



(10) **AT 515097 B1 2015-06-15**

(12)

Patentschrift

(21) Anmeldenummer: A 50236/2014
 (22) Anmeldetag: 31.03.2014
 (45) Veröffentlicht am: 15.06.2015

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

(56) Entgegenhaltungen:
 US 2010054461 A1
 US 2008112560 A1

(73) Patentinhaber:
 Hödl Josef
 2500 Baden (AT)

(74) Vertreter:
 WILDHACK & JELLINEK PATENTANWÄLTE
 OG
 WIEN

(54) Verschlüsselungsverfahren und Pseudo-Zufallszahlengenerator

(57) Die Erfindung betrifft ein Verfahren und eine Vorrichtung zur Verschlüsselung von Rohdaten, wobei die Rohdaten,
 - wobei eine Verschlüsselungsmaschine (200) umfassend eine Transformationseinheit (213) herangezogen wird, die abhängig von ihrem inneren Zustand eine Verschlüsselungsart in Form einer Transformationsvorschrift (g_i) bestimmt,
 - wobei die einzelnen Bytes (b_i) der Rohdaten durch verschiedene eineindeutige Transformationsvorschriften (g_i) verschlüsselt werden,
 - wobei die Transformationseinheit (213) vorab in einen von einem Passwort (W_0) abhängigen Initialzustand gebracht wird, und ein der Verschlüsselungsmaschine (200) zugehöriger deterministischer Zufallszahlengenerator (212) abhängig vom Passwort (W_0) ebenfalls in einen Initialzustand gebracht wird.

Erfindungsgemäß ist vorgesehen, dass der innere Zustand der Transformationseinheit (213) sowie der innere Zustand des Zufallszahlengenerators (212) nach einer Anzahl von Verschlüsselungen von Bytes, insbesondere nach der Verschlüsselung jedes Bytes, abhängig vom jeweils zu verschlüsselnden Rohdaten-Byte und/oder vom jeweiligen Endergebnis der Verschlüsselung geändert und aktualisiert wird und dieser neue innere Zustand der jeweils nächsten Verschlüsselung zugrunde gelegt wird.

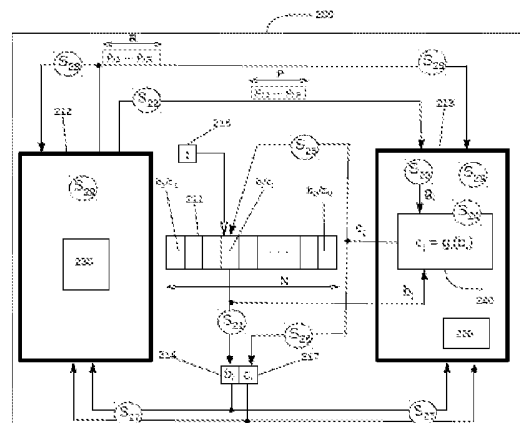


Fig. 1

Beschreibung

VERSCHLÜSSELUNGSVERFAHREN UND PSEUDO-ZUFALLSZAHLENGENERATOR

[0001] Die Erfindung betrifft ein symmetrisches Verschlüsselungsverfahren gemäß dem Patentanspruch 1 sowie ein dazugehöriges Entschlüsselungsverfahren gemäß dem Patentanspruch 8 und ein Verfahren zur Erstellung von Pseudo-Zufallszahlen - im Weiteren auch kurz nur als Zufallszahlen bezeichnet - für eine Verschlüsselung gemäß Anspruch 15 sowie ein Verfahren zur Ermittlung und Anwendung einer Transformationsvorschrift gemäß Anspruch 19. Weiters betrifft die Erfindung eine Verschlüsselungsmaschine gemäß dem Patentanspruch 24 sowie eine dazugehörige Entschlüsselungsmaschine gemäß dem Patentanspruch 25. Schließlich betrifft die Erfindung einen Datenträger mit einem Verschlüsselungs- oder Entschlüsselungsprogramm gemäß Anspruch 26 sowie einen Datenträger mit einer verschlüsselten Datenfolge gemäß Anspruch 27.

[0002] Aus dem Stand der Technik ist eine Vielzahl unterschiedlicher symmetrischer Verfahren zur Verschlüsselung von Rohdatenfolgen bekannt.

[0003] Aufgabe der Erfindung ist es, ein Verschlüsselungs- und Entschlüsselungsverfahren zur Verfügung zu stellen, das gegenüber den bekannten Verfahren eine verbesserte Sicherheit aufweist und eine effiziente und ressourcensparende Verwirklichung ermöglicht. Weiters ist es Aufgabe der Erfindung einen deterministischen Pseudo-Zufallszahlengenerator - im Weiteren auch kurz nur als Zufallszahlengenerator bezeichnet - bereitzustellen, der hoch chaotisches Verhalten zeigt. Weiters soll eine Transformationseinheit zur Verschlüsselung und/oder Entschlüsselung von Rohdaten bereitgestellt werden. Schließlich sollen eine Verschlüsselungsmaschine und eine Entschlüsselungsmaschine mit den genannten Vorteilen bereitgestellt werden.

[0004] Die Erfindung löst diese Aufgabe bei einem Verschlüsselungsverfahren der eingangs genannten Art mit den Merkmalen des Patentanspruchs 1. Die Erfindung löst diese Aufgaben bei einem Entschlüsselungsverfahren der eingangs genannten Art mit den Merkmalen des Patentanspruchs 8. Weiters löst die Erfindung diese Aufgabe bei einem Zufallszahlengenerator mit den Merkmalen des Patentanspruchs 15 und einer Transformationseinheit mit den Merkmalen des Patentanspruchs 19. Die Erfindung löst diese Aufgabe bei einer Verschlüsselungsmaschine der eingangs genannten Art mit den Merkmalen des Patentanspruchs 24. Die Erfindung löst diese Aufgaben bei einer Entschlüsselungsmaschine der eingangs genannten Art mit den Merkmalen des Patentanspruchs 25.

[0005] Zur sicheren Verschlüsselung von Rohdaten ist vorgesehen, dass der innere Zustand der Transformationseinheit sowie der innere Zustand des Zufallszahlengenerators nach einer Anzahl von Verschlüsselungen von Bytes, insbesondere nach der Verschlüsselung jedes Bytes, abhängig vom jeweils zu verschlüsselnden Rohdaten-Byte und/oder vom jeweiligen Endergebnis der Verschlüsselung geändert und aktualisiert wird und dieser neue innere Zustand der jeweils nächsten Verschlüsselung zugrunde gelegt wird,

[0006] wobei insbesondere der Zufallszahlengenerator Zufallszahlen erstellt, die für die Auswahl der Transformationsvorschriften verwendet werden, und/oder der Zufallszahlengenerator Zufallszahlen erstellt, die für die Aktualisierung der inneren Zustände des Zufallszahlengenerators und der Transformationseinheit herangezogen werden.

[0007] Mit dem vorliegenden Verfahren ist es möglich, sehr lange Passwörter, z.B. mit ca. 100 000 Bytes, zu verwenden, wobei die Änderung jedes Bytes den inneren Zustand und damit die Verschlüsselung maßgeblich beeinflusst.

[0008] Zur sicheren Entschlüsselung von Rohdaten ist vorgesehen,

[0009] - dass der innere Zustand der Transformationseinheit sowie der innere Zustand des Zufallszahlengenerators nach einer Anzahl von Entschlüsselungen von Bytes, insbesondere nach der Entschlüsselung jedes Bytes, abhängig vom jeweils verschlüsselten Byte und/oder vom jeweiligen entschlüsselten Byte geändert und aktualisiert wird und dieser neue innere

Zustand der jeweils nächsten Entschlüsselung zugrunde gelegt wird, wobei insbesondere

[0010] - der Zufallszahlengenerator Zufallszahlen erstellt, die für die Auswahl der inversen Transformationsvorschriften verwendet werden, und/oder der Zufallszahlengenerator Zufallszahlen erstellt, die für die Aktualisierung der inneren Zustände des Zufallszahlengenerators und der Transformationseinheit herangezogen werden.

[0011] Eine besonders einfache und sichere Verschlüsselung, bei der jedes Byte der verschlüsselten Datenfolge von jedem Byte der Rohdaten im jeweiligen Block abhängt, sieht vor, dass

[0012] a) die Transformationseinheit und der Zufallszahlengenerator in einen Initialzustand gesetzt werden,

[0013] b) die einzelnen sequentiell vorliegenden Bytes der Rohdaten innerhalb desselben Blocks hintereinander in der vorgegebenen Reihenfolge vom ersten bis zum letzten Byte von der Transformationseinheit verschlüsselt werden,

[0014] c) nachdem alle Bytes des Blocks verschlüsselt wurden, die Transformationseinheit sowie der Zufallszahlengenerator erneut in den Initialzustand gesetzt werden, und

[0015] d) die einzelnen sequentiell vorliegenden Bytes der im jeweiligen Block vorliegenden Daten innerhalb desselben Blocks entgegen der vorgegebenen Reihenfolge vom letzten bis zum ersten Byte von der Transformationseinheit verschlüsselt werden.

[0016] Eine Entschlüsselung sieht entsprechend vor, dass

[0017] a) die Transformationseinheit sowie der Zufallszahlengenerator in einen Initialzustand abhängig vom jeweiligen Passwort gesetzt wird.

[0018] b) die einzelnen sequentiell vorliegenden Bytes der im jeweiligen Block vorliegenden zu entschlüsselnden Daten innerhalb desselben Blocks entgegen der vorgegebenen Reihenfolge vom letzten bis zum ersten Byte von der Transformationseinheit entschlüsselt werden,

[0019] c) nachdem alle Bytes des Blocks entschlüsselt wurden, die Transformationseinheit sowie der Zufallszahlengenerator erneut in den Initialzustand gesetzt wird, und

[0020] d) die einzelnen sequentiell vorliegenden Bytes der so erhaltenen Daten innerhalb desselben Blocks hintereinander in der vorgegebenen Reihenfolge vom ersten bis zum letzten Byte von der Transformationseinheit entschlüsselt werden.

[0021] Zur Erhöhung der Abhängigkeiten der verschlüsselten Datenfolge von jedem einzelnen Byte der Rohdaten kann vorgesehen werden, dass die Verschlüsselungsschritte a) bis d) aus Anspruch 2 mehrfach für denselben Block wiederholt werden, wobei gegebenenfalls bei der letzten Wiederholung ausschließlich die Schritte a) und b) durchgeführt werden.

[0022] Eine Entschlüsselung sieht entsprechend vor, dass die Entschlüsselungsschritte a) bis d) mehrfach für denselben Block durchgeführt werden, wobei gegebenenfalls bei der letzten Durchführung ausschließlich die Schritte a) und b) ausgeführt werden.

[0023] Um Datenfolgen mit beliebiger vorgegebener Länge vorteilhaft verschlüsseln zu können und um Abhängigkeiten der verschlüsselten Datenfolge von sehr lange vorher verarbeiteten Daten zu ermöglichen, kann vorgesehen sein, dass nach der Durchführung des Schritts b) oder eines der Schritte b) oder alternativ nach der Durchführung des Schritts d) oder eines der Schritte d) und noch vor der darauffolgenden Zurücksetzung in den jeweiligen Initialzustand der innere Zustand der Transformationseinheit und des Zufallszahlengenerators abgespeichert wird und als neuer Initialzustand für die Verschlüsselung des jeweils nächsten Blocks herangezogen wird.

[0024] Eine Entschlüsselung sieht entsprechend vor, dass nach der Durchführung des Schrittes b) oder eines der Schritte b) oder alternativ nach der Durchführung des Schrittes d) oder eines der Schritte d) und noch vor der darauffolgenden Zurücksetzung in den jeweiligen Initialzustand der innere Zustand der Verschlüsselungsmaschine, bestehend aus dem inneren Zustand der

Transformationseinheit und dem inneren Zustand des Zufallszahlengenerators abgespeichert wird und als neuer Initialzustand für die Entschlüsselung des jeweils nächsten Blocks herangezogen wird.

[0025] Eine vorteilhafte Ermittlung von Transformationsvorschriften sieht vor, dass gemeinsam mit dem Setzen eines Initialzustands der Verschlüsselungsmaschine eine Anzahl von Grund-Transformationsvorschriften in Abhängigkeit vom Initialzustand vorgegeben wird, und

[0026] - abhängig vom inneren Zustand der Transformationseinheit eine Anzahl T aus diesen Grund-Transformationsvorschriften in einer vorgegebenen Reihenfolge ausgewählt wird,

[0027] - wobei die Transformationsvorschrift aus den einzelnen ausgewählten Grund-Transformationsvorschriften zusammengesetzt wird, sodass bei der Anwendung der Transformationsvorschrift auf das zu verschlüsselnde Byte die ausgewählten Grund-Transformationsvorschriften in der jeweiligen Reihenfolge auf das zu verschlüsselnde Byte angewandt werden.

[0028] Eine Entschlüsselung sieht entsprechend vor, dass gemeinsam mit dem Setzen eines Initialzustands der Entschlüsselungsmaschine eine Anzahl T von Grund-Transformationsvorschriften in Abhängigkeit vom Initialzustand der Transformationseinheit vorgegeben wird, und

[0029] - abhängig vom inneren Zustand eine Anzahl T aus diesen Grund-Transformationsvorschriften in einer vorgegebenen Reihenfolge ausgewählt wird, wobei die Transformationsvorschrift aus den einzelnen ausgewählten Grund-Transformationsvorschriften zusammengesetzt wird, sodass bei der Anwendung der Transformationsvorschrift auf das zu entschlüsselnde Byte die ausgewählten Grund-Transformationsvorschriften in der jeweils umgekehrten Reihenfolge auf das zu verschlüsselnde Byte angewandt werden, und

[0030] - dass insbesondere die Grund-Transformationsvorschriften der Transformationseinheit den Inversen der Grund-Transformationsvorschriften der Transformationseinheit einer mit demselben Passwort initialisierten Verschlüsselungsmaschine entsprechen, und

[0031] - dass insbesondere die Reihenfolge der Anwendung der einzelnen Grund-Transformationsvorschriften für die Bildung einer Transformationsvorschrift in der umgekehrten Reihenfolge wie bei der Verschlüsselung erfolgt.

[0032] Eine vorteilhafte Ermittlung von Transformationsvorschriften, mit der eine enorm große Menge an möglichen Transformationsvorschriften potentiell zur Verfügung steht, sieht vor, dass die Transformationsvorschrift aus den einzelnen ausgewählten Grund-Transformationsvorschriften sowie zusätzlich einer Spezial-Transformationsvorschrift zusammengesetzt wird, sodass bei der Anwendung der Transformationsvorschrift auf das zu verschlüsselnde Byte die ausgewählten Grund-Transformationsvorschriften in der jeweiligen Reihenfolge auf das zu verschlüsselnde Byte angewandt werden und, insbesondere danach, noch zusätzlich die Spezial-Transformationsvorschrift angewendet wird, und

[0033] dass die Spezial-Transformationsvorschrift bei jedem Verschlüsselungsschritt oder nach einer Anzahl von Verschlüsselungsschritten, abgeändert wird, insbesondere ausschließlich für eine vorgegebene Anzahl von Abänderungen, und

[0034] dass insbesondere die Verwendung und/oder Abänderung der Spezial-Transformationsvorschrift je nach Gegebenheit an- und abgeschaltet wird.

[0035] Eine Entschlüsselung sieht entsprechend vor, dass die Transformationsvorschrift aus der Spezial-Transformationsvorschrift und aus den einzelnen ausgewählten Grund-Transformationsvorschriften gebildet wird, sodass bei der Anwendung der Transformationsvorschrift auf das zu entschlüsselnde Byte, insbesondere zuerst, die Spezial-Transformationsvorschrift und, insbesondere anschließend, die ausgewählten Grund-Transformationsvorschriften in der jeweils umgekehrten Reihenfolge auf das zu entschlüsselnde Byte angewandt werden,

[0036] - dass die Spezial-Transformationsvorschrift bei jedem Entschlüsselungsschritt oder nach einer Anzahl von Entschlüsselungsschritten abgeändert wird, insbesondere ausschließlich für eine vorgegebene Anzahl von Änderungen, und

[0037] - dass insbesondere die Verwendung und/oder Abänderung der Spezial- Transformationsvorschrift je nach Gegebenheit an und ab geschaltet wird.

[0038] Zur vorteilhaften Initialisierung des Verschlüsselungsverfahrens anhand eines Passworts, kann vorgesehen sein, dass vor der Verschlüsselung der Rohdaten

[0039] a) als Passwort eine Anzahl von sequentiell angeordneten Bytes vorgegeben wird,

[0040] b) gegebenenfalls dem Passwort seine Länge in Bytes und/oder eine Checksumme beigefügt wird,

[0041] c) gegebenenfalls, sofern die Länge des Passwortes eine vorgegebene Länge unterschreitet, das Passwort, insbesondere durch Wiederholung des Passwortes, auf die vorgegebene Länge verlängert wird und somit ein normalisiertes Passwort erstellt wird,

[0042] d) ein vorgegebener Start-Zufallszahlengenerator mit vorgegebenen Werten initialisiert wird,

[0043] e) die vom Start-Zufallszahlengenerator erstellten Zufallszahlen mit den Bytes des, gegebenenfalls normalisierten Passwortes verknüpft werden und eine erste Initialisierungs-Datenfolge erstellt wird,

[0044] f) diese Initialisierungsfolge wird verwendet um eine erste Verschlüsselungsmaschine, umfassend eine erste Transformationseinheit und einen ersten Zufallszahlengenerator zu erstellen und zu initialisieren.

[0045] g) das erste Passwort mit der ersten Verschlüsselungsmaschine verschlüsselt wird, und damit ein zweites Passwort erhalten wird,

[0046] h) das erstellte zweite Passwort mit von dem Zufallszahlengenerator der ersten Verschlüsselungsmaschine erstellten Zufallszahlen verknüpft wird und so eine zweite Initialisierungs-Datenfolge erstellt wird, und diese zur Erstellung und Initialisierung einer weiteren Verschlüsselungsmaschine verwendet wird.

[0047] Eine Verbesserung der Initialisierung, die selbst bei kleinen Abweichungen vom Passwort zu völlig anderen Ergebnissen führt, sieht vor, dass i) wobei die folgenden Schritte j), k) und l) optional mehrfach, durchgeführt werden, nämlich

[0048] j) dass jeweils das für die Initialisierung der Verschlüsselungsmaschine verwendete Passwort (Wa mit dieser Verschlüsselungsmaschine verschlüsselt wird und dadurch ein nächstes Passwort erhalten wird,

[0049] k) dass dieses Passwort mit Zufallszahlen des jeweils zuletzt erstellten Zufallszahlengenerators verknüpft wird und so eine weitere Initialisierungs-Datenfolge erstellt wird, und

[0050] l) dass eine nachfolgende weitere Verschlüsselungsmaschine mit einer Transformationseinheit und einem Zufallszahlengenerator mit der jeweiligen zuletzt erstellten Initialisierungs-Datenfolge definiert, erstellt und initialisiert wird, und

[0051] m) dass die letzte derart erstellte Verschlüsselungsmaschine als Verschlüsselungsmaschine herangezogen wird.

[0052] Eine vorteilhafte Initialisierung einer Entschlüsselungsmaschine sieht vor, dass zur Initialisierung einer Entschlüsselungsmaschine

[0053] - eine Verschlüsselungsmaschine nach Anspruch 7 erstellt wird,

[0054] und dass anschließend eine Entschlüsselungsmaschine mit einem Zufallszahlengenerator und einer Transformationseinheit erstellt wird,

[0055] - der innere Zustand des Zufallszahlengenerators der Verschlüsselungsmaschine auf den inneren Zustand des Zufallszahlengenerators der Entschlüsselungsmaschine übertragen wird und

[0056] - der innere Zustand der Transformationseinheit der Verschlüsselungsmaschine auf den

inneren Zustand der Transformationseinheit der Entschlüsselungsmaschine übertragen wird, die Inversen der Transformationsvorschriften und/oder Grund-Transformationsvorschriften, sowie gegebenenfalls der Spezial-Transformationsvorschrift, ermittelt werden und als inverse Transformationsvorschriften und/oder inverse Grund-Transformationsvorschriften, sowie gegebenenfalls der inversen Spezial-Transformationsvorschrift, in die Transformationseinheit der Entschlüsselungsmaschine gespeichert werden

[0057] und insbesondere im Falle einer Änderung der Transformationsvorschriften während der Kodierung der Daten auch die nicht invertierten Transformationsvorschriften und/oder die nicht invertierte Grund-Transformationsvorschriften, sowie gegebenenfalls die nicht invertierten Spezial-Transformationsvorschrift, gespeichert werden.

[0058] Ein vorteilhaftes Verfahren zur Erstellung von deterministischen Pseudo-Zufallszahlen, das eine Rückkopplung mit den verschlüsselten Daten ermöglicht, sieht vor, dass eine Anzahl H von Auswahlregistern mit jeweils einer vorgegebenen Anzahl Z von Werten von 1 bis Z in beliebiger Reihenfolge vorgegeben werden,

[0059] a) dass dem ersten Auswahlregister eine erste Adressiereinheit vorgeschaltet ist, die mittels eines, insbesondere bei jedem Verschlüsselungsschritt zyklisch inkrementierten, in der Adressiereinheit befindlichen, ersten Zeigerwertes auf eine Speicherposition des ersten Auswahlregisters verweist,

[0060] b) dass die an der vom ersten Zeiger ausgewählten Speicherposition im ersten Auswahlregister befindliche Zahl ermittelt wird und diese mit der in der zweiten Adressiereinheit befindlichen Zahl verknüpft, insbesondere modulo Z genommen und um eins erhöht wird und dann der neue Inhalt der Adressiereinheit gebildet wird, und der so erhaltene Wert einen zweiten Zeigerwert bildet, der dann zur Adressierung einer Speicherposition im zweiten Auswahlregister herangezogen wird,

[0061] c) dass gegebenenfalls für die weiteren Adressiereinheiten und die weiteren Auswahlregister analog zur Adressiereinheit und zum Auswahlregister vorgegangen wird, und/oder für die weiteren Auswahlregister jeweils die an der vom jeweiligen Zeiger ausgewählten Speicherposition im jeweiligen Auswahlregister befindliche Zahl ermittelt wird und dem jeweils nächsten Zeiger zugewiesen oder zu diesem hinzugezählt oder von diesem abgezogen wird, und der Wert des jeweils nächsten Zeigers zur Adressierung einer Speicherposition im jeweils nächsten Auswahlregister herangezogen wird,

[0062] d) dass eine Anzahl von deterministischen Teil-Zufallszahlengeneratoren vorgegeben wird,

[0063] e) dass der vom letzten Zeigerwert der Adressiereinheit adressierte Wert des letzten Auswahlregisters zur Auswahl des Teil-Zufallszahlengenerators herangezogen wird,

[0064] f) dass gegebenenfalls zur Auswahl einer Mehrzahl von Zufallszahlengeneratoren jeweils der erste Zeiger nach vorgegebenen Kriterien abgeändert, insbesondere erhöht, verringert, inkrementiert oder dekrementiert, wird und die Schritte a) bis e) erneut durchgeführt werden, und

[0065] g) mit dem jeweils ausgewählten Teil-Zufallszahlengenerator eine Zufallszahl erstellt und zur Verfügung gehalten wird.

[0066] Eine bevorzugte externe Beeinflussung des Zufallszahlengenerators sieht vor, dass abhängig von am Eingang anliegenden Werten, insbesondere dem zu verschlüsselnden Byte und/oder dem verschlüsselten Byte und/oder von Zufallszahlen, der Wert eines der Adressiereinheiten abgeändert wird,

[0067] - wobei diese Werte mit dem in der jeweiligen Adressiereinheit befindlichen Wert verknüpft werden, vorzugsweise das Resultat dann modulo Z genommen und um eins erhöht wird, und der Wert dann der jeweiligen Adressiereinheit zugewiesen wird,

[0068] - wobei gegebenenfalls nach Vorliegen eines externen Triggersignals, insbesondere

ausgelöst durch eine vorzunehmende Verschlüsselung oder Entschlüsselung eines oder einer vorgegebenen Anzahl von Bytes, jeweils der Wert der nächsten Adressiereinheit abgeändert wird und für den Fall, dass im vorherigen Schritt der Wert der letzten Adressiereinheit abgeändert wurde, wiederum die erste Adressiereinheit abgeändert wird.

