(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4857230号
(P4857230)
(45)発行日　平成24年1月18日(2012.1.18)　(24)登録日　平成23年11月4日(2011.11.4)

(51)Int.Cl.　F I
G06F 7/58 (2006.01)　G06F 7/58 B
G09C 1/00 (2006.01)　G09C 1/00 650B

請求項の数 14　(全 18 頁)

(21)出願番号　特願2007-245710 (P2007-245710)
(22)出願日　平成19年9月21日 (2007.9.21)
(65)公開番号　特開2008-276728 (P2008-276728A)
(43)公開日　平成20年11月13日 (2008.11.13)
審査請求日　平成21年11月20日 (2009.11.20)
(31)優先権主張番号　特願2007-90214 (P2007-90214)
(32)優先日　平成19年3月30日 (2007.3.30)
(33)優先権主張国　日本国(JP)

（出願人による申告）国等の委託研究の成果に係る特許出願（平成１９年度　独立行政法人情報通信研究機構「大容量データの安全な流通・保存技術に関する研究開発」　委託研究、産業技術力強化法第１９条の適用を受ける特許出願）

(73)特許権者　000005108
株式会社日立製作所
東京都千代田区丸の内一丁目６番６号
(74)代理人　110000350
ポレール特許業務法人
(72)発明者　渡辺　大
神奈川県川崎市麻生区王禅寺１０９９番地
株式会社日立製作所システム開発研究所内
(72)発明者　吉田　博隆
神奈川県川崎市麻生区王禅寺１０９９番地
株式会社日立製作所システム開発研究所内

審査官　田中　友章

最終頁に続く

(54)【発明の名称】疑似乱数生成装置及びそれを用いた暗号化処理装置

(57)【特許請求の範囲】
【請求項１】
2ブロック(１ブロックはnビット)の容量のステート記憶部、
複数ブロックの容量のバッファ、
クロック入力に応答して、前記バッファの記憶内容を入力し、該入力データと同サイズのデータを出力する非線形変換部、
前記ステート記憶部の内容と前記非線形変換部の出力とを入力し、出力を前記ステート記憶部に格納する第1の線形変換部、
前記クロック入力に応答して、前記バッファの記憶内容と前記ステート記憶部の記憶内容とを入力し、出力を前記バッファに格納する第2の線形変換部、及び
前記ステート記憶部の記憶内容を乱数列として出力する出力部
を有する疑似乱数生成装置。
【請求項２】
n(K)ビットの秘密鍵を入力し、前記バッファの初期状態を定める初期化ユニットを有し、
前記バッファの前記複数ブロックの容量は前記n(K)ビットの少なくとも2倍の容量である
請求項１に記載の疑似乱数生成装置。
【請求項３】
前記非線形変換部への入力が4ブロックである請求項２に記載の疑似乱数生成装置。
【請求項４】

前記非線形変換部に入力する前記バッファのブロック位置をi1，i2，i3,及びi4とするとき、i2－i1，i3－i2,及びi4－i3が互いに異なる請求項３に記載の疑似乱数生成装置。

【請求項５】

さらに、i4－i1は、i2－i1，i3－i2,及びi4－i3のいずれとも異なる請求項４に記載の疑似乱数生成装置。

【請求項６】

前記第2の線形変換部は以下の処理を実行する請求項４に記載の疑似乱数生成装置。

Y(i) ← B(i)　(0≦i ＜ M),
Y(j1) ← Y(j1) XOR Y(k1),
Y(j2) ← Y(j2) XOR Y(k2),
Y(j3) ← Y(j3) XOR Y(k3),
Y(M－1) ← Y(M－1) XOR A(0),
W(i+1) ← Y(i)　(0 ≦i＜M－1),
W(0) ← Y(M－1)

ただし、A(0)は前記ステート記憶部の上位ブロック、B(i)は前記バッファの第i番目のブロック、Mは前記バッファを構成するブロック数、W(i)は前記第二の線形変換部の出力、XORはブロック単位の排他的論理和処理、及び←はデータの代入をそれぞれ表し、i1≦ j1＜k1＜ i2 ≦ j2＜ k2＜ i3≦ j3＜ k3＜ i4である。

【請求項７】

前記排他的論理和処理を定めるパラメータj1，j2，j3，k1，k2,及びk3は、k1－j1，k2－j2，k3－j3が互いに異なる請求項６に記載の疑似乱数生成装置。

【請求項８】

前記非線形変換部は、さらに1ブロックを入力し1ブロックを出力する非線形置換部を備え、以下の処理を実行する請求項４に記載の疑似乱数生成装置。

X(1) ← S[B(i1)],
X(2) ← S[B(i2)],
X(3) ← S[B(i3)],
X(4) ← S[B(i4)]

ただし、B(i)は前記バッファの第i番目のブロック、X(i)は前記非線形変換部の出力ブロック、S[Y]は前記非線形置換部による置換処理、及び←はデータの代入をそれぞれ表す。

