codificado (x(i)).

6. Procedimiento de generación de una tabla de consulta según la reivindicación 5, en el que el dato de entrada pre-codificado (yk-1(i)) se obtiene mediante la aplicación previa (105k-1) de una función previa de codificación (Ck-1) a al menos un dato intermedio cifrado previo (Yk-1(i)), donde la función preliminar aplicada (101k) al dato de entrada previo (yk-1(i)) es una función de descodificación (Dk) correspondiente a dicha función previa de codificación (Ck-1).

7. Procedimiento de generación de una pluralidad de tablas de consulta, que comprende la generación inicial (100_1) de una primera tabla de consulta (T_1) por medio de un procedimiento según una de las reivindicaciones 1 a 6, seguida por la generación (100_k) de una sucesión de tablas de consulta (T_k) por medio del procedimiento de generación según la reivindicación 6.

8. Procedimiento de generación de una pluralidad de tablas de consulta según la reivindicación 7, que comprende la generación final (100N) de una última tabla de consulta (TN) en la que están almacenados una pluralidad de datos de entrada (yN-1(i)) asociados, cada uno, a una pluralidad de datos de salida (YN-1(i)), obteniéndose cada uno de dichos datos de salida mediante la aplicación (103N) de una función criptográfica (FN) a un dato intermedio (YN-1(i)) obtenido mediante la aplicación (101N) al dato de entrada asociado (yN-1(i)) de una función de descodificación (DN) correspondiente a la función de codificación empleada durante la generación de la última tabla (TN-1) generada en el transcurso del procedimiento según la reivindicación anterior.

9. Procedimiento de cifrado de un dato de entrada (X(i)) en un dato de salida (y(i,j)), obteniéndose dicho dato de salida (y(ij)) mediante la aplicación de una función de codificación (217) a un primer dato subsidiario (s(j)) y un dato intermedio cifrado (Y(i)) obtenido mediante la aplicación (213) de una función criptográfica (F) al dato de entrada (X(i)), estando dicho dato de entrada asociado a una pluralidad de datos de salida según una tabla de consulta generada por medio de un procedimiento según una de las reivindicaciones 1 a 8.

10. Procedimiento de cifrado de un dato de entrada (X(i)) como un dato de salida (y(i,j)) según la reivindicación 9, que comprende una etapa previa de almacenamiento (205) de una pluralidad de datos subsidiarios utilizados para generar (200) al menos una tabla de consulta (T), seleccionándose (215) el primer dato subsidiario (s(j)) aleatoriamente entre dicha pluralidad de datos subsidiarios.

11. Unidad de almacenamiento (340,440) que comprende al menos un medio de almacenamiento (440k) en el que se almacenan una primera pluralidad de datos de entrada (X(i)) y una segunda pluralidad de datos de salida (y(i,j)), y para cada medio de almacenamiento, cada uno de los datos de entrada (X(i)) está asociado a una pluralidad de dichos datos de salida (y(i,j)) según una tabla de consulta (Tk) generada por medio de un procedimiento de generación según una de las reivindicaciones 1 a 8.

12. Dispositivo de implementación física de una tabla de consulta criptográfica que comprende una unidad de procesamiento (320) de datos conectada a una unidad de programación (330) capaz de recibir al menos un medio de almacenamiento (340), donde la unidad de procesamiento está dispuesta para generar, a partir de

una pluralidad de datos de entrada (X(i)), al menos una tabla de consulta criptográfica por medio del procedimiento según una de las reivindicaciones 1 a 8, estando la unidad de programación dispuesta para almacenar dicha tabla de consulta en el medio de almacenamiento.

13. Dispositivo de descifrado (400) de un dato cifrado, que comprende una unidad de almacenamiento (440) según la reivindicación 11, y un módulo de procesamiento (420) conectado a cada uno de los medios de almacenamiento de la unidad de almacenamiento, estando dicho módulo de procesamiento (440) dispuesto para leer un primer dato de entrada asociado al dato por descifrar en el medio de almacenamiento (3401) en el que se almacena la última tabla generada (TN) y para leer, sucesivamente en el orden inverso de generación de las tablas sucesivamente almacenadas en los medios de almacenamiento (340k), un dato de entrada (yk-1(i)) asociado al dato de salida correspondiente al dato de entrada (yk(i)) leído en el medio de almacenamiento leído anteriormente.