[0069] Alternativ oder zusätzlich kann zur externen Beeinflussung des Zufallszahlengenerators vorgesehen sein, dass abhängig von am Eingang anliegenden Werten, insbesondere einem zu verschlüsselnden Byte und/oder einem verschlüsselten Byte und/oder von Zufallszahlen, eines der Auswahlregister abgeändert wird, und insbesondere zwei in den Auswahlregistern abgespeicherte Werte miteinander vertauscht werden,

[0070] - wobei der Index des einen Wertes eine Reihe von vorgegebenen Werten nach einer vorab vorgegebenen Reihenfolge durchläuft, insbesondere durch zyklisches Hochzählen zwischen 1 und Z, und der Index des zweiten Wertes durch die am Eingang anliegenden Werte festgelegt wird,

[0071] - wobei gegebenenfalls nach Vorliegen eines externen Triggersignals, insbesondere ausgelöst durch eine vorzunehmende Verschlüsselung oder Entschlüsselung eines Bytes oder einer vorgegebenen Anzahl von Bytes, nach jeder Vertauschung oder nach einer vorgegebenen Anzahl von Vertauschungen in einem Auswahlregister zum jeweils nächsten Auswahlregister gewechselt wird und für den Fall, dass im vorherigen Schritt in dem letzten Auswahlregister bereits die vorgegebene Anzahl von Vertauschungen statt gefunden hat, wieder das erste Auswahlregister für Abänderungen herangezogen wird.

[0072] Bevorzugt kann vorgesehen sein, dass bei einem erfindungsgemäßen Verschlüsselungsverfahren oder Entschlüsselungsverfahren das erfindungsgemäße Verfahren zur Erstellung von Zufallszahlen verwendet wird.

[0073] Weiters betrifft die Erfindung ein Verfahren zur Ermittlung und Anwendung von Transformationsvorschriften. Die so erstellten Transformationsvorschriften sind vorteilhaft geeignet, bei Ver- und Entschlüsselungsverfahren verwendet zu werden. Hierbei ist vorgesehen, dass eine Anzahl G von Auswahlregistern mit jeweils einer vorgegebenen Anzahl A Werten 1 bis A in beliebiger Reihenfolge vorgegeben werden, wobei in jedem Auswahlregister jeweils eine Anzahl von einzelnen Speicherpositionen im jeweiligen Auswahlregister repräsentierenden Zahlen vorgegeben wird, wobei jede eine Speicherposition repräsentierende Zahl jeweils einmal im Auswahlregister enthalten ist,

[0074] a) dass dem ersten Auswahlregister eine erste Adressiereinheit vorgeschaltet ist, die mittels eines, insbesondere bei jedem Verschlüsselungsschritt zyklisch inkrementierten, in der Adressiereinheit befindlichen ersten Zeigerwertes auf eine Speicherposition des ersten Auswahlregisters verweist,

[0075] b) dass die an der vom ersten Zeiger ausgewählten Speicherposition im ersten Auswahlregister befindliche Zahl ermittelt wird und diese mit der in der zweiten Adressiereinheit befindlichen Zahl verknüpft, insbesondere modulo A genommen und um eins erhöht wird, und dann der neue Inhalt der Adressiereinheit gebildet wird, und der so erhaltene Wert einen zweiten Zeigerwert bildet, der dann zur Adressierung einer Speicherposition im zweiten Auswahlregister herangezogen wird,

[0076] c) dass gegebenenfalls für die weiteren Adressiereinheiten und die weiteren Auswahlregister analog zur Adressiereinheit und zum Auswahlregister vorgegangen wird, und/oder

[0077] dass für die weiteren Auswahlregister jeweils die an der vom jeweiligen Zeiger ausgewählten Speicherposition im jeweiligen Auswahlregister befindliche Zahl ermittelt wird und dem jeweils nächsten Zeiger zugewiesen oder zu diesem hinzugezählt oder von diesem abgezogen wird, und der Wert des jeweils nächsten Zeigers zur Adressierung einer Speicherposition im jeweils nächsten Auswahlregister herangezogen wird,

[0078] d) dass eine Anzahl von Grund-Transformationsvorschriften vorgegeben wird,

[0079] e) dass der vom letzten Zeigerwert der Adressiereinheit adressierte Wert des letzten

Auswahlregisters zur Auswahl der Grund-Transformationsvorschriften herangezogen wird,

[0080] f) dass gegebenenfalls zur Auswahl einer Mehrzahl von Grund-Transformationsvorschriften jeweils der erste Zeiger nach vorgegebenen Kriterien abgeändert, insbesondere erhöht, verringert, inkrementiert oder dekrementiert, wird und die Schritte a) bis e) erneut durchgeführt werden, und

[0081] g) ein Datenwert, insbesondere ein zu verschlüsselndes Byte oder ein zu entschlüsselndes Byte, angegeben wird und die ausgewählte Transformationsvorschrift auf den Datenwert angewendet wird, gegebenenfalls die mehreren ausgewählten Grund-Transformationsvorschriften hintereinander, im Falle der Verschlüsselung in der Reihenfolge ihrer Auswahl oder im Falle der Entschlüsselung entgegen der Reihenfolge ihrer Auswahl, auf den Datenwert angewendet werden und das Ergebnis der Anwendung zur Verfügung gehalten wird. Das Verfahren erlaubt eine vorteilhafte Rückkopplung und führt zu schwer vorhersehbaren Transformationsvorschriften.

[0082] Eine bevorzugte externe Beeinflussung des Zufallszahlengenerators sieht vor, dass abhängig von am Eingang anliegenden Werten, insbesondere dem zu verschlüsselnden Byte und/oder dem verschlüsselten Byte und/oder von Zufallszahlen, der Wert eines der Adressiereinheiten abgeändert wird,

[0083] - wobei diese Werte mit dem in der jeweiligen Adressiereinheit befindlichen Wert verknüpft werden, vorzugsweise das Resultat dann modulo A genommen und um eins erhöht wird und der Wert dann der jeweiligen Adressiereinheit zugewiesen wird,

[0084] - wobei gegebenenfalls nach Vorliegen eines externen Triggersignals, insbesondere ausgelöst durch eine vorzunehmende Verschlüsselung oder Entschlüsselung eines oder einer vorgegebenen Anzahl von Bytes, jeweils die nächste Adressiereinheit abgeändert wird und für den Fall, dass im vorherigen Schritt die letzte Adressiereinheit abgeändert wurde, wiederum die erste Adressiereinheit abgeändert wird.

[0085] Alternativ oder zusätzlich kann zur externen Beeinflussung des Zufallszahlengenerators vorgesehen sein, dass abhängig von am Eingang anliegenden Werten, insbesondere einem zu verschlüsselnden Byte und/oder einem verschlüsselten Byte und/oder Zufallszahlen eines der Auswahlregister abgeändert wird, und insbesondere zwei in den Auswahlregistern abgespeicherte Werte miteinander vertauscht werden,

[0086] - wobei der Index des einen Wertes eine Reihe von vorgegebenen Werten nach einer vorab vorgegebenen Reihenfolge durchläuft, insbesondere durch zyklisches Hochzählen zwischen 1 und Z, und der Index des zweiten Wertes durch die am Eingang anliegenden Werte festgelegt wird,

[0087] - wobei gegebenenfalls nach Vorliegen eines externen Triggersignals, insbesondere ausgelöst durch eine vorzunehmende Verschlüsselung oder Entschlüsselung eines Bytes oder einer vorgegebenen Anzahl von Bytes, nach jeder Vertauschung oder nach einer vorgegebenen Anzahl von Vertauschungen in einem Auswahlregister zum jeweils nächsten Auswahlregister gewechselt wird und für den Fall, dass im vorherigen Schritt in dem letzten Auswahlregister bereits die vorgegebene Anzahl von Vertauschungen statt gefunden hat, wieder das erste Auswahlregister für Abänderungen herangezogen wird.

[0088] Eine Transformationseinheit, die eine extrem große Anzahl an Transformationsvorschriften ermöglicht, sieht vor, dass abhängig von am Eingang anliegenden Werten, insbesondere einem zu verschlüsselnden Byte und/oder einem verschlüsselten Byte und/oder Zufallszahlen die Transformationsvorschriften oder die Grund-Transformationsvorschriften modifiziert werden,

[0089] - wobei die Transformationsvorschriften zu den Transformationsvorschriften invers sind,

[0090] - wobei nach einer vorgegebenen Anzahl von Abänderungen in einer Transformationsvorschrift oder Grund-Transformationsvorschrift, zur jeweils nächsten Transformationsvorschrift oder Grund-Transformationsvorschrift, gewechselt wird und für den Fall, dass im vorherigen Schritt die letzte Transformationsvorschrift oder Grund-Transformationsvorschrift, abgeändert

wurde, wieder die erste Transformationsvorschrift oder Grund-Transformationsvorschrift, für Abänderungen herangezogen wird.

[0091] Bevorzugt kann vorgesehen sein, dass bei einem erfindungsgemäßen Verschlüsselungsverfahren oder Entschlüsselungsverfahren das erfindungsgemäße Verfahren zur Ermittlung und Anwendung einer Transformationsvorschrift verwendet wird.

[0092] Weiters betrifft die Erfindung eine Verschlüsselungsmaschine, die eine besonders sichere Verschlüsselung gewährleistet, eine Aktualisierungseinheit, die den Zustandsspeicher der Transformationseinheit nach einer Anzahl von durchgeführten Verschlüsselungen, insbesondere nach der Verschlüsselung jedes einzelnen Bytes, abhängig vom jeweils verschlüsselten Byte sowie vom jeweiligen Endergebnis der Verschlüsselung und/oder abhängig von vom Zufallszahlengenerator erstellten Zufallszahlen ändert und aktualisiert, und die gegebenenfalls den Zustandsspeicher des Pseudo-Zufallszahlengenerators nach einer Anzahl von durchgeführten Verschlüsselungen, insbesondere nach der Verschlüsselung jedes einzelnen Bytes, abhängig vom jeweils zu verschlüsselnden Byte sowie vom jeweiligen Endergebnis der Verschlüsselung und abhängig von vom Zufallszahlengenerator vorher erstellten Zufallszahlen ändert und aktualisiert.

[0093] Die Erfindung betrifft auch eine entsprechende Entschlüsselungsmaschine eine Aktualisierungseinheit, die den Zustandsspeicher der Transformationseinheit nach einer Anzahl von durchgeführten Entschlüsselungen, insbesondere nach der Entschlüsselung jedes einzelnen Bytes, abhängig vom jeweils zu entschlüsselnden Byte sowie vom jeweiligen Endergebnis der Entschlüsselung und/oder abhängig von vom Zufallszahlengenerator erstellten Zufallszahlen ändert und aktualisiert, und die gegebenenfalls den Zustandsspeicher des Pseudo-Zufallszahlengenerators nach einer Anzahl von durchgeführten Entschlüsselungen, insbesondere nach der Entschlüsselung jedes einzelnen Bytes, abhängig vom jeweils zu entschlüsselnden Byte sowie vom jeweiligen Endergebnis der Entschlüsselung und abhängig von vom Zufallszahlengenerator vorher erstellten Zufallszahlen ändert und aktualisiert.

[0094] Hierbei kann insbesondere vorgesehen sein, dass die Verschlüsselungsmaschine zur Durchführung sämtlicher vorstehend genannter Verfahren zur Verschlüsselung ausgebildet ist.

[0095] Analog kann vorteilhaft vorgesehen sein, dass die Entschlüsselungsmaschine zur Durchführung sämtlicher vorstehend genannter Verfahren zur Entschlüsselung ausgebildet ist.

[0096] Ein Programm zur Durchführung eines der erfindungsgemäßen Verfahren kann auf einem Datenträger abgespeichert werden.

[0097] Weiters kann auch eine verschlüsselte Datenfolge auf einem Datenträger abgespeichert werden.

[0098] Fig. 1 zeigt eine bevorzugte Ausführungsform einer Verschlüsselungsmaschine.

[0099] Fig. 2 und 3 zeigen eine mögliche Art der Verschlüsselung sowie Entschlüsselung.

[00100] Fig. 4 und 5 zeigen eine bevorzugte Art der blockweisen Verschlüsselung sowie Entschlüsselung.

[00101] Fig. 6 bis 9 zeigen eine bevorzugte Art der blockweisen Verschlüsselung sowie Entschlüsselung mit erhöhter Sicherheit.

[00102] Fig. 10 zeigt eine bevorzugte Ausführungsform eines Zufallszahlengenerators für eine Verschlüsselungsmaschine.

[00103] Fig. 11 zeigt eine bevorzugte Ausführungsform einer Transformationseinheit für eine Verschlüsselungsmaschine.

[00104] Fig. 12 zeigt eine vorteilhafte Initialisierung einer Verschlüsselungsmaschine.

[00105] Fig. 13 zeigt eine bevorzugte Ausführungsform einer Entschlüsselungsmaschine.

- [00106] Fig. 14 zeigt eine bevorzugte Ausführungsform eines Zufallszahlengenerators für eine Entschlüsselungsmaschine.
- [00107] Fig. 15 zeigt eine bevorzugte Ausführungsform einer Transformationseinheit für eine Entschlüsselungsmaschine.
- [00108] Fig. 16 zeigt die Initialisierung einer Entschlüsselungsmaschine.

VERSCHLÜSSELUNG

[00109] In Fig. 1 ist eine Verschlüsselungsmaschine 200 mit einem Datenspeicher gemäß einem ersten Ausführungsbeispiel der Erfindung dargestellt. Dieser Datenspeicher 211 wird vor der eigentlichen Verschlüsselung mit den zu verschlüsselnden Rohdaten befüllt. Bei dem vorliegenden Ausführungsbeispiel umfasst der Datenspeicher eine Anzahl von $N=4096$ Bytes zu je 8 Bit.

[00110] Die Verschlüsselungsmaschine 200 umfasst einen Pseudo-Zufallszahlengenerator 212, im Folgenden auch kurz nur Zufallszahlengenerator bezeichnet, zur Erzeugung von Pseudo-Zufallszahlen, im Folgenden als Zufallszahlen μ bzw. ρ bezeichnet. Der Zufallszahlengenerator 212 verfügt über einen Zustandsspeicher sowie einen externen Eingang, mit dem der Zustandsspeicher gefüllt, initialisiert bzw. geändert werden kann. Weiters verfügt der Zufallszahlengenerator 212 über einen Ausgang, an dem er auf Anfrage Zufallszahlen μ bzw. ρ abgibt. Unter Zufallszahlen μ bzw. ρ wird im Zusammenhang mit der Erfindung eine Folge von Zahlen verstanden, die grundsätzlich vollständig deterministisch aus dem Zustand des Zufallszahlengenerators 212 und den allenfalls dem Zufallszahlengenerator 212 zugeführten Werten abgeleitet werden kann, aber idealer Weise jemandem, dem die Werte des Zustandsspeichers des Zufallszahlengenerators 212 nicht bekannt sind, keine oder zu wenig Hinweise liefert, um kryptographisch verwertbare Vorhersagen für die nächsten zu erwartenden Zahlen zu erlauben.

[00111] Die Verschlüsselungsmaschine 200 umfasst ferner eine Transformationseinheit 213. Die Transformationseinheit 213 wird von einer nicht dargestellten Steuereinheit angesteuert, die die entsprechenden zu verschlüsselnden Bytes aus dem Datenspeicher 211 an der betreffenden Stelle i , ausgewählt durch die Indexeinheit 216, entnimmt und nach der Verschlüsselung wieder an die betreffende Stelle i im Datenspeicher 211 enträgt. Die Transformationseinheit 213 verfügt über einen Eingang, an dem jeweils ein Byte der im Datenspeicher 211 abgespeicherten Daten zur Verschlüsselung anliegt. Am Ausgang der Transformationseinheit 213 liegt nach der Durchführung der Verschlüsselung jeweils das verschlüsselte Byte an. In dem vorliegenden Ausführungsbeispiel der Erfindung wird jeweils ein Byte des Datenspeichers 211 aus diesem ausgelesen und zur Verschlüsselung an die Transformationseinheit 213 übermittelt. Anschließend wird das verschlüsselte Byte an diejenige Stelle des Datenspeichers 211 geschrieben, aus dem das Rohdaten-Byte ausgelesen wurde.

[00112] Die Transformationseinheit 213 verfügt über eine Anzahl A von vorgegebenen eindeutigen Grund-Transformationsvorschriften $f_1, \dots, f_A$. Diese sind in einem dafür vorgesehenen Speicher abgelegt. Eine bevorzugte Weise der Abspeicherung einer der Grund-Transformationsvorschriften $f_1, \dots, f_A$ für Bytes mit einer Bitanzahl von B besteht darin, einen Speicher mit 2^B Bytes zur Verfügung zu stellen, wobei für jeden der möglichen 2^B mittels eines Bytes darstellbaren Werte jeweils ein meist anderer Wert angegeben wird. Um die Eindeutigkeit und damit die Umkehrbarkeit der Grund-Transformationsvorschriften $f_1, \dots, f_A$ zu gewährleisten, wird jeder Wert, entsprechend einer Zahl zwischen 0 und 2^B-1 , jeweils nur ein einziges Mal in einen der 2^B Speicher geschrieben. In dem vorliegenden Ausführungsbeispiel mit $B=8$ Bits pro Byte wird für jede der Grund-Transformationsvorschriften $f_1, \dots, f_A$ jeweils ein Speicher von 256 Bytes zur Verfügung gestellt. Die Anwendung der Grund-Transformationsvorschriften $f_1, \dots, f_A$ auf ein zu transformierendes Byte mit 8 Bits, erfolgt derart, dass als Ergebnis dieses Byte zurückgegeben wird, welches sich an dieser Speicherposition befindet, deren Ordinalnummer, beginnend mit 0, dem als Zahl verstandenen, d.h. zwischen 0 und $255 = 2^8-1$, zu transformierenden Byte entspricht.

[00113] Prinzipiell besteht die Möglichkeit, Transformationen von Bytes ausschließlich mit den bereitgestellten Grund-Transformationsvorschriften $f_1, \dots, f_A$ vorzunehmen. Dies hat jedoch den Nachteil, dass die Anzahl der möglichen Transformationen relativ klein ist. So lassen sich bei einem zur Verfügung stehenden Speicher von 1 MB etwa $A=4000$ Grund-Transformationsvorschriften $f_1, \dots, f_A$ abspeichern. Der Raum an insgesamt zur Verfügung stehenden Transformationsvorschriften zur Transformation von Bytes mit 8 Bits auf ebenso große Bytes beträgt jedoch $(2^8)! \sim 10^{506}$ Transformationen. Die Grund-Transformationsvorschriften $f_1, \dots, f_A$ stellen also nur einen extrem kleinen Teilraum aus dem gesamten Raum der möglichen Transformationsvorschriften dar.

[00114] Aus diesem Grund verwendet das vorliegende Ausführungsbeispiel der Erfindung Transformationsvorschriften $g_1, \dots$, die aus mehreren Vorschriften, ausgewählt aus den Grund-Transformationsvorschriften $f_1, \dots, f_A$, zusammengesetzt sind. Eine Transformationsvorschrift $g \in \{g_1, \dots, g_T\}$ kann, beispielsweise als Folge von T Zufallszahlen $\mu_1 - \mu_T$ vorgegeben oder durch eine Anzahl von mittels einer später beschriebenen Auswahlereinheit ermittelten Zahlenfolge erstellt werden, indem für jede Zahl der Zahlenfolge jeweils eine der Grund-Transformationsvorschriften $f_{\mu_1}, \dots, f_{\mu_T}$ ausgewählt wird und die derart ausgewählten Grund-Transformationsvorschriften $f_{\mu_1}, \dots, f_{\mu_T}$ hintereinander in der Reihenfolge der jeweiligen sie bestimmenden Zahlen auf das zu verschlüsselnde Rohdaten-Byte b angewendet werden.

$$c = g(b) = f_{\mu_1}(f_{\mu_2}(\dots f_{\mu_T}(b) \dots))$$

[00115] Um eine für einen Angreifer schwer zu entschlüsselnde Verschlüsselung bereitzustellen, ist es von großem Vorteil, die vorliegende Menge bzw. Gruppe der Transformationsvorschriften möglichst voll zu nutzen. Durch die Verkettung einer Anzahl T der vorliegenden A Grund-Transformationsvorschriften $f_1, \dots, f_A$ steht eine Anzahl von A^T Transformationsvorschriften $g_1, \dots$ zur Verfügung. Werden Transformationsvorschriften g durch Verkettung von $P=10$ Grund-Transformationsvorschriften f erstellt und stehen bei 1MB Speicherplatz 4000 Grund-Transformationsvorschriften f zur Verfügung, so können insgesamt $4000^{10} (\sim 10^{36})$ unterschiedliche Transformationsvorschriften g herangezogen werden, wobei der Aufwand der Verschlüsselung gegenüber der bloßen Verwendung der Grund-Transformationsvorschriften $f_1, \dots, f_A$ nur T -fach bzw. zehnfach höher ist. Durch die Auswahl der Anzahl A der Grund-Transformationsvorschriften $f_1, \dots, f_A$ sowie der Anzahl T der für eine Verkettung verwendeten Grund-Transformationsvorschriften f kann insgesamt eine Abwägung zwischen Speicherbedarf, Rechenzeit und Sicherheit gefunden werden.

[00116] Die Ausführung der Grund-Transformationsvorschrift f kann auf unterschiedlichste Art implementiert sein, wobei lediglich die Bijektivität der Grund-Transformationsvorschriften f zu gewährleisten ist. Eine Grund-Transformationsvorschrift f kann beispielsweise derart realisiert werden, dass für jeden möglichen Wert eines Bytes jeweils ein in einem dem Wert zugeordnetem Speicher abgespeicherter Transformationswert zur Verfügung steht. Bei einer Bytlänge von 8 Bit stehen insgesamt $256 = 2^8$ verschiedene mögliche Werte zur Verfügung, sodass für Grund-Transformationsvorschriften f für Bytes mit 8 Bit ein Speicher mit 256 Bytes zu 8 Bit zur Verfügung gestellt wird, in dem jeder der Werte jeweils einmal vorkommt. Die Anwendung der Grund-Transformationsvorschrift f auf einen Wert x erfolgt, indem auf den Speicher bzw. auf dessen Speicherposition zugegriffen wird und der im Speicher an dieser Stelle befindlichen Wert als Resultat $f(x)$ der Anwendung der Grund-Transformationsvorschrift auf den Wert x angesehen wird. Einen Speicher für die bei der Entschlüsselung benötigte Umkehrung oder Inverse f^{-1} der Grund-Transformationsvorschrift f kann auf dieselbe Weise realisiert werden.

[00117] Eine alternative Variante zur Anwendung einer Grund-Transformationsvorschrift f auf einen Byte-Wert x besteht darin, den Byte-Wert x mit einer vorgegebenen Maske M_f einer bitweise durchgeführten Exklusiven-Oder-Verknüpfung (XOR-Verknüpfung) zu unterziehen. Hierbei ist lediglich die Maske M_f abzuspeichern, die jeweils dieselbe Länge aufweist wie das jeweilige Byte.

[00118] Die Umkehrung f^{-1} einer derartigen Grund-Transformationsvorschrift ist ident der Grund-Transformationsvorschrift ($f^{-1}=f$).

[00119] Eine weitere Alternative, vorteilhaft bei Bytes bestehend aus insbesondere mehr als 8 Bits, besteht in der Vertauschung einzelner Bits bei der Anwendung der Grund- Transformationsvorschrift. Hierbei ist jeweils anzugeben, welche der Bits miteinander zu vertauschen sind. Die Umkehrung f^{-1} einer derartigen Grund-Transformationsvorschrift f ist ident der Grund-Transformationsvorschrift ($f^{-1}=f$).

[00120] Schließlich kann, insbesondere bei Bytes bestehend aus vielen Bits vorgesehen sein, dass Grund-Transformationsvorschriften f nicht auf alle Bits des Byte angewendet werden, um Speicherplatz zu sparen. In diesem Fall können auch die vorstehend beschriebenen Arten von Grund-Transformationsvorschriften f jeweils auf einer angegebenen Teilbereich des Bytes angewendet werden.

[00121] Sämtliche der dargestellten Grund-Transformationsvorschriften können aufgrund ihrer Bijektivität untereinander kombiniert werden, um eine Transformationsvorschrift g zu erreichen.

[00122] Für eine verbesserte Verschlüsselung kann die Transformationseinheit 213 vor oder nach der Anwendung der Grund-Transformationsvorschriften $f_1..f_A$ zusätzlich eine Spezial-Transformationsvorschrift h auf das jeweilige zu verschlüsselnde Byte b_i anwenden.

[00123] Die Spezial-Transformationsvorschrift h wird bei jeder Verschlüsselung oder nach einer Anzahl von Verschlüsselungen abgeändert. Insbesondere ist die Spezial-Transformationsvorschrift h durch vollständige Angabe aller 2^B Werte in einem Speicher mit 2^B Speicherplätzen analog zu den Grund-Transformationsvorschriften f festgelegt.