【請求項９】

前記第1の線形変換部は、さらに有限体上定義される乗算器をさらに備え、以下の処理を実行する請求項８に記載の疑似乱数生成装置。

P(0) ← A(0) XOR X(1),
P(1) ← A(1) XOR X(2),
Q(0) ← P(0) XOR mul(C1, P(1)),
Q(1) ← P(1) XOR mul(C2, P(0)),
R(0) ← Q(0) XOR X(3),
R(1) ← Q(1) XOR X(4)

ただし、A(i)は前記ステート記憶部の第i番目のブロック、X(i)は前記非線形変換部の出力する第i番目のブロック、R(i)は前記第一の線形変換部の出力ブロック、Ciは1ブロックサイズの0でない定数であって、C1及びC2のいずれか一方は１ではない定数、mul(x，y)は前記乗算器による乗算処理、及び←はデータの代入をそれぞれ表す。

【請求項１０】

前記初期化ユニットは、外部入力として前記秘密鍵と共に乱数列番号を入力し、前記外部入力をワード単位のデータブロックに分割し、前記分割されたデータブロックを前記バッファに上位ワードから順に入力し、前記外部入力が入力されなかった前記バッファの下位ワードおよび前記ステート記憶部に規定の定数値を設定する、請求項２に記載の疑似乱数生成装置。

【請求項１１】

前記初期化ユニットは、さらに、前記分割された各データブロックのデータを前記バッファの特定のワードと排他的論理和し、前記バッファの値を更新し、前記非線形変換部、前記第1の線形変換部及び前記第2の線形変換部を動作させて前記バッファおよび前記ステート記憶部の内部状態を更新する請求項10に記載の疑似乱数生成装置。

【請求項１２】

前記出力部は前記ステート記憶部の下位1ワードを部分乱数列として出力する請求項１に記載の疑似乱数生成装置。

【請求項１３】

2ブロック(１ブロックはnビット)の容量のステート記憶部、
複数ブロックの容量のバッファ、
秘密鍵と乱数列番号とをワード単位のデータブロックに分割し、前記分割されたデータブロックを前記バッファに上位ワードから順に入力し、前記バッファの下位ワードおよび前記ステート記憶部に規定の定数値を設定する初期化ユニット、
クロック入力に応答して、前記バッファの記憶内容を入力し、該入力データと同サイズのデータを出力する非線形変換部、
前記ステート記憶部の内容と前記非線形変換部の出力とを入力し、出力を前記ステート記憶部に格納する第1の線形変換部、
前記クロック入力に応答して、前記バッファの記憶内容と前記ステート記憶部の記憶内容とを入力し、出力を前記バッファに格納する第2の線形変換部、及び
前記ステート記憶部のデータを出力する出力部を有する疑似乱数生成装置、並びに
データ列と前記出力部の出力データとをビットごとに排他的論理和をとる排他的論理和装置を有する暗号化／復号装置。

【請求項１４】

（A）2ブロック(１ブロックはnビット)の容量の第１のステート記憶部、複数ブロックの容量の第１のバッファ、秘密鍵と乱数列番号とをワード単位のデータブロックに分割し、前記分割されたデータブロックを前記バッファに上位ワードから順に入力し、前記バッファの下位ワードおよび前記ステート記憶部に規定の定数値を設定する第１の初期化ユニット、第１のクロック入力に応答して、前記第１のバッファの記憶内容を入力し、該入力データと同サイズのデータを出力する第１の非線形変換部、前記第１のステート記憶部の内容と前記第１の非線形変換部の出力とを入力し、出力を前記第１のステート記憶部に格納する第1の線形変換部、前記第１のクロック入力に応答して、前記第１のバッファの記憶内容と前記第１のステート記憶部の記憶内容とを入力し、出力を前記第１のバッファに格納する第2の線形変換部、及び前記第１のステート記憶部のデータを出力する第1の出力部を含む第1の疑似乱数生成装置、コンテンツデータを格納する記憶装置、及び前記コンテンツデータと前記第1の出力部の出力データとをビットごとに排他的論理和をとる第１の排他的論理和装置を有し、暗号化したコンテンツデータを出力する暗号化装置を有するサーバと、
（B）ネットワークと、
（C）前記ネットワークを介して前記サーバに接続し、2ブロックの容量の第２のステート記憶部、複数ブロックの容量の第２のバッファ、前記秘密鍵と乱数列番号とをワード単位のデータブロックに分割し、前記分割されたデータブロックを前記バッファに上位ワードから順に入力し、前記バッファの下位ワードおよび前記ステート記憶部に規定の定数値を設定する第２の初期化ユニット、第２のクロック入力に応答して、前記第２のバッファの記憶内容を入力し、該入力データと同サイズのデータを出力する第２の非線形変換部、前記第２のステート記憶部の内容と前記第２の非線形変換部の出力とを入力し、出力を前記第２のステート記憶部に格納する第3の線形変換部、前記第２のクロック入力に応答して、前記第２のバッファの記憶内容と前記第２のステート記憶部の記憶内容とを入力し、出力を前記第２のバッファに格納する第4の線形変換部、及び前記第２のステート記憶部のデータを出力する第２の出力部を含む第2の疑似乱数生成装置、及び前記ネットワークを介して前記サーバから入力した前記暗号化したコンテンツデータと前記第2の出力部の出