14. Producto de programa de ordenador grabado en un medio de almacenamiento para su ejecución por una unidad de procesamiento, donde durante la ejecución mediante el ordenador, este permite la aplicación del procedimiento de generación según una de las reivindicaciones 1 a 8, con objeto de obtener al menos una tabla de consulta criptográfica.

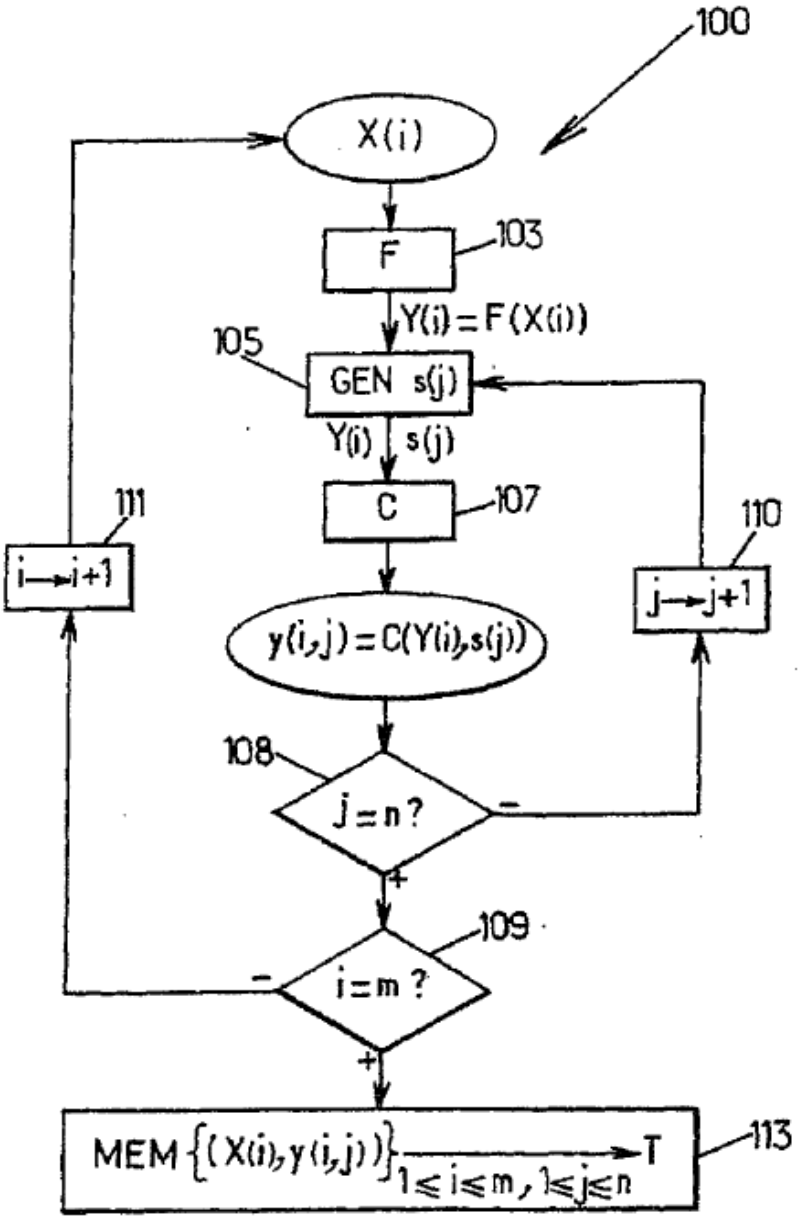
100
X(i)
F
103
Y(i) = F(X(i))
105
GEN s(j)
Y(i)
s(j)
C
107
111
i→i+1
110
j→j+1
y(i,j) = C(Y(i),s(j))
108
j=n?
-
+
109
i=m?
-
+
MEM {(X(i),y(i,j))} → T
1≤i≤m, 1≤j≤n
113

FIG.1A.

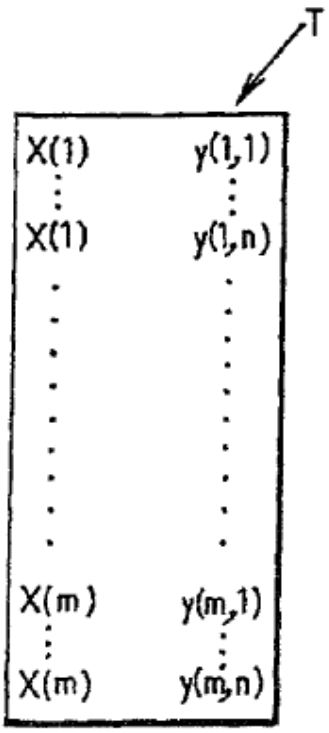
T
X(1) y(1,1)
X(1) y(1,n)
X(m) y(m,1)
X(m) y(m,n)

FIG.1B.