[00124] In Abhängigkeit vom zu verschlüsselnden Byte b_i , dem verschlüsselten Byte c_i und gegebenenfalls Zufallswerte p_i können zur Abänderung der Spezial-Transformationsvorschrift je zwei Werte miteinander vertauscht werden. Die Spezial-Transformationsvorschrift kann z.B. durch Vertauschen von zwei Funktionswerten auf je zwei vorgegebenen Positionen vorgenommen werden. Die erste Position kann durch einen Zählerwert eines fortlaufenden zyklischen Zählers, die zweite Position durch das zu verschlüsselnde Byte b_i , das verschlüsselte Byte c_i oder eine Zufallszahl p_i festgelegt werden. Die Verwendung der Spezial- Transformationsvorschrift h während der Durchführung der Kodierung erhöht die Anzahl der potentiell verwendeten Transformationsvorschriften g auf $(2^B)! = (2^8)! \approx 10^{506}$, ohne jedoch die Verarbeitungszeit wesentlich zu erhöhen.

[00125] Eine Spezial-Transformationsvorschrift h unterscheidet sich nicht prinzipiell von einer Grund- Transformationsvorschrift. Der Unterschied liegt nur darin, dass diese, sofern ihre Anwendung vorgesehen ist, zusätzlich zu den normalen Grund-Transformationsvorschriften angewendet werden kann. Ihre Anwendung wird durch bestimmte Ereignisse ein und ausgeschaltet. So ein Ereignis kann zum Beispiel der Wechsel zu einer Vorwärts- oder Rückwärtsbewegung der Kodierung (siehe Fig. 6, Fig. 8) sein, oder der Beginn der Verarbeitung eines neuen Blockes (siehe Fig. 4, Fig. 7) sein.

[00126] Es kann auch vorgesehen sein, dass die Spezial-Transformationsvorschrift h nur eine beschränkte Anzahl oft modifiziert wird.

[00127] Die Abänderung der Spezial-Transformationsvorschrift erfolgt nach Kodierung jedes oder nur bei eines Teiles der zu verschlüsselnden Bytes und kann nach einer Anzahl von Abänderungen überhaupt unterbleiben. Die Spezial-Transformationsvorschrift h bildet einen Teil des inneren Zustandes Y der Transformationseinheit.

[00128] Der Zufallszahlengenerator 212 und die Transformationseinheit 213 bieten die Möglichkeit, ihren jeweiligen momentanen inneren Zustand Y abspeichern und auszugeben und diesen zu einem beliebigen späteren Zeitpunkt auf Kommando von außen wiederherzustellen oder zu laden. Mehrere solche innere Zustände können abgespeichert und wiederhergestellt werden.

[00129] Der innere Zustand des Zufallszahlengenerators 212 und der Transformationseinheit 213 kann durch von außen vorgegebene Werte modifiziert werden.

[00130] Beim ersten Ausführungsbeispiel ("STREAM") eines erfindungsgemäßen Verschlüsse-

lungsverfahrens werden die folgenden Schritte mit der in in Fig. 1 dargestellten Verschlüsselungsmaschine 200 vorgenommen.

[00131] In einem ersten Schritt wird die Verschlüsselungsmaschine 200 mit dem Passwort W_0 initialisiert. Hierbei werden sämtliche internen Speicher des Zufallszahlengenerators 212 sowie der Transformationseinheit 213 in einen durch das Passwort W_0 bestimmten Initialzustand Y_0 gebracht. Eine bevorzugte Art der Initialisierung ist später mit Bezug auf Fig. 12 näher dargestellt.

[00132] In einem zweiten Schritt wird der Datenspeicher 211 mit Rohdaten befüllt. Diejenige Menge an Rohdaten $b_1, \dots, b_N$, die gleichzeitig im Datenspeicher 211 zur Verfügung steht, wird auch als Block oder Datenblock D bezeichnet. Ein Datenzeiger i, mit dem jeweils ein Rohdaten-Byte b_i aus dem Datenspeicher 211 ausgewählt wird, wird auf den ersten Speicherplatz des Datenspeichers 211 gesetzt.

[00133] Die folgenden Iterationsschritte $S_{21} \dots S_{29}$ werden für sämtliche Bytes des Datenspeichers 211, wie in Fig. 2 und 3 dargestellt, ausgeführt. Der im Speicher 216 abgespeicherte Datenzeiger i zeigt dabei anfangs auf das erste Byte b_1 des zu verschlüsselnden Datenblocks D und wird nach Durchführung der Iterationsschritte $S_{21} \dots S_{29}$ so lange inkrementiert, bis er auf das letzte Byte b_N des zu verschlüsselnden Datenblocks D zeigt.

[00134] In einem ersten Iterationsschritt S_{21} wird das Byte des Datenblockes D des jeweiligen vom Wert des Datenzeigers i ausgewählten Speicherplatzes des Datenspeichers 211 in einen Rohdaten- Zwischenspeicher 214 als Rohdaten-Byte b_i kopiert und der Transformationseinheit 213 zugeführt.

[00135] In einem zweiten Iterationsschritt S_{22} wird eine Folge von Zufallszahlen $\mu_{i,1}, \dots, \mu_{i,P}$ an die Transformationseinheit 213 übermittelt. Für jede Iteration werden separate Zufallszahlen $\mu_{i,1}, \dots, \mu_{i,P}$ ermittelt.

[00136] Die Transformationseinheit 213 ermittelt im dritten Iterationsschritt S_{23} die Transformationsvorschrift g_i in Abhängigkeit von ihrem internen Zustand und den ermittelten Zufallszahlen $\mu_{i,1}, \dots, \mu_{i,P}$.

[00137] Das verschlüsselte Byte c_i wird durch Anwendung der Transformationsvorschrift g_i auf das zu verschlüsselnde Byte b_i in einem vierten Iterationsschritt S_{24} gemäß $c_i = g_i(b_i)$ ermittelt. Im vorliegenden Fall wird die Transformationsvorschrift g_i durch Zusammensetzung, wie vorstehend beschrieben ermittelt, wobei die Auswahl von T Vorschriften aus den A Grund- Transformationsvorschriften $f_1, \dots, f_A$ für die Zusammensetzung aufgrund der angegebenen Zufallszahlen $\mu_{i,1}, \dots, \mu_{i,P}$ erfolgt.

[00138] In einem fünften Iterationsschritt S_{25} wird der Inhalt des jeweiligen vom Datenzeiger i ausgewählten Speicherplatzes des Datenspeichers 211 mit dem verschlüsselten Byte c_i überschrieben. Anschließend wird in einem sechsten Iterationsschritt S_{26} das verschlüsselte Byte c_i in einen eigenen Schlüsseldaten-Zwischenspeicher 217 als verschlüsseltes Byte c_i geschrieben.

[00139] In einem siebenten Iterationsschritt S_{27} werden das Rohdaten-Bytes b_i und das verschlüsselte Byte c_i an den Zufallszahlengenerator 212 und die Transformationseinheit 213 für die künftige Zustandsänderungen dieser übermittelt.

[00140] In einem achten Iterationsschritt S_{28} werden weitere vom Zufallszahlengenerator 212 erzeugte Zufallszahlen $\rho_{i,1} \dots \rho_{i,R}$ an den Zufallszahlengenerator 212 zurück und an die Transformationseinheit 213 für die künftige Zustandsänderungen übermittelt.

[00141] In einem neunten Iterationsschritt S_{29} wird der innere Zustand des Zufallszahlengenerators 212 und der Transformationseinheit 213 mit Hilfe der in den Iterationsschritten S_{27} und S_{28} übermittelten Bytes bzw Zufallszahlen abgeändert.

[00142] Insgesamt ist es beim Verschlüsseln möglich aber nicht zwingend erforderlich, bei der Durchführung des neunten Iterationsschritts S_{29} auch die Grund-Transformationsvorschriften $f_1,$

..., f_A zu ändern, beispielsweise kann bei jeder Durchführung des neunten Iterationsschritts S_{29} jeweils eine der Grund-Transformationsvorschriften f_x abgeändert werden, beispielsweise bei Grund-Transformationsvorschriften, die durch vollständige Aufzählung ihrer Funktionswerte vorgegeben wurden, durch Vertauschen zweier Funktionswerte, i.e. $f_x(b_y) \leftrightarrow f_x(b_z)$. Dadurch wird die Anzahl der möglichen Transformationsvorschriften von ca 10^{36} nochmals auf ca 10^{506} verschiedene Transformationsvorschriften vergrößert, und macht so eine unbefugte Entschlüsselung weniger wahrscheinlich.

[00143] Sofern vorgesehen, kann die Abänderung der Spezial-Transformationsvorschrift h im Rahmen des neunten Iterationsschritts S_{29} vorgenommen werden.

[00144] Je nach Fortschritt der Verschlüsselung entstehen bei der Transformationseinheit 213 sowie beim Zufallszahlengenerator 212 in Abhängigkeit von sowohl den bisherig verarbeiteten Rohdatenbytes b_i , den verschlüsselten Bytes c_i und den vom Zufallszahlengenerator 12 gelieferten Zufallszahlen $p_{i,1} \dots p_{i,R}$ unterschiedliche, innere Zustände.

[00145] Die Auswahl der Transformationsvorschrift g_i ist daher sowohl direkt abhängig von dem inneren Zustand der Transformationseinheit 213 zum Zeitpunkt der Verschlüsselung des Rohdatenbytes b_i als auch indirekt durch die von dem Zufallszahlengenerator 212 zum gleichen Zeitpunkt bereitgestellten Zufallszahlen $p_{i,1} \dots p_{i,R}$, $\mu_{i,1}, \dots, \mu_{i,P}$, die wiederum auch von dem inneren Zustand des Zufallszahlengenerators 212 in Abhängigkeit von den bis dahin verarbeiteten Rohdatenbytes und verschlüsselten Bytes unterschiedlich erzeugt werden.

[00146] Die Auswahl der Transformationsvorschrift g_i ist daher abhängig von den Werten und der Reihenfolge aller b_1 bis b_{i-1} . Nach der Verschlüsselung wird der Wert des Zeigers i inkrementiert.

[00147] Nachdem die im Datenspeicher 211 abgelegten Rohdaten-Bytes $b_1, \dots b_N$ allesamt verschlüsselt sind, können diese für die weitere Verwendung, insbesondere für die Übertragung an einen Empfänger, zur Verfügung gehalten werden. Nach der allfälligen weiteren Verarbeitung der Daten im Datenspeicher 211 wird dieser eventuell mit weiteren zu verschlüsselnden Daten wieder aufgefüllt, die analog wie oben verschlüsselt werden.

[00148] Es besteht die Möglichkeit, den Datenspeicher 211 vor der Verarbeitung mit Zufallsdaten zu ergänzen bzw zu erweitern.

[00149] Der innere Zustand sowohl des Zufallszahlengenerators 212 als auch der Transformationseinheit 213 ist am Beginn der neuerlichen Verarbeitung von weiteren Daten im Datenspeicher 211 gleich wie am Ende der vorherigen Verarbeitung der Daten des Datenspeichers 211.

[00150] Ein Vorgehen gemäß einem zweiten Ausführungsbeispiel ("BLOCK"), dargestellt in Fig. 4 und 5, der Erfindung zeigt eine Verbesserung, mit der die Abhängigkeiten der einzelnen Bytes $c_1, \dots c_N$ der verschlüsselten Datenfolge von den Bytes $b_1, \dots b_N$ der Rohdaten-Datenfolge erhöht wird. Bei diesem Vorgehen wird das vorstehend dargestellte Verfahren gemäß dem ersten Ausführungsbeispiel der Erfindung verwendet. Die für die Durchführung des Verschlüsselungsverfahrens verwendete Verschlüsselungsmaschine entspricht der Verschlüsselungsmaschine 200 des ersten Ausführungsbeispiels, das in Fig. 1 näher dargestellt ist, wobei die Unterschiede im Folgenden näher erläutert werden.

[00151] Vor dem Beginn der Iterationsschritte $S_{21} \dots S_{29}$ wird in einem Vorbereitungsschritt $S_{20,A}$ der gesamte innere Zustand der Verschlüsselungsmaschine 200, d.h. der innere Zustand des Zufallszahlengenerators 212, sowie gegebenenfalls der innere Zustand der Transformationseinheit 213, in einem Zustandsspeicher abgespeichert. Dieser Zustand wird als Initialzustand Y_0 bezeichnet. Anschließend wird die Verschlüsselung vom ersten bis zum letzten Byte gemäß den Iterationsschritten S_{21} bis S_{29} durchgeführt. Nach der Verschlüsselung des gesamten Blocks, d.h. aller im Datenspeicher 211 befindlichen Rohdaten $b_1 \dots b_N$, wird in einem Rücksetzschrift $S_{20,M}$ der Zufallszahlengenerator 212 sowie gegebenenfalls die Transformationseinheit 213 auf ihren jeweiligen Initialzustand Y_0 zurückgesetzt.

[00152] Daran anschließend wird eine Verschlüsselung gemäß den vorstehend beschriebenen

Iterationsschritten $S_{21} \dots S_{29}$ erneut durchgeführt, wobei der Wert des Datenzeigers i anfangs auf die letzte Position N des Datenspeichers 211 zeigt und entgegen der zuvor durchlaufenen Richtung geführt wird, d.h. in diesem Fall dekrementiert wird, und dadurch die einzelnen Bytes des Datenspeichers 211 in umgekehrter Reihenfolge behandelt werden.

[00153] Anschließend werden die Daten in einem Ausleseschritt $S_{20,Z}$ aus dem Speicher gelesen und gegebenenfalls durch einen neuen Datenblock ersetzt.

[00154] Aufgrund dieses Vorgehens ist gewährleistet, dass der Wert jedes verschlüsselten Bytes c_i vom Wert und Position sämtlicher Rohdaten-Bytes b_i abhängig ist und eine Änderung eines beliebigen Rohdaten-Bytes b_i Auswirkungen auf die Verschlüsselung aller anderen Bytes des Datenspeichers 211 hat. Jedes zu verschlüsselnde Rohdaten-Byte b_i wird mit dem Vorgehen darüber hinaus zweifach verschlüsselt.

[00155] Die Verschlüsselung der im Datenspeicher 211 befindlichen Rohdaten kann gegebenenfalls auch mehrfach, wie in Fig. 6 dargestellt, vorgenommen werden, wobei jeweils abwechselnd in der vorgegebenen Datenrichtung und anschließend entgegen der vorgegebenen Datenrichtung verschlüsselt wird. In einem solchen Fall wird die Verschlüsselungsmaschine 200 jeweils nach jedem Verschlüsselungsdurchlauf, d.h. nachdem das letzte Byte b_1 oder b_N in oder entgegen der Datenrichtung verschlüsselt wurde, in den vorab abgespeicherten inneren Zustand Y_0 gesetzt.

[00156] Nach der oben beschriebenen Verarbeitung der Daten des Datenspeichers 211 und einer allfälligen weiteren Verwendung der Daten, werden der Zufallszahlengenerator 212 und die Transformationseinheit 213 wieder in den jeweiligen Initialzustand Y_0 gesetzt. Weitere eventuell verbliebene Rohdaten werden in den Datenspeicher 211 übertragen und die Verarbeitung mit den neuen Rohdaten beginnt von neuem. Es besteht die Möglichkeit, den Datenspeicher 211 vor der Verarbeitung mit Zufallsdaten zu ergänzen bzw zu erweitern.

[00157] Bei einem dritten Ausführungsbeispiel ("CHAINED-BLOCK") der Erfindung, dargestellt in Fig. 7 und 8, wird der Datenspeicher 211 bei der Verarbeitung der erstmals geladenen Daten wie bei dem zweiten Ausführungsbeispiel verarbeitet. Die Verarbeitung von verbliebenen und neu in den Datenspeicher 211 geladenen Daten unterscheidet sich von der Verarbeitung gemäß dem zweiten Ausführungsbeispiel dadurch, dass für den Zufallszahlengenerator 212 und die Transformationseinheit 213 die jeweiligen Initialzustände in den Verarbeitungsschritten $S_{20,B}$, $S_{20,C}$, $S_{20,Y}$ unterschiedlich zu dem zweiten Ausführungsbeispiel festgelegt werden. Bei diesen neuen Initialzuständen handelt es sich jeweils um die bei der vorhergehenden Verarbeitung von Daten erreichten inneren Zustände $Y_1, Y_2, \dots$ der Verschlüsselungsmaschine 200.

[00158] Diese neuen Initialzustände $Y_1, Y_2, \dots$ (Fig. 8) sind die inneren Zustände von Zufallszahlengenerator 212 und Transformationseinheit 213 nach Verarbeitung des Bytes an der letzten Stelle N im Datenspeicher 211 bei einer der Vorwärtsverarbeitungen der vorhergehenden Daten. Alternativ dazu kann auch der innere Zustand nach Verarbeitung des Bytes, an der Stelle 1 im Datenspeicher 211 bei einer der Rückwärtsverarbeitungen dafür verwendet werden.

[00159] Für eine Echtzeit-Datenübertragung weisen der Datenspeicher 211 und die Datenblöcke nur eine sehr geringe Anzahl von Rohdaten-Bytes b_i , eventuell sogar nur ein einziges Rohdaten-Byte, auf. Eine Verschlüsselung erfolgt bytewise, wobei nach der Verschlüsselung eines Datenblocks D - und damit des gesamten Datenspeichers 211 - ein neuer Datenblock D in den Datenspeicher 211 geladen wird. Eine solche Vorgehensweise eignet sich besonders gut zur laufenden Verschlüsselung von Daten, die als Datenstrom vorliegen und die gleichsam in Echtzeit weitergeleitet werden. Allenfalls besteht auch die Möglichkeit, wie für das zweite Ausführungsbeispiel beschrieben, den Datenstrom in sehr kleine Blöcke D mit nur wenigen, insbesondere weniger als 10 Bytes zu unterteilen und diese Blöcke gemäß dem zweiten oder dritten Ausführungsbeispiel der Erfindung zu verschlüsseln und als Block-Datenstrom zu übertragen. Sofern die Anzahl der Bytes pro Datenblock gering gehalten werden kann, kann die Verschlüsselung auch zur Echtzeit-Datenübertragung verwendet werden. Hierbei ermöglicht es die Erfindung durch Auswahl der Größe der Datenblöcke, eine Abwägung zwischen Echtzeitfähigkeit

bzw. Verzögerung und Sicherheit zu treffen.

ZUFALLZAHLENSGENERATOR UND AUSWAHLEINHEIT

[00160] Grundsätzlich können für die Erfindung sehr unterschiedliche deterministische Zufallszahlengeneratoren 212 verwendet werden. Eine Verbesserung der Erfindung, die für alle vorstehend dargestellten Ausführungsbeispiele der Erfindung verwendet werden kann, verwendet einen in Fig. 10 dargestellten optimierten Zufallszahlengenerator 212 mit einer Auswahlinheit 230.

[00161] Im vorliegenden Fall ist der Zufallszahlengenerator 212 aus einer Anzahl Z von einfachen, deterministischen Teil-Zufallszahlengeneratoren 233-1 bis 233- Z zusammengesetzt, die - wie später noch beschrieben wird - abhängig vom inneren Zustand der Auswahlinheit 230 des Zufallszahlengenerators 212 ausgewählt werden. Die Anzahl Z der Teil-Zufallszahlengeneratoren 233 ist vorab nicht vorgegeben und kann aufgrund eines Passwortes W bestimmt werden. Darüber hinaus ist es auch möglich, unterschiedliche Typen von Teil-Zufallszahlengeneratoren 233 vorab vorzugeben. Weiters kann auch vorgegeben werden, mit welcher Wahrscheinlichkeit die einzelnen Teil-Zufallszahlengeneratoren 233 aus den gegebenen Typen von Teil-Zufallszahlengeneratoren ausgewählt werden. Das Passwort W bestimmt dann mit Hilfe eines bereits vorhandenen Zufallszahlengenerators 212 die Anzahl Z der Teil-Zufallszahlengeneratoren 233, die Typen der Teil-Zufallszahlengeneratoren 232 und die Reihenfolge der Teilzufallszahlen 232 in der diese für den Aufbau des Zufallszahlengenerators 212 verwendet werden.

[00162] Die Initialisierung der Teil-Zufallszahlengeneratoren 233 erfolgt ebenso durch das Passwort W im Zusammenhang mit dem vorgegebenem Zufallszahlengenerator und wird später näher dargestellt.

[00163] Der Zufallszahlengenerator 212 weist eine Auswahlinheit 230 auf. Die Auswahlinheit 230 des Zufallszahlengenerators 212 dient zur Auswahl eines der Teil-Zufallszahlengeneratoren 233-1 bis 233- Z . Dessen Werte werden dann am Ausgang des Zufallszahlengenerators 212 bereitgehalten. Die Auswahlinheit 230 verfügt über eine Anzahl H von Auswahlregistern 231-1...231- H mit jeweils einer vorgegebenen Anzahl Z von Einzelregistern. Die Gesamtheit der Einzelregister eines Auswahlregisters 231-1...231- H enthält eine Permutation der Zahlen 1 bis Z .

[00164] Dem ersten der Auswahlregister 231-1 ist eine erste Adressiereinheit 232-1 vorgeschaltet, die einen ersten Zeiger k_1 enthält. Dieser erste Zeiger k_1 wird nach jeder Auswahl automatisch zyklisch inkrementiert oder dekrementiert. Der erste Zeiger k_1 bezeichnet bzw. referenziell eine Speicherposition im ersten Auswahlregister 231-1.

[00165] In einem weiteren Schritt wird die an der vom ersten Zeiger k_1 ausgewählte Speicherposition im ersten Auswahlregister 231-1 befindliche Zahl θ_1 ermittelt. Diese Zahl θ_1 wird in eine zweite Adressiereinheit 232-2 als zweiter Zeiger k_2 eingetragen oder zu diesem zweiten Zeiger k_2 hinzugezählt oder der zweite Zeiger k_2 wird durch eine sonstige Operation auf Grundlage der Zahl θ_1 und seinem vorherigen Wert ermittelt. Der Wert des zweiten Zeigers k_2 , der stets im Bereich von 1 bis Z liegt, wird zur Adressierung einer Speicherposition k_2 im zweiten Auswahlregister 231-2 herangezogen.

[00166] Je ein Auswahlregister 231 und je eine Auswahlinheit 232 werden jeweils zu einer Auswahlstufe 230-1...230- H zusammengefasst. Analog zur Auswahlstufe 230-1 werden auch die folgenden Auswahlstufen bis zur Auswahlstufe 230- H durchlaufen. Der Registerwert θ_H der letzten Auswahlstufe 230- H wählt dann den Teil-Zufallszahlengenerator 233- θ_H aus.

[00167] Grundsätzlich kann eine große Anzahl an Auswahlstufen 230-1...230- H in der Auswahlinheit 230 vorgesehen werden, wobei eine größere Anzahl von Auswahlstufen 230-1...230- H zu einer stärkeren Ressourcenbelastung führt und letztlich in einer erhöhten Datensicherheit resultiert, jedoch die Ausführungszeit der Verschlüsselung erhöht. Durch die Auswahl der Anzahl der Auswahlstufen 230-1...230- H kann somit eine Abwägung zwischen der Rechenleistung und Datensicherheit getroffen werden.

[00168] Im vorliegenden Fall sind insgesamt zehn $H=10$ Auswahlstufen 230-1...230-H vorgesehen. Für die Auswahlregister 231 und Adressiereinheiten 232 wird jeweils die an der vom jeweiligen Zeiger $k_1...k_H$ in der Adressiereinheit 231 ausgewählten Speicherposition im jeweiligen Auswahlregister 231 befindliche Zahl ermittelt und in den jeweils nächsten Zeiger k in der nächsten Adressiereinheit 231 eingetragen oder zu diesem hinzugezählt oder durch eine sonstige Operation mit den vorherigen Wert ermittelt. Der Wert des jeweils nächsten Zeigers k , dieser liegt stets im Bereich von 1 bis Z , wird zur Adressierung einer Speicherposition im jeweils nächsten Auswahlregister 231 herangezogen. Der vom jeweiligen Zeiger adressierte Wert θ_H des letzten Auswahlregisters 231-H wird zur Auswahl eines der Teil-Zufallszahlengeneratoren 233-1...233-Z herangezogen.

[00169] Die einzelnen Teil-Zufallszahlengeneratoren 233-1 ... 233-Z haben einen gemeinsamen inneren Zustand, wobei im Normalfall jeder der Teil-Zufallszahlengeneratoren 233-1 ... 233-Z jeweils über einen ihm zugewiesenen Teilbereich des inneren Zustandes verfügt. Es ist auch möglich, dass einzelne der Teil-Zufallszahlengeneratoren 233-1...233-Z auf die Teilbereiche der übrigen Teil-Zufallszahlengeneratoren 233-1...233-Z Lesezugriff oder gegebenenfalls auch Schreibzugriff haben.