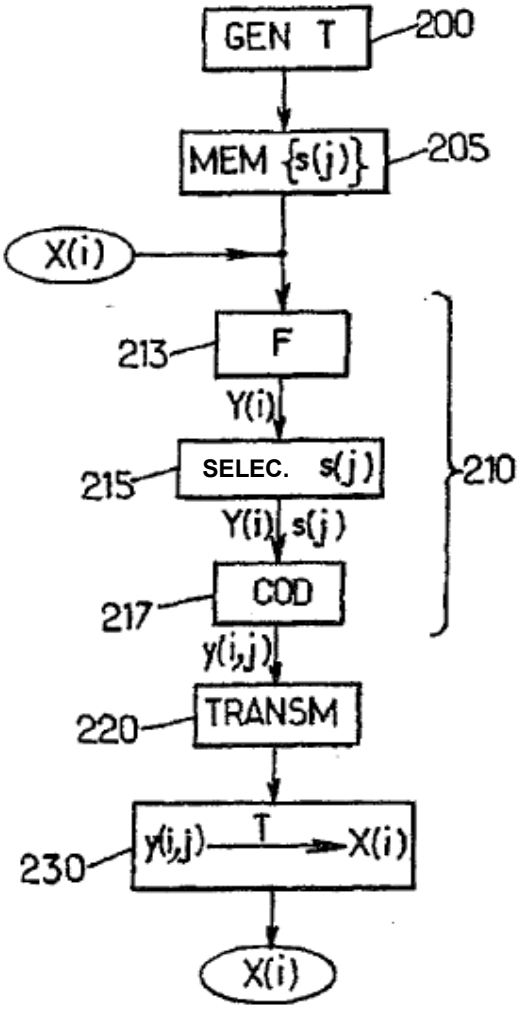
GEN T
200
MEM {s(j)}
205
X(i)
213
F
Y(i)
215
SELEC. s(j)
210
Y(i) s(j)
217
COD
y(i,j)
220
TRANSM
230
y(i,j) T X(i)
X(i)

FIG.1C.

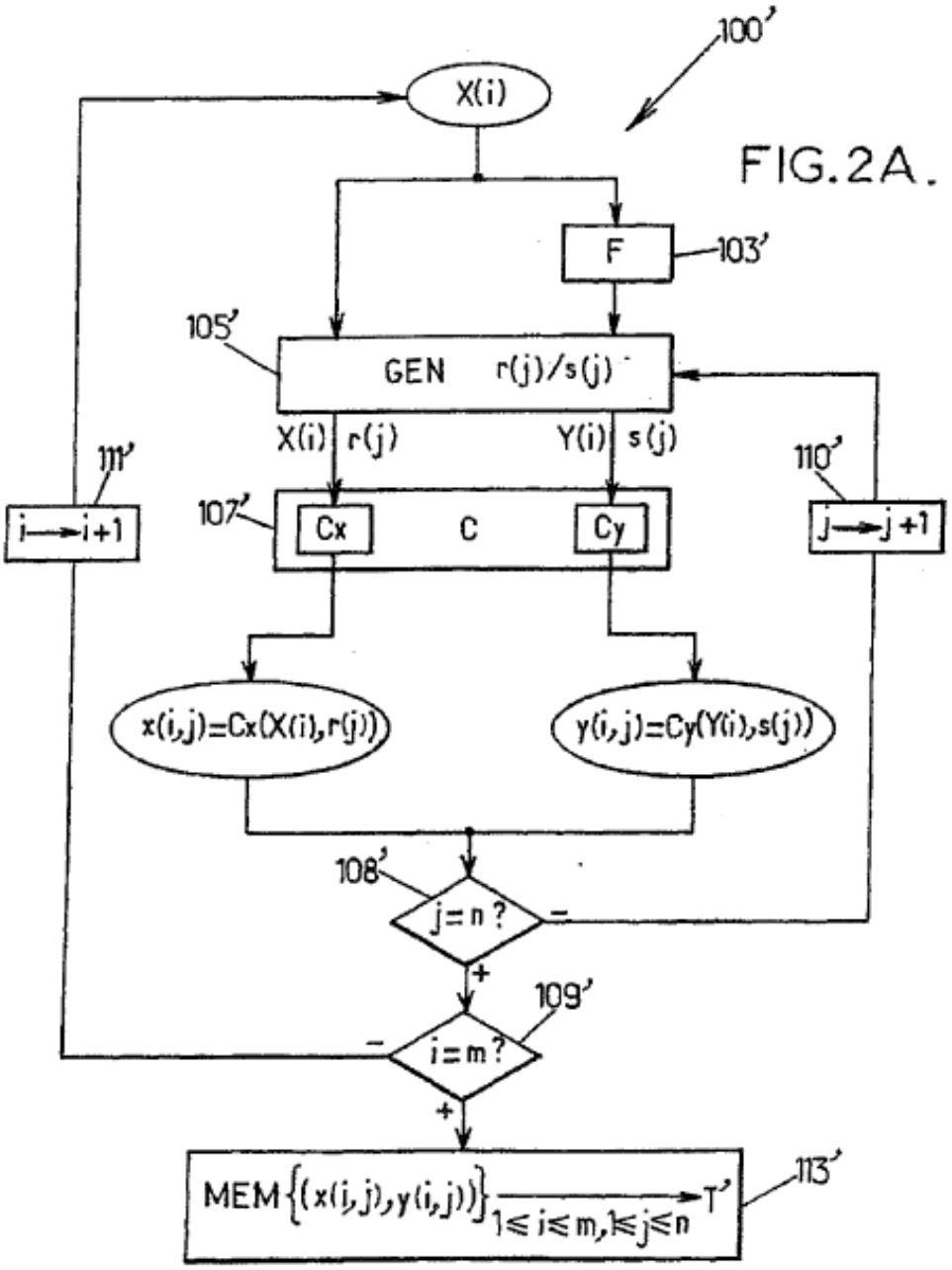
100'
X(i)
F
103'
105'
GEN r(j)/s(j)
X(i)
r(j)
Y(i)
s(j)
111'
110'
i→i+1
107'
Cx
C
Cy
j→j+1
x(i,j)=Cx(X(i),r(j))
y(i,j)=Cy(Y(i),s(j))
108'
j=n ?
=
+
109'
i=m ?
−
+
MEM{(x(i,j),y(i,j))} 1≤i≤m, 1≤j≤n → T'
113'

FIG.2A.

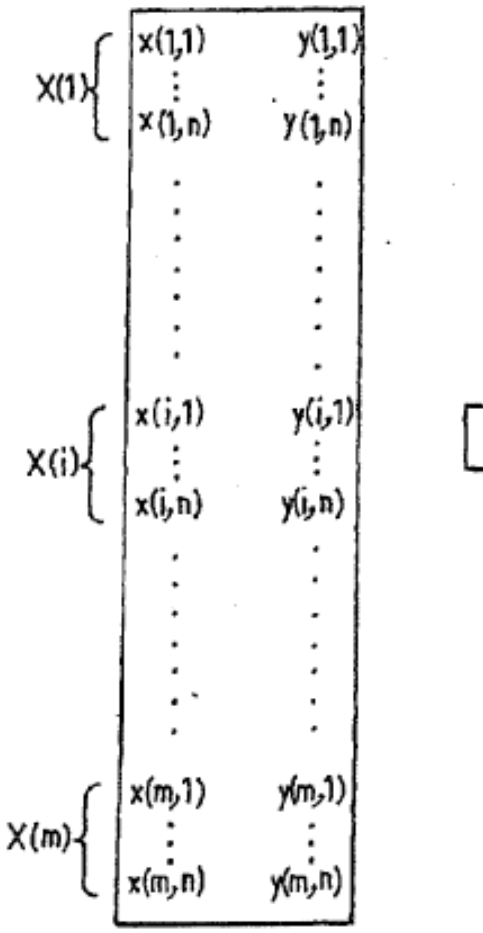
T'
X(1)
x(1,1) y(1,1)
x(1,n) y(1,n)
X(i)
x(i,1) y(i,1)
x(i,n) y(i,n)
X(m)
x(m,1) y(m,1)
x(m,n) y(m,n)

FIG.2B.