TRANSFORMATIONSEINHEIT

[00170] Eine analog zu 320 aufgebaute Auswahleinheit 220 dient zur Auswahl einer Grundtransformation f aus den gegebenen Grundtransformationen f_1 bis f_A .

[00171] Fig. 11 zeigt die Auswahleinheit 220 zur Auswahl von Grundtransformationsvorschriften f . Für die Zusammensetzung der Transformationsvorschrift g ist eine Anzahl von A Grundtransformationsvorschriften $f_1...f_A$ vorgegeben, die - wie später noch beschrieben wird - abhängig vom inneren Zustand des Zufallszahlengenerators 212 gebildet und ausgewählt werden. Die Anzahl der Grund-Transformationsvorschriften $f_1...f_A$ sowie deren Inhalt kann aufgrund des Passwortes W_0 bestimmt werden. Darüber hinaus ist es auch möglich, unterschiedliche Typen von Grund-Transformationsvorschriften $f_1...f_A$ vorab vorzugeben.

[00172] Die Transformationseinheit 213 weist eine Auswahleinheit 220 auf. Die Auswahleinheit 220 dient zur Auswahl einer der Grund-Transformationsvorschriften $f_1...f_A$. Die Auswahleinheit 220 der Transformationseinheit 213 wird in Fig. 11 näher dargestellt. Sie verfügt über eine Anzahl G von Auswahlregistern 221-1...221-G mit jeweils einer vorgegebenen Anzahl A von Einzelregistern. Die Gesamtheit der Einzelregister eines Auswahlregisters 221-1...221-G enthält eine Permutation der Zahlen 1 bis A .

[00173] Dem ersten der Auswahlregister 221-1 ist eine erste Adressiereinheit 222-1 vorgeschaltet, die einen ersten Zeiger j_1 enthält. Dieser erste Zeiger j_1 wird nach jeder Auswahl automatisch inkrementiert oder dekrementiert. Der erste Zeiger j_1 bezeichnet bzw. referenziell eine Speicherposition im ersten Auswahlregister 221-1.

[00174] In einem weiteren Schritt wird die an der vom ersten Zeiger j_1 ausgewählten Speicherposition im ersten Auswahlregister 221-1 befindliche Zahl Φ_1 ermittelt. Diese Zahl Φ_1 wird in eine zweite Adressiereinheit 222-2 als zweiter Zeiger j_2 eingetragen oder zu diesem zweiten Zeiger j_2 hinzugezählt oder der zweite Zeiger j_2 wird durch eine sonstige Operation auf Grundlage der Zahl Φ_1 und seinem vorherigen Wert ermittelt. Der Wert des zweiten Zeigers j_2 , der stets im Bereich von 1 bis A liegt, wird zur Adressierung einer Speicherposition j_2 im zweiten Auswahlregister 221-2 herangezogen. Analog zur Auswahleinheit des Zufallszahlengenerators werden auch hier je ein Auswahlregister 221 und je eine Auswahleinheit 222 jeweils zu einer Auswahlstufe 220-1...220-G zusammengefasst. Analog zur Auswahlstufe 220-1 werden auch die folgenden Auswahlstufen bis zur Auswahlstufe 220-G durchlaufen. Der Registerwert ϕ_G der letzten Auswahlstufe 220-G wählt dann die Grundtransformation ϕ_G aus.

[00175] Grundsätzlich kann eine große Anzahl an Auswahlstufen in der Auswahleinheit 220 vorgesehen werden, wobei eine größere Anzahl von Auswahlstufen 220-1...220-G zu einer stärkeren Ressourcenbelastung führt und letztlich in einer erhöhten Datensicherheit resultiert,

jedoch die Ausführungszeit der Verschlüsselung erhöht. Durch die Auswahl der Anzahl G der Auswahlstufen 220-1...220- G kann somit eine Abwägung zwischen der Rechenleistung und Datensicherheit getroffen werden.

[00176] Im vorliegenden Fall sind insgesamt zehn $G=10$ Auswahlstufen 220-1...220- G vorgesehen. Für die Auswahlregister 221 und Adressiereinheiten 222 wird jeweils die an der vom jeweiligen Zeiger $j_1...j_G$ in der Adressiereinheit 221 ausgewählten Speicherposition im jeweiligen Auswahlregister 221 befindliche Zahl ermittelt und in den jeweils nächsten Zeiger j in der nächsten Adressiereinheit 221 eingetragen oder zu diesem hinzugezählt oder durch eine sonstige arithmetische Operation ermittelt. Der Wert des jeweils nächsten Zeigers j wird zur Adressierung einer Speicherposition im jeweils nächsten Auswahlregister 221 herangezogen. Der vom jeweiligen Zeiger adressierte Wert Φ_G des letzten Auswahlregisters 221- G wird zur Auswahl einer Grund-Transformationsvorschrift $f_1...f_A$ herangezogen. Die Länge des Auswahlregisters 221 entspricht normalerweise der Anzahl der verwendeten Grund-Transformationsvorschriften $f_1...f_A$.

[00177] Die Auswahlereinheit 220, 230 des Zufallszahlengenerators 212 und der Transformationseinheit 213 sind sehr ähnlich aufgebaut. Die folgenden Verbesserungen werden für beide Auswahlereinheiten 220, 230 gemeinsam beschrieben, können aber für jede der Auswahlereinheiten 220, 230 separat vorgenommen werden.

[00178] Durch das vorstehend genannte Vorgehen wird lediglich eine Auswahl einer Grund-Transformationsvorschrift $f_1...f_A$ oder des Teil-Zufallszahlengenerators 233-1...233- Z getätigt, die von den einzelnen zu verschlüsselnden Rohdaten-Bytes b_i vollkommen unabhängig ist und daher eine leichtere Rekonstruktion ermöglicht.

[00179] Die vorliegende Verbesserung der Erfindung sieht nunmehr eine von der Aktualisierungseinheit vorgenommene Aktualisierung der Auswahlereinheit 220, 230 vor, wobei abhängig vom jeweiligen zu verschlüsselnden Byte b_i und/oder vom jeweils verschlüsselten Byte c_i und/oder von bereitgestellten Zufallszahlen $p_1...p_R$ der Wert eines der Zeiger $j_1...j_G, k_1...k_H$ in den Adressiereinheiten 222, 232 abgeändert wird. Hierbei wird der Zahlenwert der Adressiereinheit 222, 232 durch eine arithmetische Funktion aus dem alten Zahlenwert der Adressiereinheit 222, 232 und einem externen, über einen Eingang 224, 234 zur Verfügung gestellten Wert, z.B. dem Zahlenwert b_i des unverschlüsselten Bytes, dem Zahlenwert c_i des verschlüsselten Bytes oder den bereitgestellten Zufallszahlen p_i , neu berechnet. Sofern der sich daraus ergebende Wert die Länge des Auswahlregisters 221, 231 übersteigt, wird diese Länge entsprechend oft vom Zeigerwert abgezogen, dass der Zeiger j, k zwischen 1 und A bzw. Z liegt. Bei jedem Verschlüsselungsschritt wird jeweils der nächste Zeiger j, k abgeändert und für den Fall, dass im vorherigen Schritt der letzte G -te oder H -te Zeiger abgeändert wurde wird im nächsten Schritt wiederum der erste Zeiger j, k abgeändert.

[00180] Alternativ oder zusätzlich ist es möglich, dass abhängig vom jeweiligen zu verschlüsselnden Byte b_i und/oder vom jeweils verschlüsselten Byte c_i und/oder von bereitgestellten Zufallszahlen p zwei Einzelregister eines der Auswahlregister 221, 231 abgeändert werden und zwar durch Vertauschung der in ihnen enthaltenen Werte. Die Auswahl dieser zwei Einzelregister ist auch abhängig von der Anzahl der bereits in diesem Auswahlregister 221, 231 durchgeführten Vertauschungen. Bei jedem Verschlüsselungsschritt wird eine Anzahl von Vertauschung durchgeführt. Nachdem eine bestimmte Anzahl von Vertauschungen in einem Auswahlregister 221, 231 durchgeführt wurden, wird dann zum nächsten Auswahlregister 221, 231 gewechselt.

[00181] Für den Fall, dass im vorherigen Schritt das letzte Auswahlregister 221, 231 abgeändert wurde, wird dann wieder zum ersten Auswahlregister 221, 231 gewechselt.

[00182] Zur Auswahl einer Mehrzahl von Grund-Transformationsvorschriften $f_1, ..., f_A$ oder einer Mehrzahl von Teil-Zufallszahlengeneratoren 233-1...233- Z wird jeweils der erste Zeiger j_1, k_1 nach vorgegebenen Kriterien abgeändert, insbesondere erhöht, verringert, inkrementiert oder dekrementiert, und die vorstehend genannten Schritte zur Adressierung der einzelnen Auswahlregister 221, 231 erneut durchgeführt, wobei jeweils diejenige Grund-Transformationsvorschrift

$f_1, \dots, f_A$ für die Bildung der Transformationsvorschriften oder die Zufallszahl desjenigen Teil-Zufallszahlengenerators 233 herangezogen wird, deren Ordnungszahl dem Zahlenwert des im letzten Auswahlregister 221-G, 231-H referenzierten Wertes entspricht. Die Anzahl der Einzelregister eines Auswahlregisters 221, 231 entspricht der Anzahl A der zur Verfügung stehenden Grund-Transformationsvorschriften $f_1, \dots, f_A$ bzw. der Anzahl Z der zur Verfügung stehenden Teil-Zufallszahlengeneratoren 233.

[00183] Bevorzugt besteht auch die Möglichkeit, dass der Auswahlinheit 230 des Zufallszahlengenerators 212 oder der Auswahlinheit 220 der Transformationseinheit 213 jeweils weitere vom Zufallszahlengenerator 212 erstellte Zufallszahlen p zugeführt werden und dass mittels dieser Zufallszahlen p die Werte der jeweiligen Auswahlregister 221, 231 neu gesetzt bzw. modifiziert oder die Zeiger in den Adressiereinheiten 222, 232 überschrieben bzw. modifiziert werden.

[00184] Bei der Ermittlung von Zufallszahlen p durch den Zufallszahlengenerator 212 kann eine vorab festgelegte Anzahl von diesen erstellten Zufallszahlen herangezogen werden.

[00185] Mit einer solchen Auswahlinheit 220, 230 wird es vermieden, dass eine in einem Schritt vorgenommene Zustandsänderung durch eine in einem noch folgenden Schritt vorgenommene Zustandsänderung rückgängig gemacht wird.

[00186] Auch wird vermieden, dass Klartexte, die einander sehr ähnlich sind und einander nur in wenigen Zeichen unterscheiden, sich der verschlüsselte Text auch nur in diesen wenigen Zeichen unterscheidet.

[00187] Je größer die Anzahl G, H der Auswahlstufen 220, 230 gewählt wird, desto geringer ist die Wahrscheinlichkeit für eine spätere Übereinstimmung des verschlüsselten Textes oder des inneren Zustandes.

[00188] Die einzelnen Auswahlregister 221, 231 werden im Zuge der Initialisierung der Transformationseinheit 213 und des Zufallszahlengenerators 212 auf Werte gesetzt, die vom Passwort W sowie von aus dem Passwort W abgeleiteten Daten abhängen. Auch der Wert der Adressiereinheiten 222, 232 wird bei der Initialisierung auf einen vom Passwort W abgeleiteten Wert gesetzt. Der Inhalt der Auswahlregister 221, 231 sowie der Adressiereinheiten 222, 232 zählt zum inneren Zustand des Zufallszahlengenerators 212 oder der Transformationseinheit 213 und somit zum inneren Zustand der Verschlüsselungsmaschine 200.

[00189] Die Initialisierung der Auswahlinheiten 220, 230 erfolgt separat, sodass die inneren Zustände des Zufallszahlengenerators 212 und der Transformationseinheit 213 unterschiedlich sind.

[00190] Auch ist der Inhalt der Auswahlregister 221, 231 und der Auswahlzeiger 222, 232 grundsätzlich unterschiedlich, d.h. die Auswahlinheiten 220, 230 greifen auf unterschiedliche innere Zustände zu, die jeweils einen Teil des inneren Zustandes der Verschlüsselungsmaschine bilden.

[00191] Auch die Grund-Transformationsvorschriften f_1 , bis f_A können während der Durchführung der Kodierung, insbesondere des Iterationsschrittes S_{29} , bzw. bei der Entschlüsselung können die inversen Grund-Transformationsvorschriften f_1^{-1} , bis f_A^{-1} insbesondere beim Iterationsschritt S_{39} , in Abhängigkeit von b_i , c_i und p modifiziert werden.

[00192] Die Anzahl der möglicherweise verwendeten Transformationen g steigt dadurch auf $(2^8)! \approx 10^{506}$. Dies sind alle möglichen Transformationen von 8-Bit Bytes. Durch diese Maßnahme wird ein unberechtigter Zugriff auf den Klartext ohne Kenntnis des Passwortes und des Klartextes nahezu ausgeschlossen. Durch diese Maßnahme wird die für den neunten Iterationsschritt S_{29} bzw. bei der Entschlüsselung bei Iterationsschritt S_{39} benötigte Zeit bedeutend länger.

[00193] Eine günstige Anwendung einer Spezial-Transformationsvorschrift h für die STREAM-Verarbeitung gemäß dem ersten Ausführungsbeispiel der Erfindung besteht darin, während der Verschlüsselung der ersten 256 Bytes die Spezial-Transformationsvorschrift h zu modifizieren

und danach unverändert zu lassen. Aber dennoch für den Rest der Daten immer aktiv zu belassen.

[00194] Eine günstige Anwendung einer Spezial-Transformationsvorschrift h für die BLOCK Verarbeitung gemäß dem zweiten oder dritten Ausführungsbeispiel der Erfindung besteht darin, bei der ersten Vorwärts-Kodierung die Spezial-Transformationsvorschrift nicht anzuwenden und/oder nicht abzuändern und für die erste Rückwärts-Kodierung die Spezial-Transformationsvorschrift anzuwenden und/oder abzuändern aber eine Modifikation dieser nur für die ersten 256 Verschlüsselungen zu vorzunehmen.

[00195] Eine weitere günstige Anwendung einer Spezial-Transformationsvorschrift h für die BLOCK Verarbeitung gemäß dem zweiten oder dritten Ausführungsbeispiel der Erfindung besteht darin, die bei Kodierung des ersten Datenblockes erstellte Spezialtransformationsvorschrift h bei der Verarbeitung der folgenden Datenblöcke unverändert während der Vorwärts- und/oder Rückwärts-Kodierung einzusetzen.

[00196] Bei der Entschlüsselung wird entsprechend die Inverse h^{-1} der Spezial-Transformationsvorschrift h verwendet und gegebenenfalls auch entsprechend abgeändert.

INITIALISIERUNG

[00197] Im Folgenden wird unter Verweis auf die Fig. 12 eine mögliche vorteilhafte Ausführungsvariante einer Initialisierung des Zufallszahlengenerators 212, einer Transformationseinheit 213 sowie der Verschlüsselungsmaschine 200 beschrieben, wobei die Initialisierung für sämtliche der vorstehend dargestellten Ausführungsbeispiele der Erfindung verwendet werden kann. Grundsätzlich wird die konkrete Art der Initialisierung der Verschlüsselungsmaschine 200 durch das bei der Verschlüsselung verwendete Passwort W_0 bestimmt. Das Passwort W_0 wird an eine Initialisierungseinheit übertragen, die die im Folgenden genannten Schritte durchführt.

[00198] In Fig. 12 ist die Initialisierung der Verschlüsselungsmaschine 200 dargestellt.

[00199] In einem ersten Schritt wird ein Start-Zufallszahlengenerator 212-UR verwendet, der vorab bekannt ist und im Prinzip allgemein bekannt gegeben werden kann.

[00200] Das Passwort W_0 wird vor der Verschlüsselung der Rohdaten vom Benutzer zur Verfügung gestellt. Als Passwort W_0 dient eine Anzahl von sequentiell angeordneten Bytes mit 8 Bits pro Byte. Um die Auswirkungen unterschiedlicher Passwörter W_0 mit identischem Beginn möglichst ausschließen zu können, wird dem jeweiligen Passwort jeweils seine Länge in Bytes vorangestellt. Alternativ oder zusätzlich kann dem Passwort W_0 auch eine Checksumme vorangestellt werden. Sofern die Länge des Passwortes W_0 einen vorgegebenen Wert unterschreitet wird das Passwort W_0 durch Wiederholung oder Hintereinanderreihung des Passwortes, auf die vorgegebene Länge verlängert und in einem Passwort-Erstellungsschritt S101 ein Start-Passwort W_N erstellt.

[00201] Anschließend wird eine erste Verschlüsselungsmaschine 200a mit Werten initialisiert, die sich aus dem Start-Passwort W_N und den Werten des Start-Zufallszahlengenerators 212-UR ergeben. Das Start-Passwort W_N und die Werte des Start-Zufallszahlengenerators 212-UR werden einer Verknüpfungseinheit 219a in den Schritten S102a und S103a zugeführt. Das Initial-Passwort W_N und die Zufallszahlen des Start-Zufallszahlengenerators 212-UR werden dort verknüpft, beispielsweise durch bitweise durchgeführte Exklusiven-Oder-Verknüpfung (XOR-Verknüpfung) entsprechender Bytes, die daraus resultierenden Werte werden zur Initialisierung der ersten Verschlüsselungsmaschine 200a, d.h. zur Festlegung ihres inneren Zustandes, insbesondere des inneren Zustandes ihrer Transformationseinheit 213a im Schritt S105a und ihres Zufallszahlengenerators 212a im Schritt S104a verwendet. Anschließend wird das Start-Passwort W_N mittels der ersten Verschlüsselungsmaschine 200a in einem Verschlüsselungsschritt S106a verschlüsselt und man erhält ein erstes Passwort W_a .

[00202] Es wird eine zweite Verschlüsselungsmaschine 200b erstellt, indem das erste Passwort W_a und Zufallszahlen, die mit den Zufallszahlengenerator 212a der ersten Verschlüsselungsmaschine 200a erstellt wurden, in den Schritten S102b und S103b an die Verknüpfungs-

einheit 219b übermittelt und von dieser verknüpft werden und die aus dieser Verknüpfung resultierenden Daten zur Festlegung des inneren Zustandes der zweiten Verschlüsselungsmaschine 200b in den Schritten S104b und S105b verwendet werden. Es wird ein zweites Passwort W_b durch Verschlüsselung des ersten Passwortes W_a mit der zweiten Verschlüsselungsmaschine 200b im Schritt S106b erstellt. Dieser Vorgang kann beliebig oft wiederholt werden. Am Ende erhält man eine in Fig. 12 unten dargestellte Verschlüsselungsmaschine 200.

WAHL DER BITS PRO BYTE

[00203] Ganz allgemein kann anstelle des bei dem vorliegenden Ausführungsbeispiel verwendeten Datenspeichers 211 auch ein anderes Speicherlayout verwendet werden. Es ist dabei grundsätzlich möglich, auch Speicher mit einer abweichenden Anzahl von Bits pro Byte zu verwenden, um eine bessere Durchmischung der einzelnen Eingangsdaten zu erreichen. In diesem Fall werden die zu 8 Bit pro Byte vorliegenden Rohdaten bitweise in den Datenspeicher geschrieben, wobei die Zuordnung zwischen Bits und Bytes geändert wird.

ERHÖHUNG DER VERARBEITUNGSGESCHWINDIGKEIT

[00204] Es ist vorteilhaft, jedoch nicht zwingend erforderlich, dass die Transformationsvorschrift g für jedes zu verschlüsselnde Byte separat geändert wird. Es ist auch möglich, dass mehrere Bytes mit derselben Transformationsvorschrift g verschlüsselt werden und erst anschließend eine neue Transformationsvorschrift g bestimmt wird.

[00205] Es ist ebenso möglich, dass die beiden Auswahleinheiten 230, 220 nach der Auswahl einer Grund-Transformationsvorschrift f oder eines Teil-Zufallszahlengenerators 233 lediglich dadurch abgeändert werden, dass der Auswahlzeiger 222-1, 232-1 der ersten Auswahlstufe 220-1 bzw. 230-1 inkrementiert wird, jedoch sonst keine Änderungen vorgenommen werden. Um mehrfache Verschlüsselungen mit derselben Transformationsvorschrift g zu vermeiden, sollte die Auswahl der Inkrementierung des Auswahlzeigers 222, 232 ohne extern verursachte Zustandsänderung durch Zufuhr externer Daten geringer sein als die Länge A bzw. Z der Auswahlregister 221 bzw. 231.

[00206] Ein Verfahren, bei dem mehrere Bytes mit derselben Vorschrift verschlüsselt werden, ist eher anfällig für Angriffe als ein Verfahren, das jeweils eine eigene Transformationsvorschrift g für jedes Byte verwendet.

[00207] Ebenso braucht der Rückkopplungsmechanismus aus Zeitgründen nicht bei jedem Schritt durchgeführt werden. Es ist auch möglich, dass Rückkoppelung der Auswahleinheiten 220, 230 alternierend oder auch nicht synchron erfolgt. Auch kann die Rückkopplung zum Zufallszahlengenerator 212 oder zur Transformationseinheit 213 bei einigen Schritten entfallen. Dennoch ist in diesen Fällen die verwendete Transformationsvorschrift eine andere.

ENTSCHLÜSSELUNG

[00208] Die Entschlüsselung wird im Wesentlichen gleich wie die Verschlüsselung mit einer in Fig. 13 gezeigten Entschlüsselungsmaschine 300 vorgenommen, wobei der innere Zustand der Transformationseinheit 313 und des Zufallszahlengenerators 312 mit dem inneren Zustand der Transformationseinheit 213 und des Zufallszahlengenerators 212 nach der Verarbeitung des jeweils i -ten Bytes bei der Verschlüsselung und bei der Entschlüsselung jeweils identisch sind, natürlich mit Ausnahme der Daten, die die inversen Grund-Transformationen f^{-1} beschreiben und im Falle der dynamischen Änderung der (Grund-)Transformationsvorschriften werden allenfalls noch neben den Inversen der (Grund-)Transformationsvorschriften die jeweils geänderten, nicht invertierten (Grund-)Transformationsvorschriften zur Verfügung gehalten. Gleiches gilt natürlich auch für die Spezial-Transformationsvorschrift h .

[00209] Auch der Aufbau der Verschlüsselungsmaschine 200 und der Entschlüsselungsmaschine 300 ist bis auf die im Folgenden dargestellten Unterschiede identisch. Die Entschlüsselungsmaschine 300 verfügt über einen Datenspeicher 311, einen Zufallszahlengenerator 312 und eine Transformationseinheit 313, wie in Fig. 13 dargestellt.

[00210] Wie auch bei einer Verschlüsselung gemäß dem ersten Ausführungsbeispiel der Erfindung erfolgt die Entschlüsselung (Fig. 3) derart, dass die einzelnen Bytes $c_1, \dots, c_N$ des Datenspeichers 11 sequentiell vom Anfang bis zum Ende in Datenrichtung durchlaufen werden.

[00211] Anders als bei der Verschlüsselung werden anstelle der Transformationsvorschriften g und Grund-Transformationsvorschriften $f_1, \dots, f_A$ die inversen Transformationsvorschriften g^{-1} und inversen Grund-Transformationsvorschriften $f_1^{-1}, \dots, f_A^{-1}$ verwendet. Sofern, wie eine Verbesserung der Erfindung vorschlägt, die Transformationsvorschriften aus Grund-Transformationsvorschriften $f_1, \dots, f_A$ zusammengesetzt werden, reicht es aus, anstelle der Grund-Transformationsvorschriften $f_1, \dots, f_A$ jeweils die inversen Grund-Transformationsvorschriften $f_1^{-1}, \dots, f_A^{-1}$ abzuspeichern.

[00212] Für den Fall, dass die Grund-Transformationsvorschriften f_i oder die Spezial-Transformationsvorschrift h während der Verschlüsselung modifiziert werden, so werden für die Entschlüsselung zusätzlich zu den inversen Transformationsvorschriften $h^{-1}, f_1^{-1} \dots f_A^{-1}$ auch die ursprünglichen Transformationsvorschriften $h, f_1 \dots f_A$ vorgehalten.