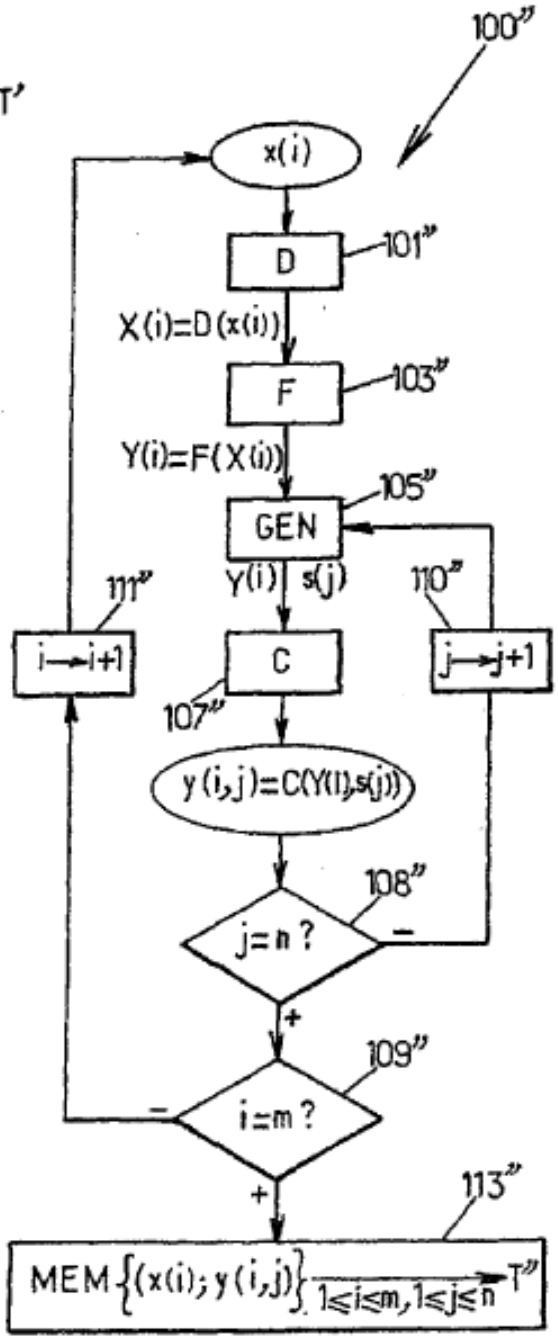
100"
x(i)
D
101"
X(i)=D(x(i))
F
103"
Y(i)=F(X(i))
GEN
105"
Y(i)
s(j)
111"
110"
i→i+1
C
j→j+1
107"
y(i,j)=C(Y(i),s(j))
108"
j=n ?
+
109"
i=m ?
+
113"
MEM {(x(i); y(i,j)} 1≤i≤m, 1≤j≤n T"

FIG.3.

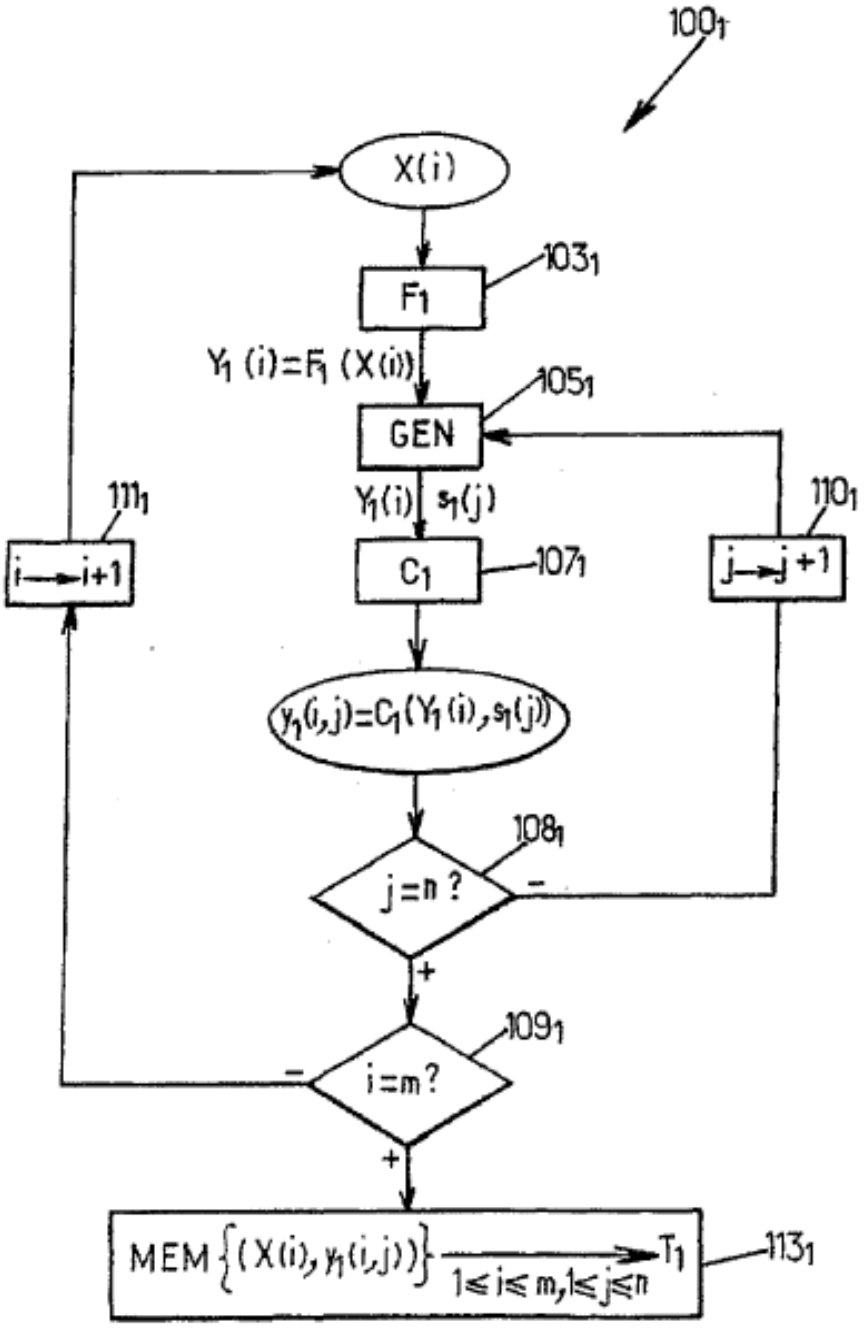
100₁
X(i)
F₁
103₁
Y₁ (i)=F₁ (X(i))
GEN
105₁
Y₁(i)
s₁(j)
C₁
107₁
111₁
i→i+1
110₁
j→j+1
y₁(i,j)=C₁(Y₁(i),s₁(j))
108₁
j=n ?
-
+
109₁
i=m?
-
+
MEM {(X(i), y₁(i,j))} → T₁
1≤i≤m, 1≤j≤n
113₁

FIG.4A.