[00213] Will man die Inverse g^{-1} einer aus mehreren Grund-Transformationsvorschriften $f_{\mu 1}, f_{\mu 2}, \dots, f_{\mu p}$ zusammengesetzten Transformationsvorschrift g ermitteln, so sind die inversen Grund-Transformationsvorschriften $f_1^{-1}, \dots, f_A^{-1}$ i gegebenenfalls h^{-1} in der umgekehrten Reihenfolge anzuwenden:

$$c = g(b) = h(f_{\mu 1}(f_{\mu 2}(\dots f_{\mu p}(b)\dots)))$$

$$b = g^{-1}(c) = f_{\mu p}^{-1}(f_{\mu p-1}^{-1}(\dots f_{\mu 1}^{-1}(h^{-1}(c))\dots))$$

[00214] Um tatsächlich denselben inneren Zustand konsistent zu erhalten, werden jeweils die Rohdaten Bytes b und die verschlüsselten Bytes c und eventuelle Zufallszahlen in gleicher Weise verwendet, wie bei der Verschlüsselung, beispielsweise zur Abänderung des inneren Zustands der Auswahlseinheiten 320, 330 des Zufallszahlengenerators 312 sowie der Transformationseinheit 313. Insbesondere werden die verschlüsselten und zu entschlüsselnden Bytes in einen hierfür vorgesehenen Datenspeicher 317 geschrieben, die entschlüsselten Bytes in einen hierfür vorgesehenen Datenspeicher 314.

[00215] Die in Fig. 13 dargestellte Entschlüsselungsmaschine wird in Bezug auf die Entschlüsselung gemäß dem ersten Ausführungsbeispiel der Erfindung dargestellt.

[00216] In einem ersten Schritt wird die Entschlüsselungsmaschine 300 mit dem Passwort W_0 initialisiert. Hierbei werden sämtliche internen Speicher des Zufallszahlengenerators 312 sowie der Transformationseinheit 313 in einen durch ein Passwort W_0 bestimmten Initialzustand Y'_0 gebracht. Die spezielle Initialisierung der Entschlüsselungsmaschine für die Entschlüsselungsmaschine wird später unter Bezugnahme auf Fig. 16 näher beschrieben.

[00217] In einem zweiten Schritt wird der Datenspeicher 311 mit verschlüsselten Daten befüllt. Diejenige Menge an verschlüsselten Daten $c_1, \dots, c_N$, die gleichzeitig im Datenspeicher 311 zur Verfügung steht, wird auch als Block oder Datenblock D bezeichnet. Ein Datenzeiger i , mit dem jeweils ein Byte c_i aus dem Datenspeicher 311 ausgewählt wird, wird auf den ersten Speicherplatz des Datenspeichers 311 gesetzt.

[00218] Die folgenden Iterationsschritte werden für sämtliche Bytes des Datenspeichers 311 ausgeführt. Der im Speicher 316 abgespeicherte Datenzeiger i zeigt dabei anfangs auf das erste Byte c_1 des zu verschlüsselnden Datenblocks D und wird nach Durchführung der Iterationsschritte $S_{31} \dots S_{39}$ so lange inkrementiert, bis er auf das letzte Byte b_N des zu verschlüsselnden Datenblocks D zeigt.

[00219] In einem ersten Iterationsschritt S_{31} wird das Byte des Datenblockes D des jeweiligen vom Wert des Datenzeigers i ausgewählten Speicherplatz des Datenspeichers 311 in einen Schlüsseldaten-Zwischenspeicher 317 als Schlüsseldaten-Byte c_i kopiert und der Transformationseinheit 313 zugeführt.

[00220] In einem zweiten Iterationsschritt S_{32} wird eine Folge von Zufallszahlen $\mu_{i,1}, \dots, \mu_{i,p}$ an

die Transformationseinheit 313 übermittelt. Für jede Iteration werden separate Zufallszahlen $\mu_{i,1}$, ..., $\mu_{i,p}$ ermittelt. Diese Zufallszahlen sind identisch zu den Zufallszahlen $\mu_{i,1}$, ..., $\mu_{i,p}$ die im Iterationsschritt S_{22} während der Verschlüsselung der Transformationseinheit 213 übermittelt wurden.

[00221] Die Transformationseinheit 313 ermittelt im dritten Iterationsschritt S_{33} die Transformationsvorschrift g_i^{-1} in Abhängigkeit von ihrem internen Zustand und den ermittelten Zufallszahlen $\mu_{i,1}$, ..., $\mu_{i,p}$.

[00222] Das verschlüsselte Byte c_i wird durch Anwendung der Transformationsvorschrift g_i^{-1} in einem vierten Iterationsschritt S_{34} gemäß $b_i = g_i^{-1}(c_i)$ entschlüsselt, man erhält wiederum das Rohdaten-Byte b_i . Im vorliegenden Fall wird die Transformationsvorschrift g_i durch Zusammensetzung, wie vorstehend beschrieben ermittelt, wobei die Auswahl von T Vorschriften aus den A Grund-Transformationsvorschriften f_1^{-1} , ..., f_A^{-1} für die Zusammensetzung aufgrund der angegebenen Zufallszahlen $\mu_{i,1}$, ..., $\mu_{i,p}$ erfolgt.

[00223] In einem fünften Iterationsschritt S_{35} wird der Inhalt des jeweiligen vom Datenzeiger i ausgewählten Speicherplatzes des Datenspeichers 311 mit dem entschlüsselten Byte b_i überschrieben. Anschließend wird in einem sechsten Iterationsschritt S_{36} das entschlüsselte Byte b_i in einen eigenen Rohdaten-Zwischenspeicher 314 als Rohdaten-Byte b_i geschrieben.

[00224] In einem siebenten Iterationsschritt S_{37} werden das Rohdaten-Byte b_i und das verschlüsselte Byte c_i an den Zufallszahlengenerator 312 und die Transformationseinheit 313 für die künftige Zustandsänderungen dieser übermittelt.

[00225] In einem achten Iterationsschritt S_{38} werden weitere vom Zufallszahlengenerator 312 erzeugte Zufallszahlen $\rho_{i,1}$... $\rho_{i,R}$, diese Zufallszahlen sind identisch zu den Zufallszahlen $\rho_{i,1}$... $\rho_{i,R}$ die im Iterationsschritt S_{28} während der Verschlüsselung übermittelt wurden, an den Zufallszahlengenerator 312 zurück und an die Transformationseinheit 313 für die künftige Zustandsänderungen übermittelt.

[00226] In einem neunten Iterationsschritt S_{39} wird der innere Zustand des Zufallszahlengenerators 312 und der Transformationseinheit 313 mit Hilfe der in den Iterationsschritten S_{37} und S_{38} übermittelten Bytes bzw Zufallszahlen abgeändert. Der innere Zustand des Zufallszahlengenerators 312 ist jetzt identisch zum inneren Zustand des Zufallszahlengenerators 212 nach der Bearbeitung des i -ten Datenbytes. Der innere Zustand der Transformationseinheit 313 ist jetzt gleich dem inneren Zustand der Transformationseinheit 213 nach der Bearbeitung des i -ten Datenbytes, natürlich mit Ausnahme der Daten, die die inversen Grund-Transformationen f^{-1} beschreiben.

[00227] Werden die Grund-Transformationsvorschriften f_i oder die Spezial-Transformationsvorschrift h nach der Verschlüsselung im Iterationsschritt S_{29} modifiziert, so sind bei der Entschlüsselung ebenfalls die Inversen der Grund-Transformationsvorschriften f_i^{-1} bzw die Inverse der Spezial-Transformationsvorschrift h^{-1} analog abzuändern, sodass im Iterationsschritt S_{39} jeweils die Inverse der abgewandelten Grund-Transformationsvorschriften f_i bzw die Inverse der abgewandelten Spezial-Transformationsvorschrift h ermittelt und für den nächsten Entschlüsselungsschritt zur Verfügung steht.

[00228] Je nach Fortschritt der Verschlüsselung entstehen bei der Transformationseinheit 313 sowie beim Zufallszahlengenerator 312 in Abhängigkeit von sowohl den bisherig verarbeiteten entschlüsselten Bytes b_i , den verschlüsselten Bytes c_i und den vom Zufallszahlengenerator 312 gelieferten Zufallszahlen $\rho_{i,1}$... $\rho_{i,R}$ unterschiedliche, innere Zustände.

[00229] Die Auswahl der Transformationsvorschrift g_i bzw. von deren Inversen g_i^{-1} ist daher sowohl direkt abhängig von dem inneren Zustand der Transformationseinheit 313 zum Zeitpunkt der Verschlüsselung des Rohdatenbytes b_i als auch indirekt durch die von dem Zufallszahlengenerator 312 zum gleichen Zeitpunkt bereitgestellten Zufallszahlen $\rho_{i,1}$... $\rho_{i,R}$, $\mu_{i,1}$, ..., $\mu_{i,p}$, die wiederum auch von dem inneren Zustand des Zufallszahlengenerators 312 in Abhängigkeit von den bis dahin verarbeiteten Rohdatenbytes und verschlüsselten Bytes erzeugt werden.

[00230] Die Auswahl der Transformationsvorschrift g_i ist daher abhängig von den Werten und der Reihenfolge aller verschlüsselten Bytes $c_1, \dots, c_{i-1}$ sowie der Bytes $b_1, \dots, b_{i-1}$. Nach der Verschlüsselung wird der Wert des Zeigers i inkrementiert.

[00231] Nachdem die im Datenspeicher 311 abgelegten verschlüsselten Bytes $c_1, \dots, c_N$ alleamt entschlüsselt sind, werden diese für die weitere Verwendung zur Verfügung gehalten. Nach der allfälligen weiteren Verarbeitung der Daten im Datenspeicher 311 wird dieser gemäß dem zweiten Schritt eventuell mit weiteren zu entschlüsselnden Daten wieder aufgefüllt, die analog wie oben verschlüsselt werden.

[00232] Der innere Zustand sowohl des Zufallszahlengenerators 312 als auch der Transformationseinheit 313 ist am Beginn der neuerlichen Verarbeitung von weiteren Daten im Datenspeicher 311 gleich wie am Ende der vorherigen Verarbeitung der Daten des Datenspeichers 311.

[00233] Bei der Entschlüsselung von nach dem zweiten Ausführungsbeispiel der Erfindung verschlüsselten Daten beginnt die Entschlüsselung in umgekehrter Abfolge der Daten (Fig. 4, 5). Die Entschlüsselungsmaschine 300 wird in den Initialzustand Y'_0 versetzt. Der Initialzustand Y'_0 unterscheidet sich vom Initialzustand Y_0 nur durch die Daten, die die inversen Grund-Transformationen f^{-1} an Stelle der Grundtransformationen f beschreiben. Allenfalls kann es bei der Änderung von Grund-Transformationsvorschriften $f_1^{-1} \dots f_A^{-1}$ notwendig werden, die Grund-Transformationsvorschriften $f_1 \dots f_A$ zur Verfügung zu halten. Ebenso kann es notwendig sein, im Falle der Verwendung der Spezial-Transformationsvorschriften, diese auch in nicht invertierter Form zur Verfügung zu haben. Der Datenzeiger i wird zunächst entgegen der Datenrichtung des Datenspeichers 311 verschoben, wobei die einzelnen Bytes jeweils entschlüsselt werden. Anschließend wird die Transformationseinheit 313 wiederum in den Initialzustand Y'_0 versetzt, wobei die einzelnen Bytes im Datenspeicher 311 vom ersten Byte bis zum letzten Byte in Datenrichtung entschlüsselt werden. Nach diesem zweiten Durchlauf befindet sich die Rohdatenfolge im Datenspeicher 311. Sofern die Rohdatenfolge mehrfach verschlüsselt wurde, wird auch der Entschlüsselungsvorgang entsprechend mehrfach wiederholt, um die Rohdatenfolge zu erhalten. (Fig. 6)

[00234] Bei der Entschlüsselung von nach dem dritten Ausführungsbeispiel der Erfindung verschlüsselten Daten beginnt die Entschlüsselung beim letzten Byte des jeweiligen Blocks (Fig. 7, 9). Die Entschlüsselungsmaschine 300 wird in den Initialzustand Y'_0 versetzt. Der Datenzeiger i wird zunächst entgegen der Datenrichtung des Datenspeichers 311 verschoben, wobei die einzelnen Bytes jeweils entschlüsselt werden. Anschließend wird die Entschlüsselungsmaschine 300 wiederum in den Initialzustand Y'_0 versetzt, ihr momentaner innerer Zustand Y_x wird verworfen. Anschließend werden die einzelnen Bytes im Datenspeicher 311 vom ersten Byte bis zum letzten Byte in Datenrichtung entschlüsselt. Nach diesem zweiten Durchlauf befindet sich die Rohdatenfolge im Datenspeicher 311. Sofern die Rohdatenfolge mehrfach verschlüsselt wurde, wird auch der Entschlüsselungsvorgang entsprechend mehrfach wiederholt, um die Rohdatenfolge des jeweiligen Datenblocks zu erhalten.

[00235] Die Entschlüsselungsmaschine 300 hat den inneren Zustand Y'_1 . Beim nächsten Durchlauf wird der Zustand Y'_1 als Initialzustand verwendet und das vorstehende Vorgehen erneut durchgeführt, wobei der innere Zustand, in dem die Entschlüsselungsmaschine 300 am Ende der Entschlüsselung des jeweiligen Datenblocks ist, als Initialzustand für die Entschlüsselung des jeweils nächsten Datenblocks verwendet wird.

[00236] Der jeweils entschlüsselte Datenblock D wird für eine weitere Verwendung zur Verfügung gehalten.

[00237] Der nächste verschlüsselte bzw. zu entschlüsselnde Datenblock wird in den Datenspeicher 311 geladen. Der bei der Entschlüsselung des vorherigen Blocks zwischengespeicherte innere Zustand ist der Initialzustand für den jetzt zu bearbeitenden Block. Dieser wird in die Entschlüsselungsmaschine 300 geladen und der für den ersten Block dargestellte Vorgang erneut durchgeführt. Dieser Vorgang kann für sämtliche zu entschlüsselnde Datenblöcke durchgeführt werden.

[00238] Wie bereits erwähnt, besteht bei der Verschlüsselung auch die Möglichkeit, den inneren Zustand, in dem sich die Verschlüsselungsmaschine 200 nach dem ersten Verschlüsselungsdurchlauf befindet, als inneren Zustand zwischenspeichern und vor der Verschlüsselung des jeweils nächsten Datenblocks D als Initialzustand zu verwenden. In diesem Fall braucht bei der Entschlüsselung der innere Zustand der Entschlüsselungsmaschine 300 nicht abgespeichert zu werden. Bei der Entschlüsselung eines zweiten oder späteren Datenblocks kann jeweils der innere Zustand als Initialzustand herangezogen werden, der an der Entschlüsselungsmaschine 300 am Ende der Entschlüsselung des ersten bzw. jeweils vorangehenden Datenblocks vorlag.

[00239] Der Zufallszahlengenerator 312 (Fig. 13) der Entschlüsselungsmaschine 300 entspricht ident dem Zufallszahlengenerator 212 (Fig. 1) der Verschlüsselungsmaschine 200. Die Teil-Zufallszahlengeneratoren 233 und 333 sind sich gleich. Die Auswahlseinheiten 230 und 330 (Fig. 10 und Fig. 14) sind sich gleich.

[00240] Die Transformationseinheit 313 (Fig. 13) der Entschlüsselungsmaschine 300 entspricht der Transformationseinheit 213 der Verschlüsselungsmaschine 200 mit dem Unterschied, dass anstelle der Mittel 240 zur Bildung der Grund-Transformationsvorschrift f Mittel 340 zur Bildung ihrer jeweiligen Inversen f^{-1} zur Verfügung stehen. Die Auswahlseinheiten 220 und 320 (Fig. 11 und Fig. 15) sind sich gleich.

INITIALISIERUNG DER ENTSCHLÜSSELUNGSMASCHINE

[00241] Die Initialisierung der Entschlüsselungsmaschine 300 erfolgt im Wesentlichen wie die Initialisierung der Verschlüsselungsmaschine 200 (Fig. 12). Bei der Initialisierung wird eine Anzahl von Verschlüsselungsmaschinen 1a, 1b, ... gebildet, jeweils auf dieselbe Weise wie bei der Initialisierung der betreffenden Verschlüsselungsmaschine 200. Aus dem Initial-Zustand Y_0 der letzten gebildeten Verschlüsselungsmaschine wird anschließend der Initialzustand Y'_0 der Entschlüsselungsmaschine 300 gebildet. Die Zufallszahlengeneratoren 212 und 312 (Fig. 1, 13) sowie die Auswahlseinheiten 220 und 320 (Fig. 11, 15) sind sich gleich. Der Unterschied der Zustände Y_0 und Y'_0 besteht zwischen den Verarbeitungseinheiten 240 und 340. Um eine Decodierung erzielen zu können, werden anstelle der Grund-Transformationsvorschrift $f_1 \dots f_A$ jeweils ihre Inversen $f_1^{-1} \dots f_A^{-1}$ verwendet. Diese werden von der Verschlüsselungsmaschine 200 einer Inversionseinheit 319 übermittelt, diese bestimmt die jeweiligen Inversen $f_1^{-1} \dots f_A^{-1}$ sowie gegebenenfalls die Inverse h^{-1} der Spezial-Transformationsvorschrift und speichert diese in den betreffenden Speichern der Transformationseinheit 313 der Entschlüsselungsmaschine 300. Ebenso werden gegebenenfalls die Grund-Transformationsvorschriften $f_1 \dots f_A$ selbst sowie die Spezial-Transformationsvorschrift h und ihre Inverse h^{-1} abgespeichert.

[00242] Zum Abspeichern und neu Laden eines inneren Zustands Y'_0 steht bei der Entschlüsselungsmaschine 300 ein Zustandsspeicher 318 zur Verfügung.

Patentansprüche

1. Verfahren zur Verschlüsselung von Rohdaten, wobei die Rohdaten Bytes (b_i) umfassend eine vorgegebene Anzahl B von Bits aufweisen und die Bytes in einer vorgegebenen Reihenfolge vorliegen,
 - wobei eine Verschlüsselungsmaschine (200) umfassend eine Transformationseinheit (213) herangezogen wird, die abhängig von ihrem inneren Zustand eine Verschlüsselungsart in Form einer Transformationsvorschrift (g_i) zur Ersetzung des Rohdaten-Byte (b_i) durch ein verschlüsseltes Byte (c_i) bestimmt,
 - wobei die einzelnen Bytes (b_i) der Rohdaten jeweils separat und durch verschiedene eindeutige Transformationsvorschriften (g_i) verschlüsselt werden und für jedes Byte (b_i) der Rohdaten von der Transformationseinheit (213) jeweils ein verschlüsseltes Byte (c_i) ermittelt wird,
 - wobei gegebenenfalls der jeweilige das Byte (b_i) der Rohdaten enthaltende Datenspeicher (211) mit dem verschlüsselten Byte (c_i) überschrieben wird, und
 - wobei die Transformationseinheit (213) vorab in einen von einem Passwort (W_0) abhängigen Initialzustand gebracht wird, und ein der Verschlüsselungsmaschine (200) zugehöriger deterministischer Zufallszahlengenerator (212) abhängig vom Passwort (W_0) ebenfalls in einen Initialzustand gebracht wird,

dadurch gekennzeichnet,

dass der innere Zustand der Transformationseinheit (213) sowie der innere Zustand des Zufallszahlengenerators (212) nach einer Anzahl von Verschlüsselungen von Bytes, insbesondere nach der Verschlüsselung jedes Bytes, abhängig vom jeweils zu verschlüsselnden Rohdaten-Byte und/oder vom jeweiligen Endergebnis der Verschlüsselung geändert und aktualisiert wird und dieser neue innere Zustand der jeweils nächsten Verschlüsselung zugrunde gelegt wird,

wobei insbesondere der Zufallszahlengenerator (212) Zufallszahlen erstellt, die für die Auswahl der Transformationsvorschriften verwendet werden, und/oder der Zufallszahlengenerator (212) Zufallszahlen erstellt, die für die Aktualisierung der inneren Zustände des Zufallszahlengenerators (212) und der Transformationseinheit (213) herangezogen werden.

2. Verfahren nach Anspruch 1, wobei die Rohdaten in Blöcke, insbesondere in gleich lange, Blöcke (D) umfassend eine vorgegebene Anzahl von N Bytes, unterteilt werden und blockweise verschlüsselt werden, indem
 - a) die Transformationseinheit (213) und der Zufallszahlengenerator (212) in einen Initialzustand (Y_0) gesetzt werden,
 - b) die einzelnen sequentiell vorliegenden Bytes der Rohdaten innerhalb desselben Blocks hintereinander in der vorgegebenen Reihenfolge vom ersten bis zum letzten Byte von der Transformationseinheit (213) verschlüsselt werden,
 - c) nachdem alle Bytes des Blocks (D) verschlüsselt wurden, die Transformationseinheit (213) sowie der Zufallszahlengenerator (212) erneut in den Initialzustand (Y_0) gesetzt werden, und
 - d) die einzelnen sequentiell vorliegenden Bytes der im jeweiligen Block (D) vorliegenden Daten innerhalb desselben Blocks (D) entgegen der vorgegebenen Reihenfolge vom letzten bis zum ersten Byte von der Transformationseinheit (213) verschlüsselt werden.
3. Verfahren nach Anspruch 2, **dadurch gekennzeichnet**, dass die Verschlüsselungsschritte a) bis d) aus Anspruch 2 mehrfach für denselben Block (D) wiederholt werden, wobei gegebenenfalls bei der letzten Wiederholung ausschließlich die Schritte a) und b) durchgeführt werden.
4. Verfahren nach Anspruch 2 oder 3, **dadurch gekennzeichnet**, dass nach der Durchführung des Schritts b) oder eines der Schritte b) oder alternativ nach der Durchführung des Schritts d) oder eines der Schritte d) und noch vor der darauffolgenden Zurücksetzung in den jeweiligen Initialzustand (Y_0) der innere Zustand der Transformationseinheit (213) und des Zufallszahlengenerators (212) abgespeichert wird und als neuer Initialzustand für die Verschlüsselung des jeweils nächsten Blocks (D) herangezogen wird.

5. Verfahren nach einem der vorangehenden Ansprüche, **dadurch gekennzeichnet**, dass gemeinsam mit dem Setzen eines Initialzustands der Verschlüsselungsmaschine (200) eine Anzahl von Grund-Transformationsvorschriften (f) in Abhängigkeit vom Initialzustand vorgegeben wird, und
 - abhängig vom inneren Zustand der Transformationseinheit (213) eine Anzahl T aus diesen Grund-Transformationsvorschriften in einer vorgegebenen Reihenfolge ausgewählt wird,
 - wobei die Transformationsvorschrift (g) aus den einzelnen ausgewählten Grund-Transformationsvorschriften (f) zusammengesetzt wird, sodass bei der Anwendung der Transformationsvorschrift (g) auf das zu verschlüsselnde Byte die ausgewählten Grund-Transformationsvorschriften (f) in der jeweiligen Reihenfolge auf das zu verschlüsselnde Byte angewandt werden.
6. Verfahren nach Anspruch 5, **dadurch gekennzeichnet**, dass die Transformationsvorschrift (g) aus den einzelnen ausgewählten Grund-Transformationsvorschriften (f) sowie zusätzlich einer Spezial-Transformationsvorschrift (h) zusammengesetzt wird, sodass bei der Anwendung der Transformationsvorschrift (g) auf das zu verschlüsselnde Byte die ausgewählten Grund-Transformationsvorschriften (f) in der jeweiligen Reihenfolge auf das zu verschlüsselnde Byte angewandt werden und, insbesondere danach, noch zusätzlich die Spezial-Transformationsvorschrift (h) angewendet wird, und dass die Spezial-Transformationsvorschrift (h) bei jedem Verschlüsselungsschritt oder nach einer Anzahl von Verschlüsselungsschritten, abgeändert wird, insbesondere ausschließlich für eine vorgegebene Anzahl von Abänderungen, und dass insbesondere die Verwendung und/oder Abänderung der Spezial-Transformationsvorschrift (h) je nach Gegebenheit an- und abgeschaltet wird.
7. Verfahren, nach einem der vorangehenden Ansprüche zur Initialisierung einer Verschlüsselungsmaschine (200), **dadurch gekennzeichnet**, dass vor der Verschlüsselung der Rohdaten
 - a) als Passwort (W_0) eine Anzahl von sequentiell angeordneten Bytes vorgegeben wird,
 - b) gegebenenfalls dem Passwort (W_0) seine Länge in Bytes und/oder eine Checksumme beigefügt wird,
 - c) gegebenenfalls, sofern die Länge des Passwortes (W_0) eine vorgegebene Länge unterschreitet, das Passwort (W_0), insbesondere durch Wiederholung des Passwortes (W_0), auf die vorgegebene Länge verlängert wird und somit ein normalisiertes Passwort (W_N) erstellt wird,
 - d) ein vorgegebener Start-Zufallszahlengenerator (212-UR) mit vorgegebenen Werten initialisiert wird,
 - e) die vom Start-Zufallszahlengenerator (212-UR) erstellten Zufallszahlen mit den Bytes des, gegebenenfalls normalisierten Passwortes (W_N) verknüpft werden und eine erste Initialisierungs-Datenfolge erstellt wird,
 - f) diese Initialisierungsfolge wird verwendet um eine erste Verschlüsselungsmaschine (200a), umfassend eine erste Transformationseinheit (213a) und einen ersten Zufallszahlengenerator (212a) zu erstellen und zu initialisieren,
 - g) das erste Passwort (W_N) mit der ersten Verschlüsselungsmaschine (200a) verschlüsselt wird, und damit ein zweites Passwort (W_a) erhalten wird,
 - h) das erstellte zweite Passwort (W_a) mit von dem Zufallszahlengenerator (212a) der ersten Verschlüsselungsmaschine (200a) erstellten Zufallszahlen verknüpft wird und so eine zweite Initialisierungs-Datenfolge erstellt wird, und diese zur Erstellung und Initialisierung einer weiteren Verschlüsselungsmaschine (200b) verwendet wird,
 - i) wobei die folgenden Schritte j), k) und l) optional mehrfach durchgeführt werden, nämlich
 - j) dass jeweils das für die Initialisierung der Verschlüsselungsmaschine (200b,...) verwendete Passwort ($W_{a,...}$) mit dieser Verschlüsselungsmaschine (200b,...) verschlüsselt wird und dadurch ein nächstes Passwort ($W_{b,...}$) erhalten wird,

- k) dass dieses Passwort (W_b) mit Zufallszahlen des jeweils zuletzt erstellten Zufallszahlengenerators (212b, ...) verknüpft wird und so eine weitere Initialisierungs- Datenfolge erstellt wird, und
 - l) dass eine nachfolgende weitere Verschlüsselungsmaschine (200c,...) mit einer Transformationseinheit (213c,...) und einem Zufallszahlengenerator (212c,...) mit der jeweiligen zuletzt erstellten Initialisierungs-Datenfolge definiert, erstellt und initialisiert wird, und
 - m) dass die letzte derart erstellte Verschlüsselungsmaschine (200a, 200b, ...) als Verschlüsselungsmaschine (200) herangezogen wird.
8. Verfahren zur Entschlüsselung von nach einem der vorangehenden Ansprüche verschlüsselten Daten, wobei die verschlüsselten Daten Bytes umfassend eine vorgegebene Anzahl B von Bits aufweisen und die Bytes in einer vorgegebenen Reihenfolge vorliegen,
- wobei eine Entschlüsselungsmaschine (300) umfassend eine Transformationseinheit (313) herangezogen wird, die abhängig von ihrem inneren Zustand eine Entschlüsselungsart in Form einer Transformationsvorschrift (g_i^{-1}) bestimmt,
 - wobei die einzelnen Bytes (c_i) der zu entschlüsselnden Daten jeweils separat mittels der Transformationsvorschrift (g_i^{-1}) entschlüsselt werden und für jedes Byte (c_i) der verschlüsselten Daten jeweils ein entschlüsseltes Byte (b_i) von der Entschlüsselungsmaschine (313) ermittelt wird,
 - wobei gegebenenfalls der jeweilige das Byte (c_i) der verschlüsselten Daten enthaltende Datenspeicher (311) mit dem entschlüsselten Byte (b_i) überschrieben wird,
 - wobei die Transformationseinheit (313) vorab in einen von einem Passwort (W_0) abhängigen Initialzustand gebracht wird und ein der Entschlüsselungsmaschine (300) zugehöriger deterministischer Zufallszahlengenerator (312) abhängig vom Passwort (W_0) in einen Initialzustand gebracht wird,
- dadurch gekennzeichnet,**
- dass der innere Zustand der Transformationseinheit (313) sowie der innere Zustand des Zufallszahlengenerators (312) nach einer Anzahl von Entschlüsselungen von Bytes, insbesondere nach der Entschlüsselung jedes Bytes, abhängig vom jeweils verschlüsselten Byte und/oder vom jeweiligen entschlüsselten Byte geändert und aktualisiert wird und dieser neue innere Zustand der jeweils nächsten Entschlüsselung zugrunde gelegt wird, wobei insbesondere
 - der Zufallszahlengenerator (312) Zufallszahlen erstellt, die für die Auswahl der inversen Transformationsvorschriften verwendet werden, und/oder der Zufallszahlengenerator (312) Zufallszahlen erstellt, die für die Aktualisierung der inneren Zustände des Zufallszahlengenerators (312) und der Transformationseinheit (313) herangezogen werden.
9. Verfahren nach Anspruch 8, **dadurch gekennzeichnet,** dass die zu entschlüsselnden Daten in, insbesondere gleich lange, Blöcke (D) umfassend eine vorgegebene Anzahl N von Bytes unterteilt werden, wobei die Anzahl der Bytes pro Block (D) der Anzahl der bei der Verschlüsselung verwendeten Bytes pro Block (D) entspricht, und dass die zu entschlüsselnden Daten blockweise entschlüsselt werden, indem
- a) die Transformationseinheit (313) sowie der Zufallszahlengenerator (312) in einen Initialzustand (Y'_0) abhängig vom jeweiligen Passwort (W_0) gesetzt wird.
 - b) die einzelnen sequentiell vorliegenden Bytes der im jeweiligen Block (D) vorliegenden zu entschlüsselnden Daten innerhalb desselben Blocks (D) entgegen der vorgegebenen Reihenfolge vom letzten bis zum ersten Byte von der Transformationseinheit (313) entschlüsselt werden,
 - c) nachdem alle Bytes des Blocks (D) entschlüsselt wurden, die Transformationseinheit (313) sowie der Zufallszahlengenerator (312) erneut in den Initialzustand (Y'_0) gesetzt wird, und
 - d) die einzelnen sequentiell vorliegenden Bytes der so erhaltenen Daten innerhalb desselben Blocks (D) hintereinander in der vorgegebenen Reihenfolge vom ersten bis zum letzten Byte von der Transformationseinheit (313) entschlüsselt werden.

10. Verfahren nach Anspruch 9, **dadurch gekennzeichnet**, dass die Entschlüsselungsschritte a) bis d) mehrfach für denselben Block (D) durchgeführt werden, wobei gegebenenfalls bei der letzten Durchführung ausschließlich die Schritte a) und b) ausgeführt werden.
11. Verfahren nach einem der Ansprüche 9 oder 10, **dadurch gekennzeichnet**, dass nach der Durchführung des Schrittes b) oder eines der Schritte b) oder alternativ nach der Durchführung des Schrittes d) oder eines der Schritte d) und noch vor der darauffolgenden Zurücksetzung in den jeweiligen Initialzustand (Y'_0) der innere Zustand der Verschlüsselungsmaschine (300), bestehend aus dem inneren Zustand der Transformationseinheit (313) und dem inneren Zustand des Zufallszahlengenerators (312) abgespeichert wird und als neuer Initialzustand für die Entschlüsselung des jeweils nächsten Blocks (D) herangezogen wird.
12. Verfahren nach Anspruch 8 bis 11, **dadurch gekennzeichnet**, dass gemeinsam mit dem Setzen eines Initialzustands (Y'_0) der Entschlüsselungsmaschine (300) eine Anzahl T von Grund-Transformationsvorschriften (f^{-1}) in Abhängigkeit vom Initialzustand der Transformationseinheit (313) vorgegeben wird, und
 - abhängig vom inneren Zustand eine Anzahl T aus diesen Grund-Transformationsvorschriften (f^{-1}) in einer vorgegebenen Reihenfolge ausgewählt wird, wobei die Transformationsvorschrift (g^{-1}) aus den einzelnen ausgewählten Grund-Transformationsvorschriften (f^{-1}) zusammengesetzt wird, sodass bei der Anwendung der Transformationsvorschrift (g^{-1}) auf das zu entschlüsselnde Byte die ausgewählten Grund-Transformationsvorschriften (f^{-1}) in der jeweils umgekehrten Reihenfolge auf das zu verschlüsselnde Byte angewandt werden, und
 - dass insbesondere die Grund-Transformationsvorschriften (f^{-1}) der Transformationseinheit (313) den Inversen der Grund-Transformationsvorschriften (f) der Transformationseinheit (213) einer mit demselben Passwort (W_0) initialisierten Verschlüsselungsmaschine (300) entsprechen, und
 - dass insbesondere die Reihenfolge der Anwendung der einzelnen Grund-Transformationsvorschriften (f^{-1}) für die Bildung einer Transformationsvorschrift (g^{-1}) in der umgekehrten Reihenfolge wie bei der Verschlüsselung erfolgt.
13. Verfahren nach Anspruch 12, **dadurch gekennzeichnet**, dass die Transformationsvorschrift (g^{-1}) aus der Spezial-Transformationsvorschrift (h^{-1}) und aus den einzelnen ausgewählten Grund-Transformationsvorschriften (f^{-1}) gebildet wird, sodass bei der Anwendung der Transformationsvorschrift (g^{-1}) auf das zu entschlüsselnde Byte, insbesondere zuerst, die Spezial-Transformationsvorschrift (h^{-1}) und, insbesondere anschließend, die ausgewählten Grund-Transformationsvorschriften (f^{-1}) in der jeweils umgekehrten Reihenfolge auf das zu entschlüsselnde Byte (c_i) angewandt werden,
 - dass die Spezial-Transformationsvorschrift (h^{-1}) bei jedem Entschlüsselungsschritt oder nach einer Anzahl von Entschlüsselungsschritten abgeändert wird, insbesondere ausschließlich für eine vorgegebene Anzahl von Änderungen, und
 - dass insbesondere die Verwendung und/oder Abänderung der Spezial-Transformationsvorschrift (h^{-1}) je nach Gegebenheit an und ab geschaltet wird.
14. Verfahren nach einem der Ansprüche 8 bis 13, **dadurch gekennzeichnet**, dass zur Initialisierung einer Entschlüsselungsmaschine (300)
 - eine Verschlüsselungsmaschine (200) nach Anspruch 7 erstellt wird, und dass anschließend eine Entschlüsselungsmaschine (300) mit einem Zufallszahlengenerator (312) und einer Transformationseinheit (313) erstellt wird,
 - der innere Zustand des Zufallszahlengenerators (212) der Verschlüsselungsmaschine (200) auf den inneren Zustand des Zufallszahlengenerators (312) der Entschlüsselungsmaschine (300) übertragen wird und
 - der innere Zustand der Transformationseinheit (213) der Verschlüsselungsmaschine (200) auf den inneren Zustand der Transformationseinheit (313) der Entschlüsselungsmaschine (300) übertragen wird, die Inversen der Transformationsvorschriften (g) und/oder Grund-Transformationsvorschriften (f), sowie gegebenenfalls der Spezial-Transformationsvorschrift (h), ermittelt werden und als inverse Transformationsvor-

schriften (g^{-1}) und/oder inverse Grund- Transformationsvorschriften (f^{-1}), sowie gegebenenfalls der inversen Spezial- Transformationsvorschrift (h^{-1}), in die Transformationseinheit (313) der Entschlüsselungsmaschine (300) gespeichert werden.

und insbesondere im Falle einer Änderung der Transformationsvorschriften während der Kodierung der Daten auch die nicht invertierten Transformationsvorschriften (g) und/oder die nicht invertierte Grund-Transformationsvorschriften (f), sowie gegebenenfalls die nicht invertierten Spezial-Transformationsvorschrift (h), gespeichert werden

15. Verfahren zum Erstellen von Zufallszahlen, **dadurch gekennzeichnet**, dass eine Anzahl H von Auswahlregistern (231; 331) mit jeweils einer vorgegebenen Anzahl Z von Werten von 1 bis Z in beliebiger Reihenfolge vorgegeben werden,
 - a) dass dem ersten Auswahlregister (231-1; 331-1) eine erste Adressiereinheit (232-1; 332-1) vorgeschaltet ist, die mittels eines, insbesondere bei jedem Verschlüsselungsschritt zyklisch inkrementierten, in der Adressiereinheit (232-1; 332-1) befindlichen, ersten Zeigerwertes (k_1) auf eine Speicherposition des ersten Auswahlregisters (231-1; 331-1) verweist,
 - b) dass die an der vom ersten Zeiger (k_1) ausgewählten Speicherposition im ersten Auswahlregister (231-1; 331-1) befindliche Zahl (θ_1) ermittelt wird und diese mit der in der zweiten Adressiereinheit (232-2; 332-2) befindlichen Zahl verknüpft, insbesondere modulo Z genommen und um eins erhöht wird und dann der neue Inhalt der Adressiereinheit (232-2; 332-2) gebildet wird, und der so erhaltene Wert einen zweiten Zeigerwert (k_2) bildet, der dann zur Adressierung einer Speicherposition im zweiten Auswahlregister (231-2; 331-2) herangezogen wird,
 - c) dass gegebenenfalls für die weiteren Adressiereinheiten (232-3,...232- H ; 332-3,...332- H) und die weiteren Auswahlregister (231-3...231- H ; 331-1...331- H) analog zur Adressiereinheit (232-2; 332-2) und zum Auswahlregister (231-2; 331-2) vorgegangen wird, und/oder für die weiteren Auswahlregister (231-3, ... 231- H ; 331-3, ... 331- H) jeweils die an der vom jeweiligen Zeiger ($k_1 \dots k_G$) ausgewählten Speicherposition im jeweiligen Auswahlregister (231; 331) befindliche Zahl ermittelt wird und dem jeweils nächsten Zeiger ($k_1 \dots k_H$) zugewiesen oder zu diesem hinzugezählt oder von diesem abgezogen wird, und der Wert des jeweils nächsten Zeigers ($k_1 \dots k_H$) zur Adressierung einer Speicherposition im jeweils nächsten Auswahlregister (231; 331) herangezogen wird,
 - d) dass eine Anzahl von deterministischen Teil-Zufallszahlengeneratoren (233-1...233- Z , 333-1...333- Z) vorgegeben wird,
 - e) dass der vom letzten Zeigerwert (k_H) der Adressiereinheit (232- H ; 332- H) adressierte Wert (θ_H) des letzten Auswahlregisters (231- H ; 331- H) zur Auswahl des Teil-Zufallszahlengenerators (233-1...233- Z , 333-1...333- Z) herangezogen wird,
 - f) dass gegebenenfalls zur Auswahl einer Mehrzahl von Zufallszahlengeneratoren jeweils der erste Zeiger (k_1) nach vorgegebenen Kriterien abgeändert, insbesondere erhöht, verringert, inkrementiert oder dekrementiert, wird und die Schritte a) bis e) erneut durchgeführt werden, und
 - g) mit dem jeweils ausgewählten Teil-Zufallszahlengenerator (233; 333) eine Zufallszahl erstellt und zur Verfügung gehalten wird.
16. Verfahren nach Anspruch 15, **dadurch gekennzeichnet**, dass abhängig von am Eingang (234; 334) anliegenden Werten, insbesondere dem zu verschlüsselnden Byte und/oder dem verschlüsselten Byte und/oder von Zufallszahlen (p), der Wert eines der Adressiereinheiten (232; 332) abgeändert wird,
 - wobei diese Werte mit dem in der jeweiligen Adressiereinheit (232; 332) befindlichen Wert verknüpft werden, vorzugsweise das Resultat dann modulo Z genommen und um eins erhöht wird, und der Wert dann der jeweiligen Adressiereinheit (232) zugewiesen wird,
 - wobei gegebenenfalls nach Vorliegen eines externen Triggersignals, insbesondere ausgelöst durch eine vorzunehmende Verschlüsselung oder Entschlüsselung eines oder einer vorgegebenen Anzahl von Bytes, jeweils der Wert der nächsten Adressier-

einheit abgeändert wird und für den Fall, dass im vorherigen Schritt der Wert der letzten Adressiereinheit (232-H; 332-H) abgeändert wurde, wiederum die erste Adressiereinheit (232-1; 332-1) abgeändert wird.

17. Verfahren nach Anspruch 15 oder 16, **dadurch gekennzeichnet**, dass abhängig von am Eingang (234; 334) anliegenden Werten (b_i , c_i , p_i), insbesondere einem zu verschlüsseln- den Byte (b_i) und/oder einem verschlüsselten Byte (c_i) und/oder von Zufallszahlen (p_i) eines der Auswahlregister (231; 331) abgeändert wird, und insbesondere zwei in den Auswahlregistern (231; 331) abgespeicherte Werte miteinander vertauscht werden,
 - wobei der Index des einen Wertes eine Reihe von vorgegebenen Werten nach einer vorab vorgegebenen Reihenfolge durchläuft, insbesondere durch zyklisches Hochzählen zwischen 1 und Z, und der Index des zweiten Wertes durch die am Eingang (234; 334) anliegenden Werte festgelegt wird,
 - wobei gegebenenfalls nach Vorliegen eines externen Triggersignals, insbesondere ausgelöst durch eine vorzunehmende Verschlüsselung oder Entschlüsselung eines Bytes oder einer vorgegebenen Anzahl von Bytes, nach jeder Vertauschung oder nach einer vorgegebenen Anzahl von Vertauschungen in einem Auswahlregister (231; 331) zum jeweils nächsten Auswahlregister (231; 331) gewechselt wird und für den Fall, dass im vorherigen Schritt in dem letzten Auswahlregister (231-H; 331-H) bereits die vorgegebene Anzahl von Vertauschungen statt gefunden hat, wieder das erste Auswahlregister (231-1; 331-1) für Abänderungen herangezogen wird.
18. Verfahren nach einem der Ansprüche 1 bis 14, **dadurch gekennzeichnet**, dass die Erstellung von Zufallszahlen gemäß mindestens einem der Ansprüche 15 bis 17 erfolgt.
19. Verfahren zur Ermittlung und Anwendung einer Transformationsvorschrift (g_i , g_i^{-1}) für Verschlüsselungsverfahren oder Entschlüsselungsverfahren, insbesondere nach einem der Ansprüche 1 bis 14 oder 18, **dadurch gekennzeichnet**, dass eine Anzahl G von Auswahlregistern (221; 321) mit jeweils einer vorgegebenen Anzahl A Werten 1 bis A in beliebiger Reihenfolge vorgegeben werden, wobei in jedem Auswahlregister (221; 321) jeweils eine Anzahl von einzelnen Speicherpositionen im jeweiligen Auswahlregister (221; 321) repräsentierenden Zahlen vorgegeben wird, wobei jede eine Speicherposition repräsentierende Zahl jeweils einmal im Auswahlregister (221; 321) enthalten ist,
 - a) dass dem ersten Auswahlregister (221-1; 321-1) eine erste Adressiereinheit (222-1; 322-1) vorgeschaltet ist, die mittels eines, insbesondere bei jedem Verschlüsselungsschritt zyklisch inkrementierten, in der Adressiereinheit (222-1; 322-1) befindlichen ersten Zeigerwertes (j_1) auf eine Speicherposition des ersten Auswahlregisters (221-1; 321-1) verweist,
 - b) dass die an der vom ersten Zeiger (j_1) ausgewählten Speicherposition im ersten Auswahlregister (221-1; 321-1) befindliche Zahl (ϕ_1) ermittelt wird und diese mit der in der zweiten Adressiereinheit (222-2; 322-2) befindlichen Zahl verknüpft, insbesondere modulo A genommen und um eins erhöht wird, und dann der neue Inhalt der Adressiereinheit (222-2; 322-2) gebildet wird, und der so erhaltene Wert einen zweiten Zeigerwert (j_2) bildet, der dann zur Adressierung einer Speicherposition im zweiten Auswahlregister (221-2; 321-2) herangezogen wird,
 - c) dass gegebenenfalls für die weiteren Adressiereinheiten (222-3,...222-G; 322-3,...322-G) und die weiteren Auswahlregister (221-3...221-G; 321-1...321-G) analog zur Adressiereinheit (222-2; 322-2) und zum Auswahlregister (221-2; 321-2) vorgegangen wird, und/oder dass für die weiteren Auswahlregister (221-3, ... 221-G ; 321-3, ... 321-G) jeweils die an der vom jeweiligen Zeiger ($j_1 \dots j_G$) ausgewählten Speicherposition im jeweiligen Auswahlregister (221; 321) befindliche Zahl ermittelt wird und dem jeweils nächsten Zeiger ($j_1 \dots j_G$) zugewiesen oder zu diesem hinzugezählt oder von diesem abgezogen wird, und der Wert des jeweils nächsten Zeigers ($j_1 \dots j_G$) zur Adressierung einer Speicherposition im jeweils nächsten Auswahlregister (221; 321) herangezogen wird,

- d) dass eine Anzahl von Grund-Transformationsvorschriften ($f_1...f_A$; $f_1^{-1}...f_A^{-1}$) vorgegeben wird,
 - e) dass der vom letzten Zeigerwert (j_G) der Adressiereinheit (222-G ; 322-G) adressierte Wert (ϕ_G) des letzten Auswahlregisters (221-G; 321-G) zur Auswahl der Grund- Transformationsvorschriften ($f_1...f_A$; $f_1^{-1}...f_A^{-1}$) herangezogen wird,
 - f) dass gegebenenfalls zur Auswahl einer Mehrzahl von Grund-Transformationsvorschriften ($f_1...f_A$; $f_1^{-1}...f_A^{-1}$) jeweils der erste Zeiger (j_1) nach vorgegebenen Kriterien abgeändert, insbesondere erhöht, verringert, inkrementiert oder dekrementiert, wird und die Schritte a) bis e) erneut durchgeführt werden, und
 - g) ein Datenwert, insbesondere ein zu verschlüsselndes Byte (b_i) oder ein zu entschlüsselndes Byte (c_i), angegeben wird und die ausgewählte Transformationsvorschrift (g_i ; g_i^{-1}) auf den Datenwert angewendet wird, gegebenenfalls die mehreren ausgewählten Grund- Transformationsvorschriften ($f_{i,1}...f_{i,T}$; $f_{i,1}^{-1}...f_{i,T}^{-1}$) hintereinander, im Falle der Verschlüsselung in der Reihenfolge ihrer Auswahl oder im Falle der Entschlüsselung entgegen der Reihenfolge ihrer Auswahl, auf den Datenwert angewendet werden und das Ergebnis der Anwendung zur Verfügung gehalten wird.
20. Verfahren nach Anspruch 19, **dadurch gekennzeichnet**, dass abhängig von am Eingang (224; 324) anliegenden Werten, insbesondere dem zu verschlüsselnden Byte und/oder dem verschlüsselten Byte und/oder von Zufallszahlen (p), der Wert eines der Adressiereinheiten (222; 322) abgeändert wird,
- wobei diese Werte mit dem in der jeweiligen Adressiereinheit (222; 322) befindlichen Wert verknüpft werden, vorzugsweise das Resultat dann modulo A genommen und um eins erhöht wird und der Wert dann der jeweiligen Adressiereinheit (222; 322) zugewiesen wird,
 - wobei gegebenenfalls nach Vorliegen eines externen Triggersignals, insbesondere ausgelöst durch eine vorzunehmende Verschlüsselung oder Entschlüsselung eines oder einer vorgegebenen Anzahl von Bytes, jeweils die nächste Adressiereinheit abgeändert wird und für den Fall, dass im vorherigen Schritt die letzte Adressiereinheit (222-G; 322-G) abgeändert wurde, wiederum die erste Adressiereinheit (222-1; 322-1) abgeändert wird.
21. Verfahren nach Anspruch 19 oder 20, **dadurch gekennzeichnet**, dass abhängig von am Eingang (224; 324) anliegenden Werten (b_i , c_i , p), insbesondere einem zu verschlüsselnden Byte (b_i) und/oder einem verschlüsselten Byte (c_i) und/oder Zufallszahlen (p) eines der Auswahlregister (221; 321) abgeändert wird, und insbesondere zwei in den Auswahlregistern (221; 321) abgespeicherte Werte miteinander vertauscht werden,
- wobei der Index des einen Wertes eine Reihe von vorgegebenen Werten nach einer vorab vorgegebenen Reihenfolge durchläuft, insbesondere durch zyklisches Hochzählen zwischen 1 und Z, und der Index des zweiten Wertes durch die am Eingang (234; 334) anliegenden Werte festgelegt wird,
 - wobei gegebenenfalls nach Vorliegen eines externen Triggersignals, insbesondere ausgelöst durch eine vorzunehmende Verschlüsselung oder Entschlüsselung eines Bytes oder einer vorgegebenen Anzahl von Bytes, nach jeder Vertauschung oder nach einer vorgegebenen Anzahl von Vertauschungen in einem Auswahlregister (221; 321) zum jeweils nächsten Auswahlregister (221; 321) gewechselt wird und für den Fall, dass im vorherigen Schritt in dem letzten Auswahlregister (221-G; 321-G) bereits die vorgegebene Anzahl von Vertauschungen statt gefunden hat, wieder das erste Auswahlregister (221-1; 321-1) für Abänderungen herangezogen wird.
22. Verfahren nach Anspruch 1 bis 14 und 18 bis 21, **dadurch gekennzeichnet**, dass abhängig von am Eingang (224; 324) anliegenden Werten (b_i , c_i , p), insbesondere einem zu verschlüsselnden Byte (b_i) und/oder einem verschlüsselten Byte (c_i) und/oder Zufallszahlen (p) die Transformationsvorschriften (g_i , g_i^{-1}) oder die Grund-Transformationsvorschriften (f_i ; f_i^{-1}) modifiziert werden,