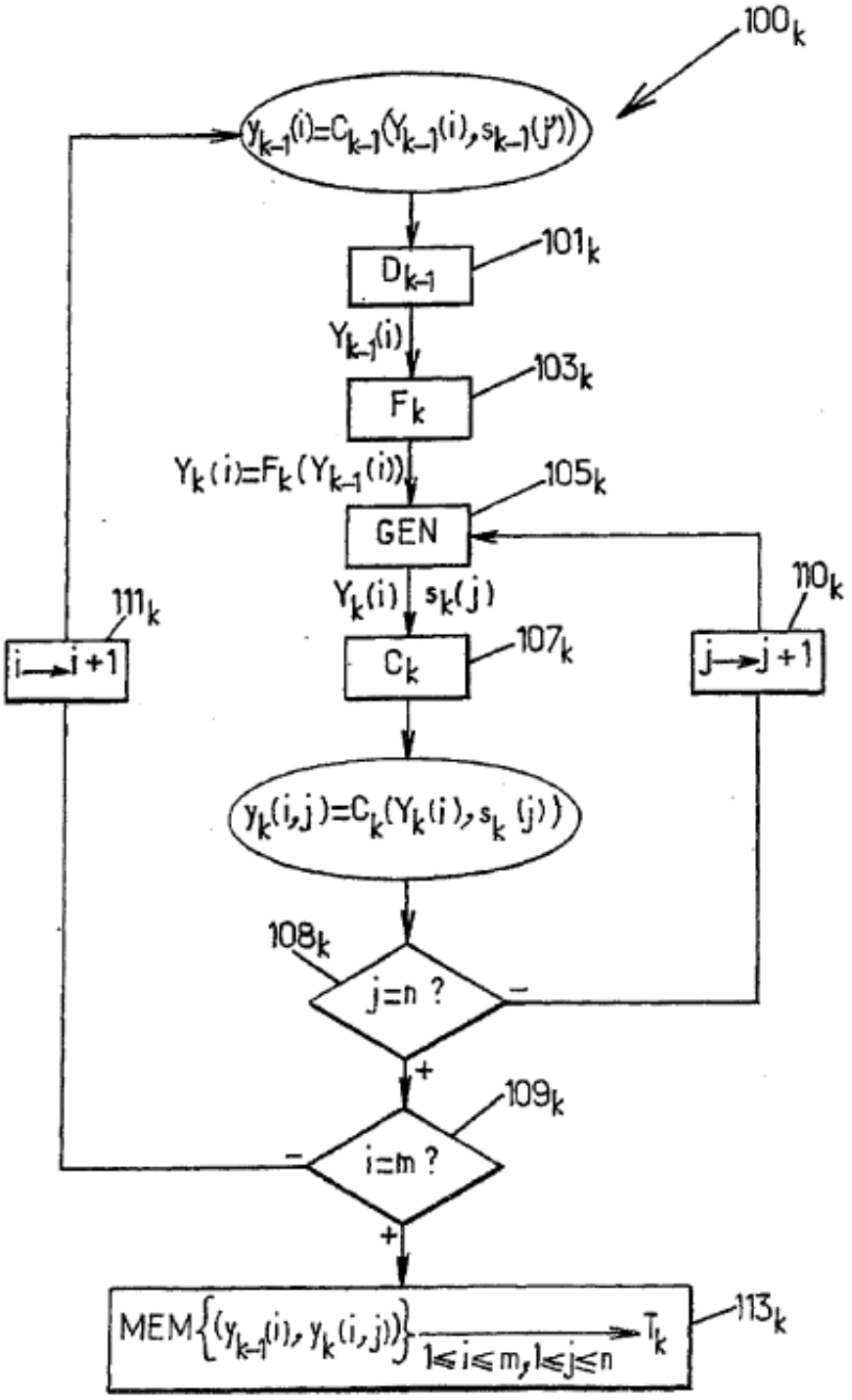
100k
yk-1(i)=Ck-1(Yk-1(i), sk-1(j'))
101k
Dk-1
Yk-1(i)
103k
Fk
Yk(i)=Fk(Yk-1(i))
105k
GEN
Yk(i)
sk(j)
111k
i→i+1
107k
Ck
110k
j→j+1
yk(i,j)=Ck(Yk(i), sk(j))
108k
j=n ?
+
109k
i=m ?
+
MEM{(yk-1(i), yk(i,j))} 1≤i≤m, 1≤j≤n → Tk
113k

FIG. 4B.

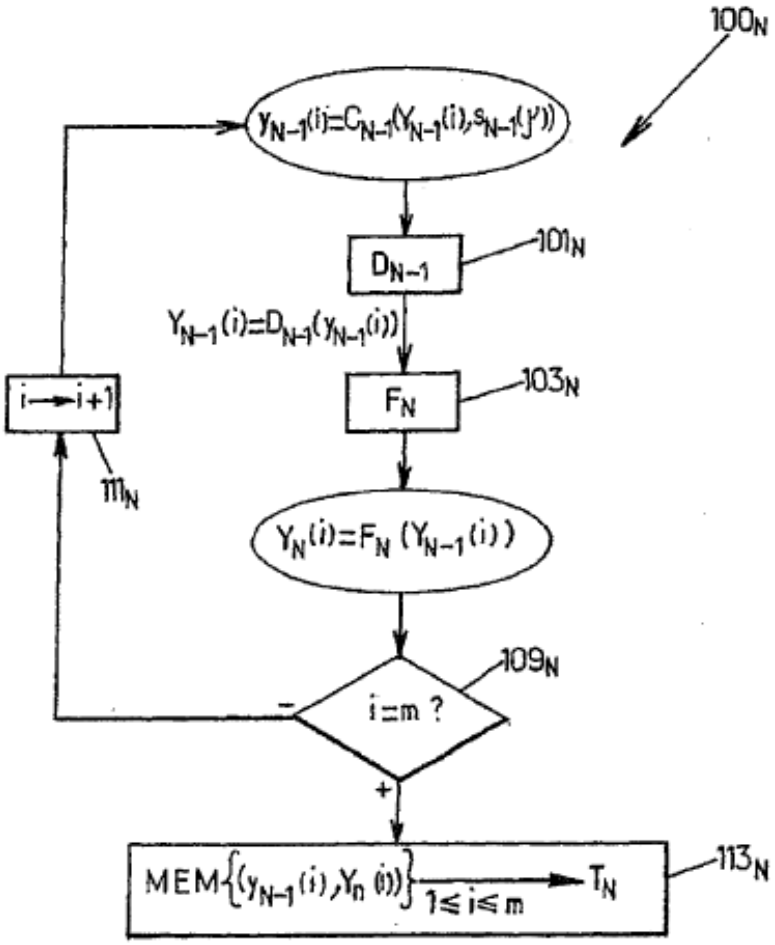
100N
yN-1(i)=CN-1(YN-1(i),sN-1(j'))
DN-1
101N
YN-1(i)=DN-1(yN-1(i))
FN
103N
i→i+1
111N
YN(i)=FN (YN-1(i))
109N
i=m ?
-
+
MEM{(yN-1(i),Yn (i))} 1≤i≤m → TN
113N