- wobei die Transformationsvorschriften (g_i ; f_i) zu den Transformationsvorschriften (g_i^{-1} ; f_i^{-1}) invers sind,
 - wobei nach einer vorgegebenen Anzahl von Abänderungen in einer Transformationsvorschrift oder Grund-Transformationsvorschrift (f), zur jeweils nächsten Transformationsvorschrift oder Grund-Transformationsvorschrift (f), gewechselt wird und für den Fall, dass im vorherigen Schritt die letzte Transformationsvorschrift oder Grund-Transformationsvorschrift (f_A), abgeändert wurde, wieder die erste Transformationsvorschrift oder Grund-Transformationsvorschrift, für Abänderungen herangezogen wird.
23. Verfahren nach einem der Ansprüche 1 bis 14 oder 18, **dadurch gekennzeichnet**, dass die Anwendung der Transformationsvorschriften (g_i , g_i^{-1}) oder Grund-Transformationsvorschriften (f_i , f_i^{-1}) bei der Verschlüsselung eines Byte der Rohdaten oder der Entschlüsselung eines verschlüsselten Byte gemäß mindestens einem der Ansprüche 19 bis 23 vorgenommen wird.
24. Verschlüsselungsmaschine (200) zur Verschlüsselung von Rohdaten, insbesondere zur Durchführung eines Verfahrens nach einem der Ansprüche 1 bis 7, umfassend
- einen Datenspeicher (211) zur Aufnahme von Rohdaten mit einer Anzahl N Bytes umfassend jeweils eine vorgegebene Anzahl von B Bits,
 - dass der Zufallszahlengenerator (212) gegebenenfalls Zufallszahlen erstellt, die für die Auswahl der Transformationsvorschriften verwendet werden, und
 - eine Transformationseinheit (213), die abhängig von einem, durch ein Passwort (W_0) vorgegebenen inneren Zustand (Y) gegebenenfalls mit Hilfe von durch den Zufallszahlengenerator (212) bereitgestellten Zufallszahlen eine, insbesondere für jedes Byte der Rohdaten andere, Verschlüsselungsart bestimmt und durchführt
 - einen von einem Passwort (W_0) initialisierten deterministischen Zufallszahlengenerator (212), der bei der Erstellung der Zufallszahlen auf einen inneren Zustand (Y) zugreift,
 - einen Zustandsspeicher zur Abspeicherung des inneren Zustands der Transformationseinheit (213) und des Zufallszahlengenerators (212),
 - eine Steuereinheit zur Ansteuerung der Transformationseinheit (213) zur separaten Verschlüsselung der einzelnen Bytes der Rohdaten im Datenspeicher (211),
 - eine Initialisierungseinheit zum Setzen des Zustandsspeichers auf einen von einem Passwort (W_0) abhängigen Initialzustand (Y_0),
- gekennzeichnet durch**
- eine Aktualisierungseinheit, die den Zustandsspeicher der Transformationseinheit (213) nach einer Anzahl von durchgeführten Verschlüsselungen, insbesondere nach der Verschlüsselung jedes einzelnen Bytes, abhängig vom jeweils verschlüsselten Byte (b_i) sowie vom jeweiligen Endergebnis der Verschlüsselung (c_i) und/oder abhängig von vom Zufallszahlengenerator erstellten Zufallszahlen (p) ändert und aktualisiert, und die gegebenenfalls den Zustandsspeicher des Pseudo-Zufallszahlengenerators (212) nach einer Anzahl von durchgeführten Verschlüsselungen, insbesondere nach der Verschlüsselung jedes einzelnen Bytes, abhängig vom jeweils zu verschlüsselnden Byte (b_i) sowie vom jeweiligen Endergebnis der Verschlüsselung (c_i) und abhängig von vom Zufallszahlengenerator vorher erstellten Zufallszahlen (p) ändert und aktualisiert.
25. Entschlüsselungsmaschine (300) zur Entschlüsselung von verschlüsselten Daten, insbesondere zur Durchführung eines Verfahrens nach einem der Ansprüche 8 bis 14, umfassend
- einen Datenspeicher (321) zur Aufnahme von verschlüsselten Daten mit einer Anzahl N Bytes umfassend jeweils eine vorgegebene Anzahl von B Bits,
 - eine Transformationseinheit (313), die abhängig von einem durch ein Passwort (W_0) vorgegebenen inneren Zustand (Y) eine Entschlüsselungsart bestimmt und durchführt,
 - einen von einem Passwort (W_0) initialisierten deterministischen Zufallszahlengenerator (312), der bei der Erstellung von Zufallszahlen auf einen inneren Zustand (Y) zugreift,
 - einen Zustandsspeicher zur Abspeicherung des inneren Zustandes der Transformationseinheit (313) und des Zufallszahlengenerators (312),

- eine Steuereinheit zur Ansteuerung der Transformationseinheit (313) zur separaten Entschlüsselung der einzelnen Bytes der Rohdaten im Datenspeicher (311),
- eine Initialisierungseinheit zum Setzen des Zustandsspeichers auf einen von einem Passwort (W_0) abhängigen Initialzustand,

gekennzeichnet durch

eine Aktualisierungseinheit, die den Zustandsspeicher der Transformationseinheit (313) nach einer Anzahl von durchgeführten Entschlüsselungen, insbesondere nach der Entschlüsselung jedes einzelnen Bytes, abhängig vom jeweils zu entschlüsselnden Byte (c_i) sowie vom jeweiligen Endergebnis der Entschlüsselung (b_i) und/oder abhängig von vom Zufallszahlengenerator erstellten Zufallszahlen (ρ) ändert und aktualisiert, und die gegebenenfalls den Zustandsspeicher des Pseudo-Zufallszahlengenerators (212) nach einer Anzahl von durchgeführten Entschlüsselungen, insbesondere nach der Entschlüsselung jedes einzelnen Bytes, abhängig vom jeweils zu entschlüsselnden Byte (c_i) sowie vom jeweiligen Endergebnis der Entschlüsselung (b_i) und abhängig von vom Zufallszahlengenerator vorher erstellten Zufallszahlen (ρ) ändert und aktualisiert.

26. Datenträger, auf dem ein Computerprogramm zur Durchführung eines Verfahrens nach einem der Ansprüche 1 bis 23 abgespeichert ist.
27. Datenträger, auf dem verschlüsselte Daten, erzeugt nach einem der Ansprüche 1 bis 7, 18, 22 und 23 abgespeichert sind.