FIG.4C.

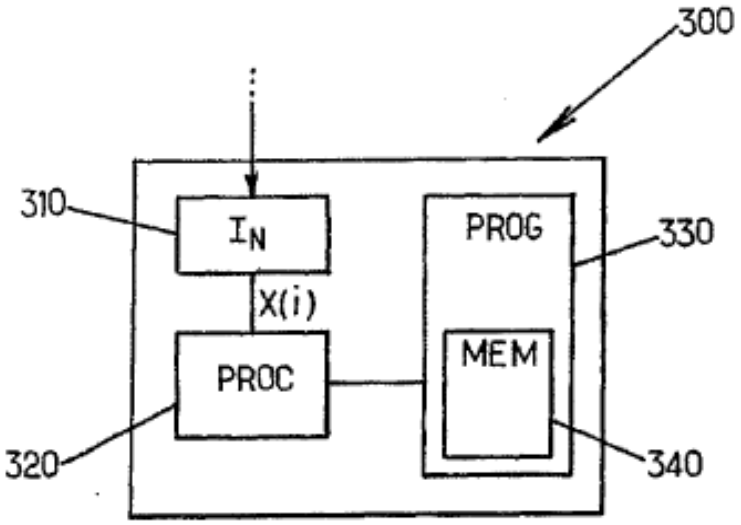
300
310
I_N
X(i)
PROC
320
PROG
330
MEM
340

FIG.5.

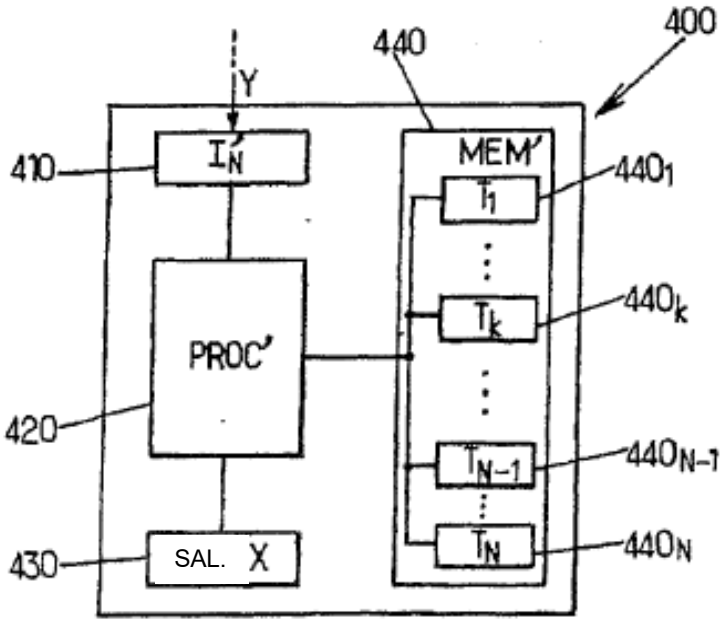
400
440
Y
410
I'_N
PROC'
420
430
SAL. X
MEM'
T_1
T_k
T_{N-1}
T_N
440_1
440_k
440_{N-1}
440_N

FIG.6.