Hierzu 10 Blatt Zeichnungen

1/10

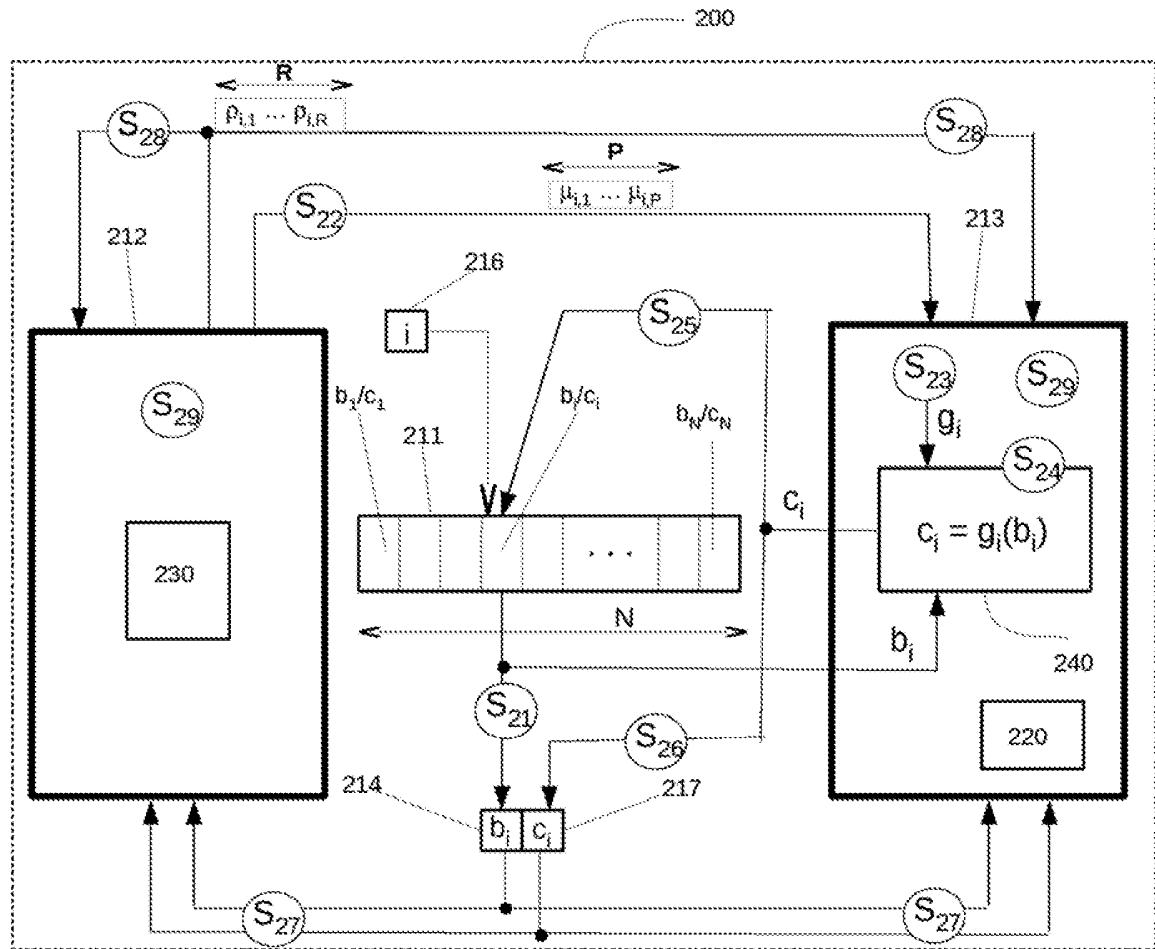


Fig. 1

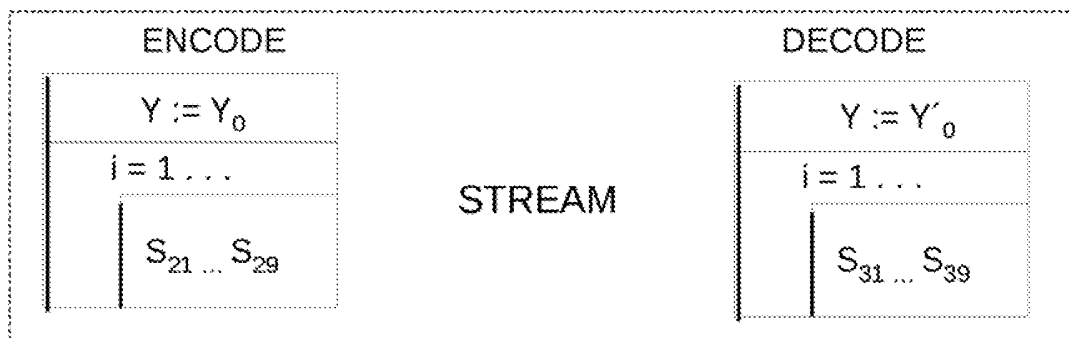


Fig. 2

2/10

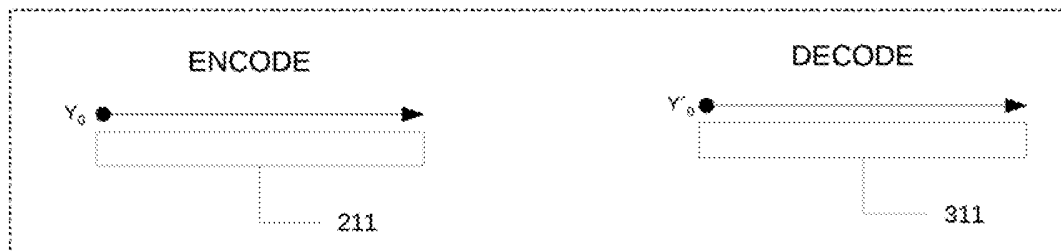


Fig. 3

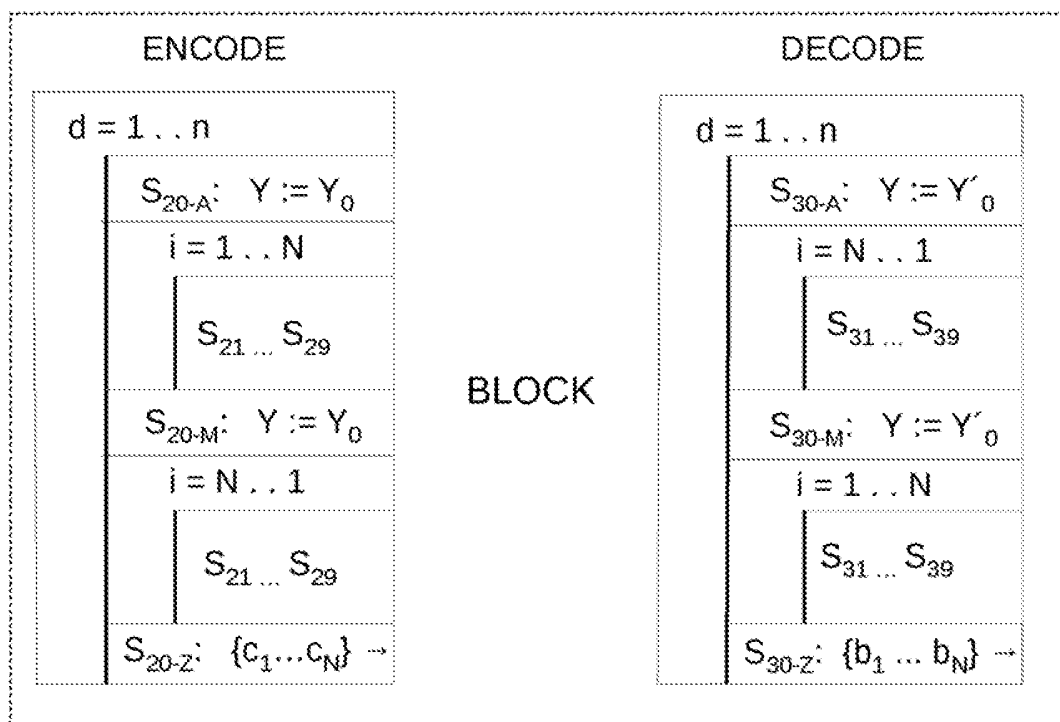


Fig. 4

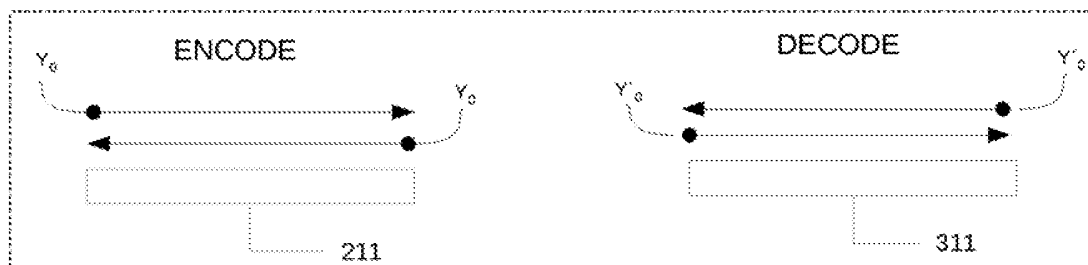


Fig. 5

3/10

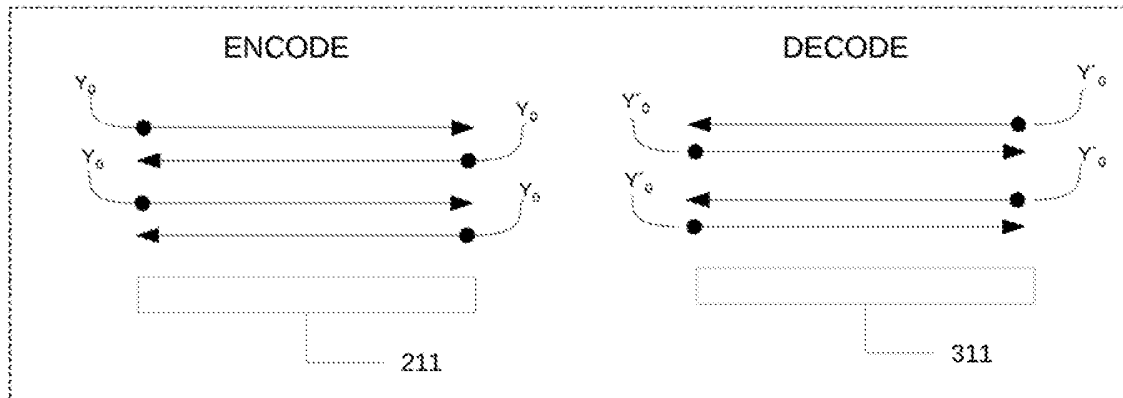


Fig. 6

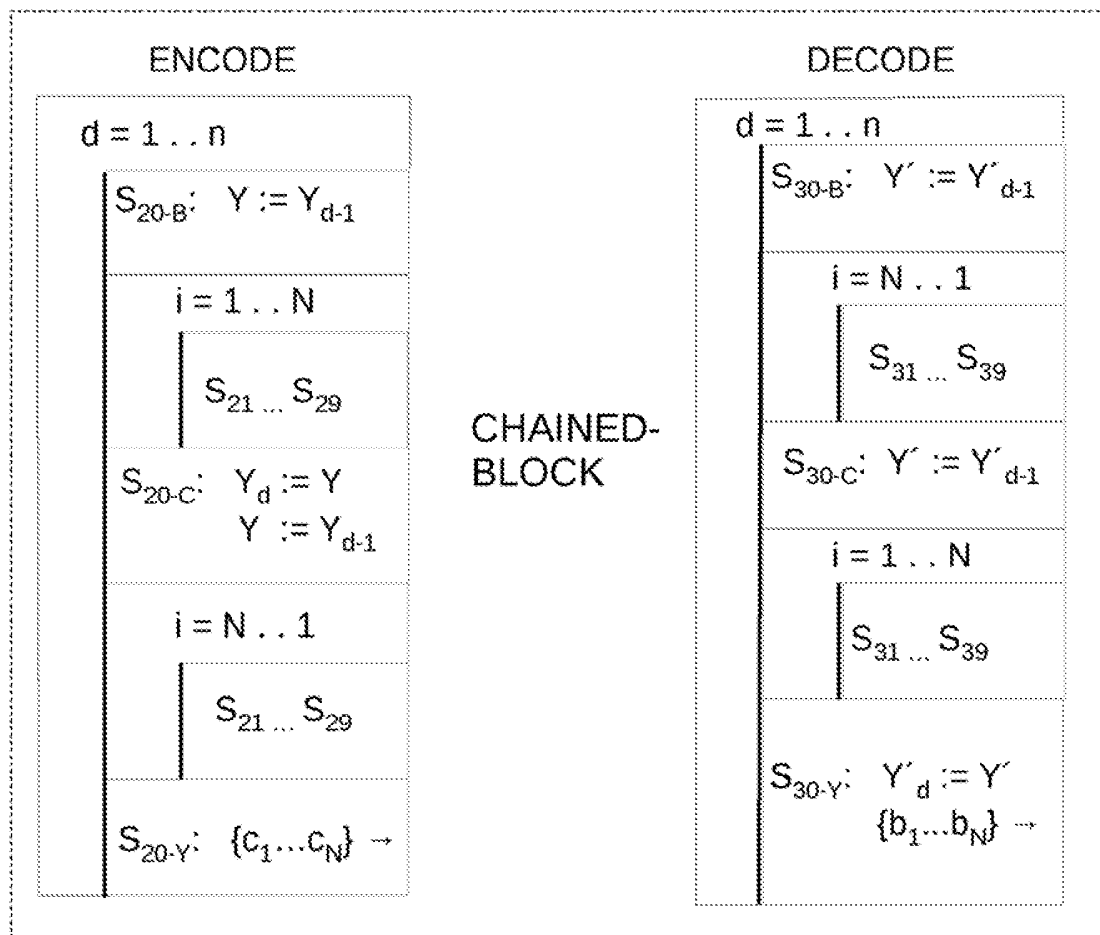


Fig. 7

4/10

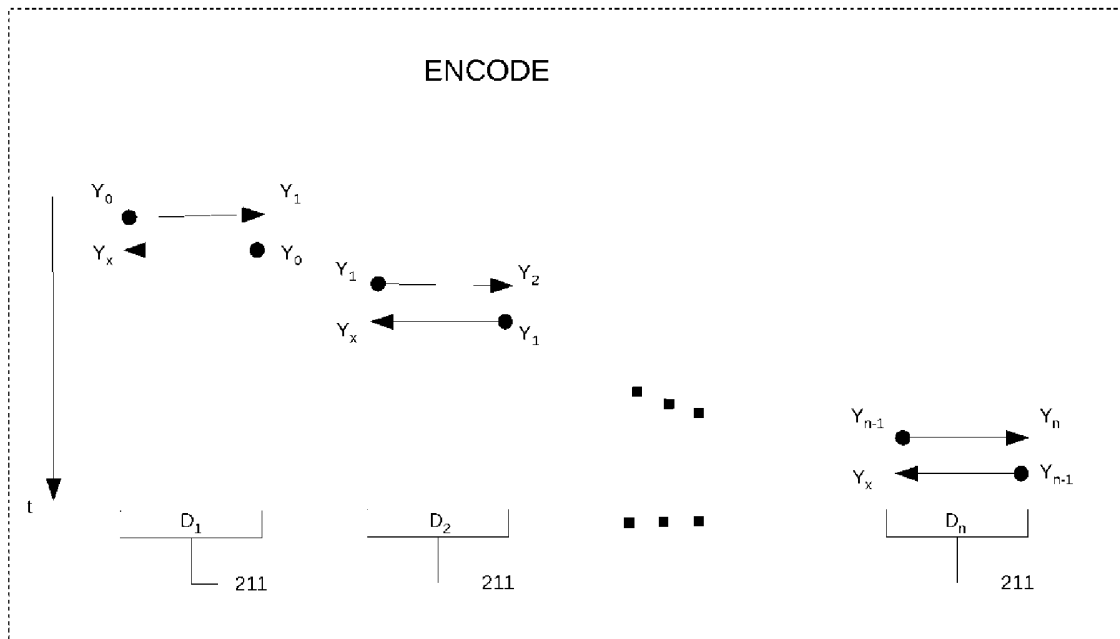


Fig. 8

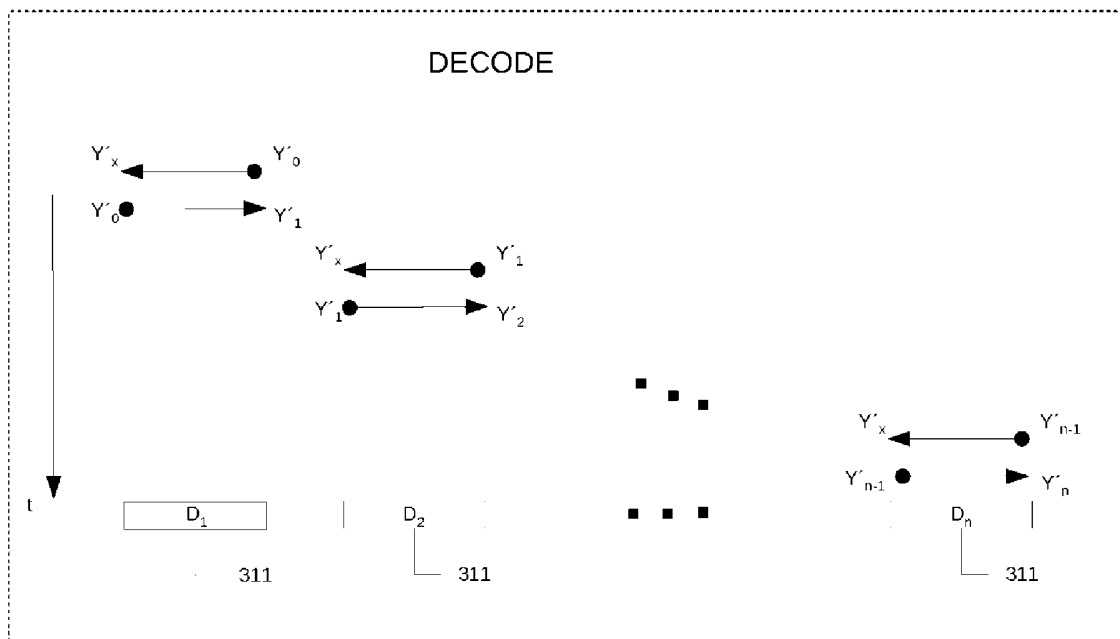


Fig. 9

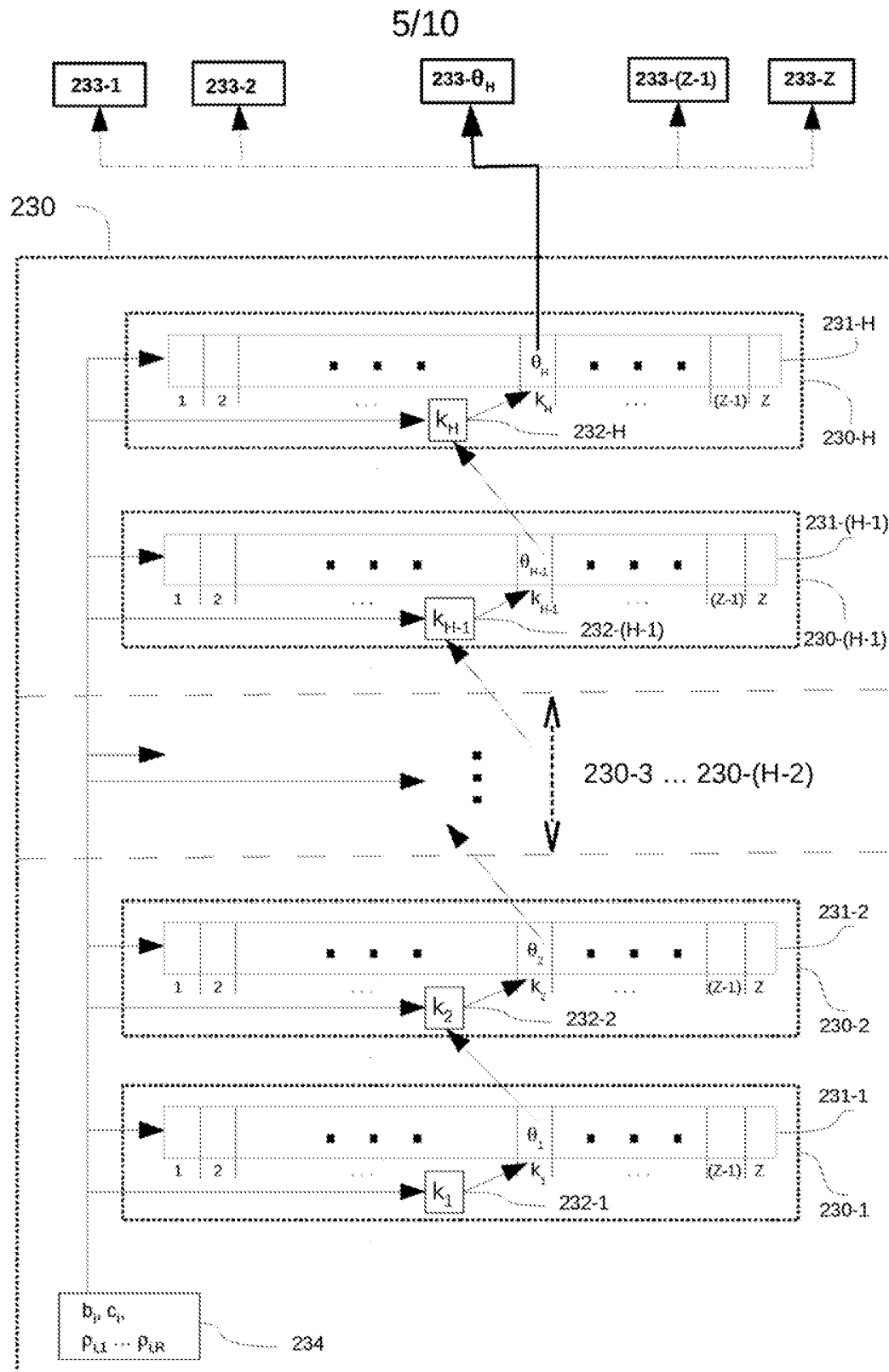


Fig. 10

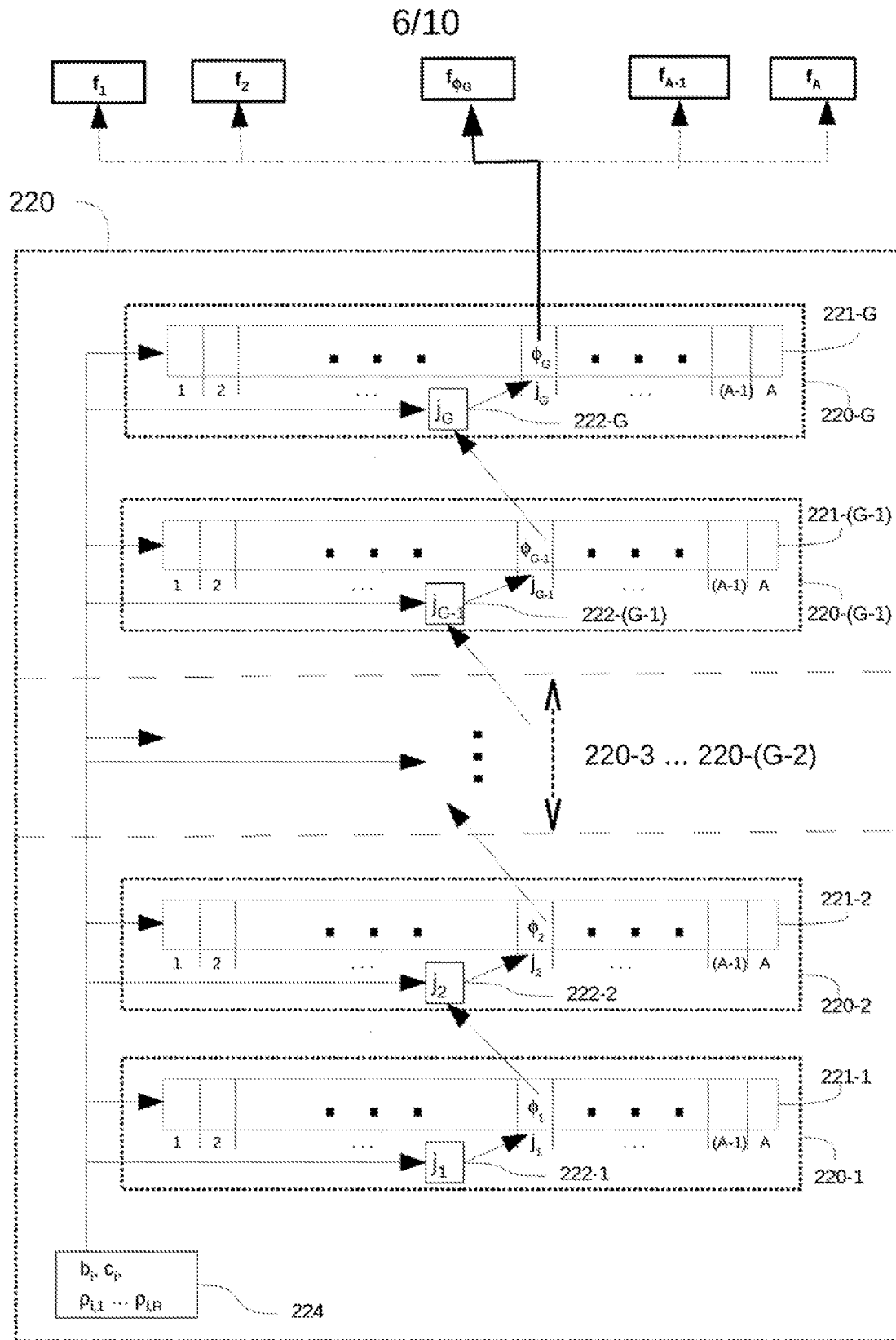


Fig. 11

7/10

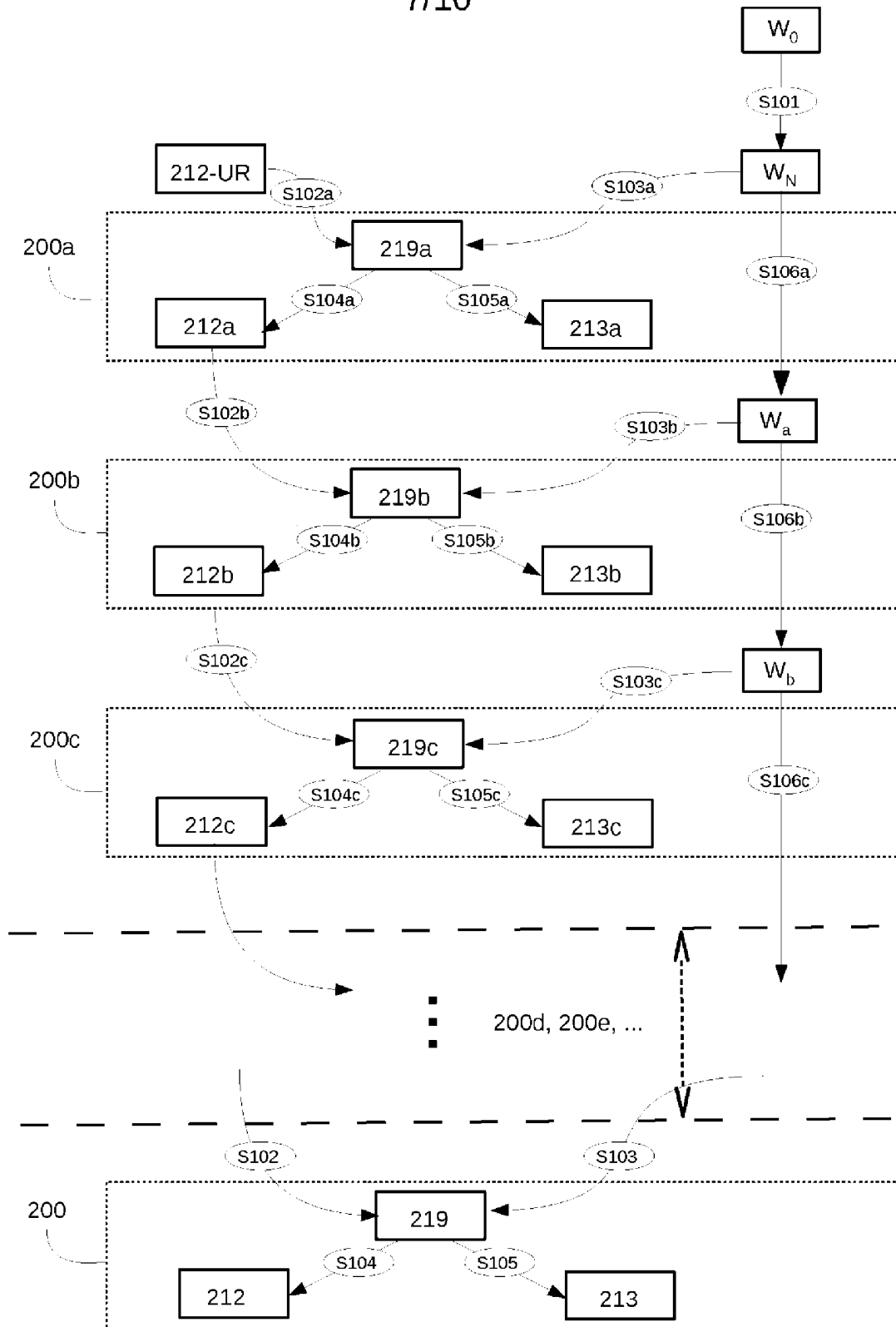


Fig. 12

8/10

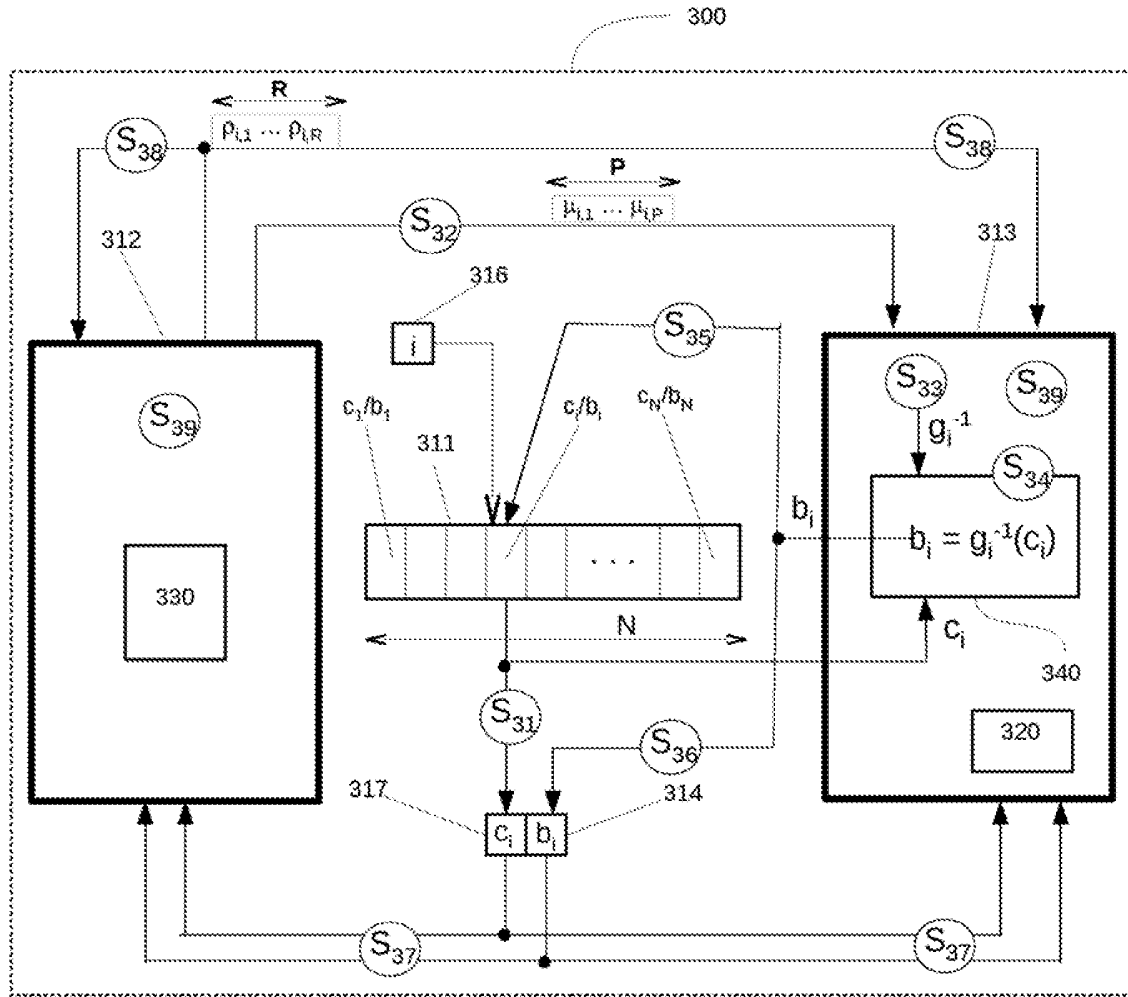


Fig. 13

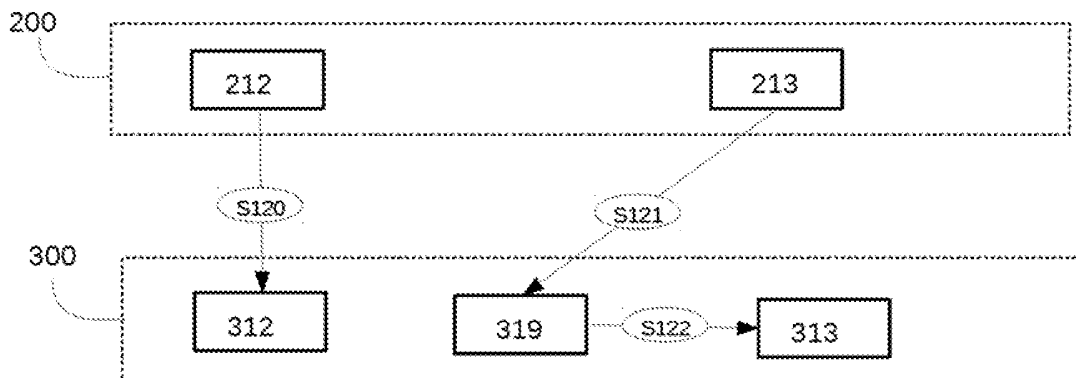


Fig. 16

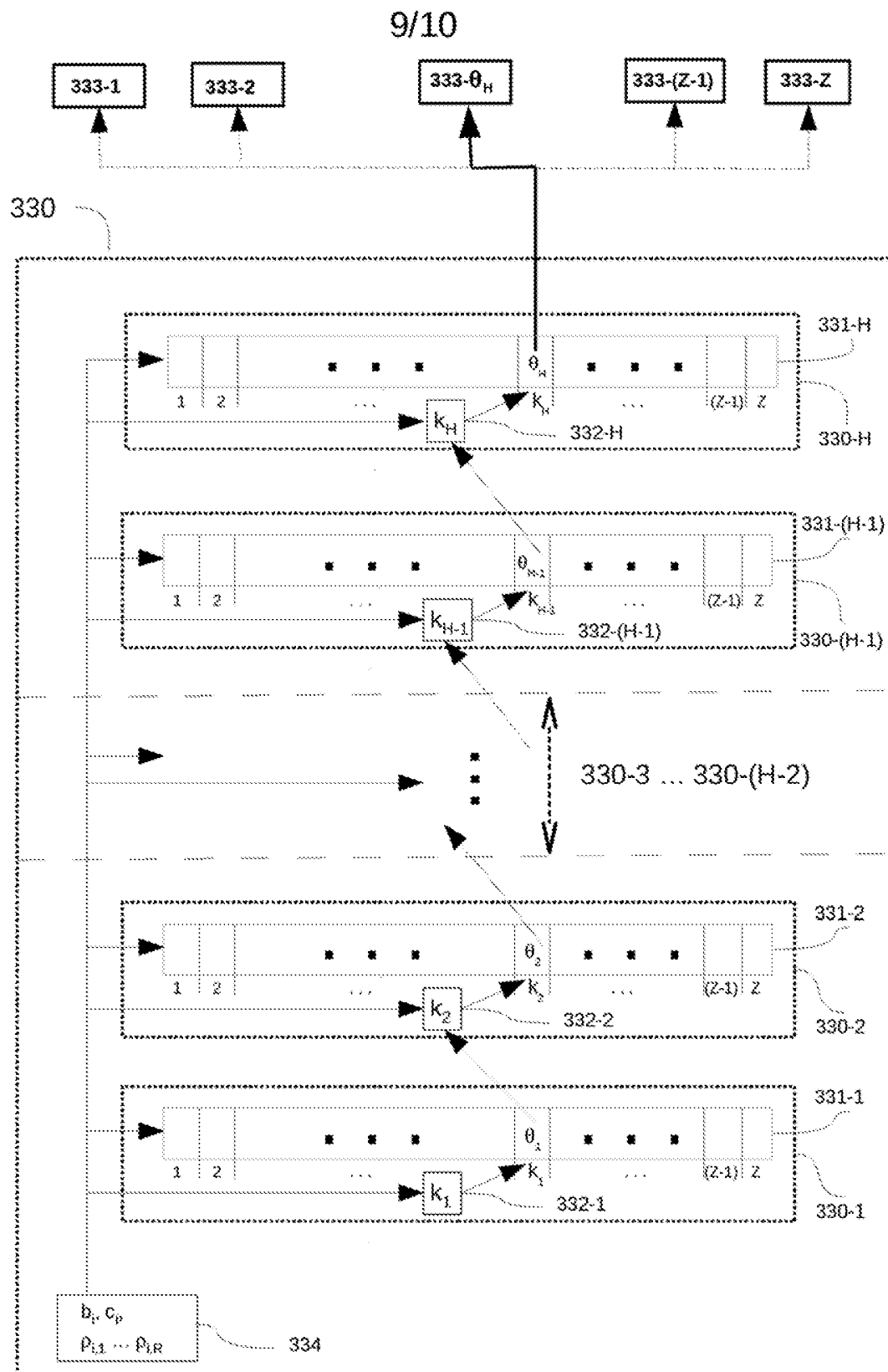


Fig. 14

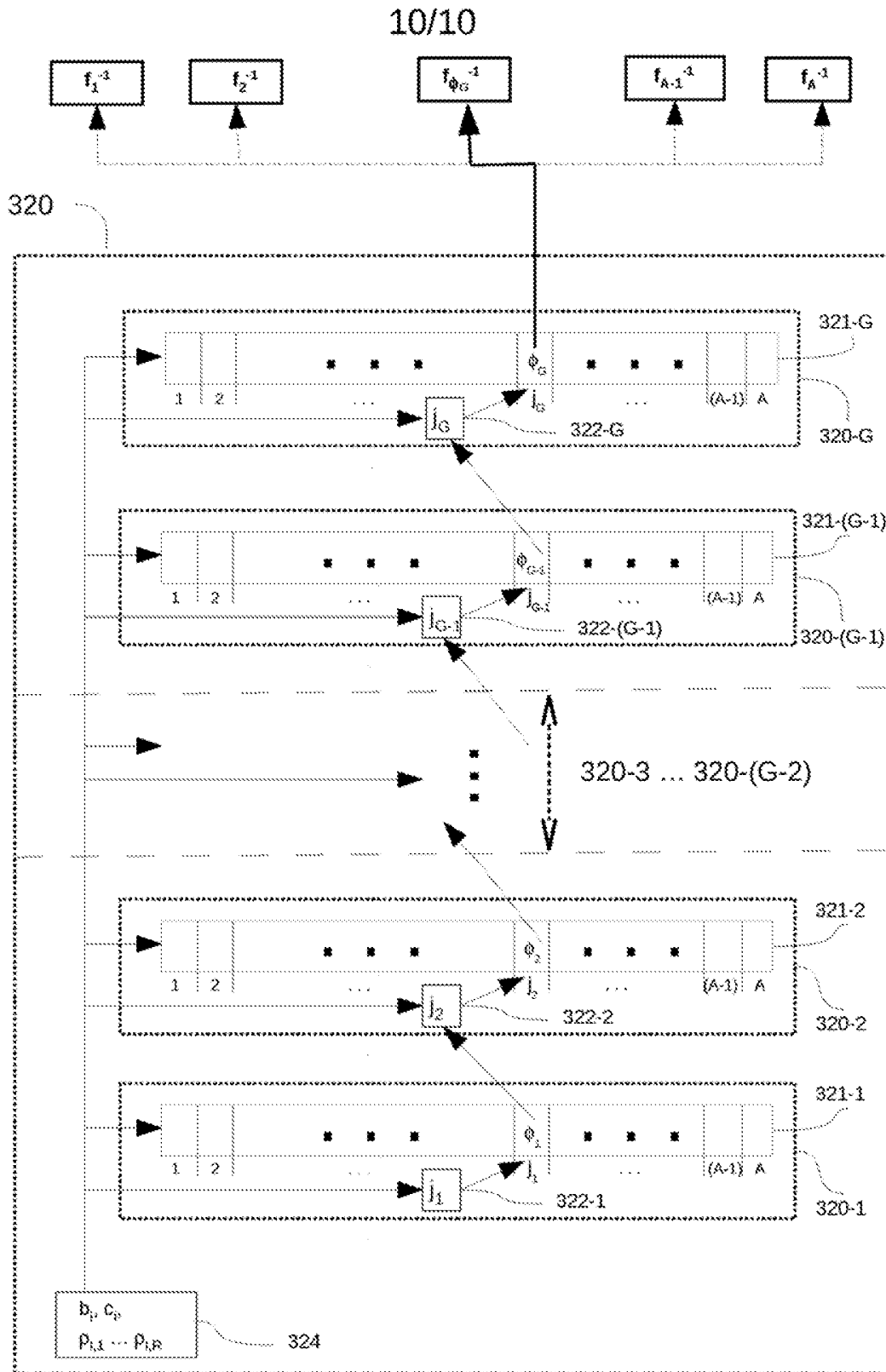


Fig. 15