力データとをビットごとに排他的論理和をとる第2の排他的論理和装置を含み、前記コンテンツデータを再生するする復号化装置を有する端末装置とを備えたデータ配信システム。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は、実用的な乱数列を生成する装置とその応用装置に関する。

【背景技術】

【０００２】

公開鍵暗号技術を利用した署名生成、秘密通信を行う際の鍵の生成、およびストリーム暗号(stream cipher)技術などにおいて、乱数の必要性は高い。

【０００３】

しかし、これらの場面において真性乱数を用いようとすることは非現実的であり、実際には疑似乱数生成方法またはそれを用いた装置により生成される疑似乱数(以下、単に乱数ともいう)が用いられる。

【０００４】

暗号での使用に関して疑似乱数に要求される条件として、予測不可能性や、乱数を決定するための初期値を生成した乱数から導出できないこと、という安全性に関する性質がある。さらに疑似乱数生成方法または生成装置が実用に耐えるためにはソフトウェア実装またはハードウェア実装において高速な処理が求められる。さらに、ハードウェア実装する場合の必要ゲート数、およびソフトウェア実装した場合のステップ数や実行時の必要メモリ領域などが小さい、といった実装コストの面からも効率的である必要がある。

【０００５】

汎用的な暗号アルゴリズムとして、これらの評価項目全てを高いレベルで満たすものが好ましい。

【０００６】

ソフトウェア、ハードウェアいずれのプラットフォームにおいても実装が可能な疑似乱数生成技術は、たとえば、特許文献１に開示されている。特許文献１に記載の疑似乱数生成技術は、小型の非線形置換と有限体上で定義される最大距離分離符号行列とを組み合わせて大型で高密度な置換を構成することでソフトウェア実装における処理の効率化と高い安全性を実現している。

【０００７】

【特許文献１】特許第3724399号公報

【発明の開示】

【発明が解決しようとする課題】

【０００８】

特許文献１に記載の疑似乱数生成技術では、高い安全性を実現するために大きな内部状態を持ち、ハードウェア実装では回路規模が大きくなる。また、上記の大型の高密度置換をソフトウェアで実装する場合には、大きなメモリ容量を必要とする。このため、特許文献１に記載の疑似乱数生成方法を実装する際は、その小型化は制限を受ける。

【０００９】

したがって、小型の電子機器での利用に適した、より小さい回路規模での実装やより小容量のメモリでの実装が可能な疑似乱数生成技術が望まれている。

【課題を解決するための手段】

【００１０】

本発明の態様は次のとおりである。2ブロック(１ブロックはnビット)の容量のステート記憶部と複数ブロックの容量のバッファとを有し、これらの内容を攪拌し、乱数列を得る疑似乱数生成装置である。

【００１１】

クロック入力に応答して、バッファの記憶内容を入力し、この入力データと同サイズの

データを出力する非線形変換部と、ステート記憶部の内容と非線形変換部の出力とを入力し、出力をステート記憶部に格納する第1の線形変換部、クロック入力に応答して、バッファの記憶内容とステート記憶部の記憶内容とを入力し、出力をバッファに格納する第2の線形変換部とにより攪拌する。攪拌したステート記憶部の内容を乱数列として出力する。

【００１２】

本発明の疑似乱数生成装置の実現に当たっては、n(K)ビットの秘密鍵を入力し、バッファの初期状態を定める初期化ユニットを有し、バッファの容量はn(K)ビットの少なくとも2倍の容量であることが望ましい。

【００１３】

本発明の疑似乱数生成装置は、その出力の乱数列を用いて、データ列を暗号化及び/又は複合化する暗号化／複合化装置に適用される。さらに、その暗号化／複合化装置を、サーバ及び端末に応用したデータ配信システムにも適用が広がる。

【発明の効果】

【００１４】

本発明によれば、線形変換を小型化でき、装置全体として小型化した疑似乱数生成装置を提供できる。

【発明を実施するための最良の形態】

【００１５】

(用語の説明)

・疑似乱数生成装置(pseudorandom number generator)：乱数列(random number sequence)を決定するための初期値(initial value)を与えて、疑似乱数列(pseudorandom number sequence)を生成する装置。

・疑似乱数(pseudorandom number)：有限、もしくは無限のビット列であり、どのような方法でも真性乱数と区別することができないもの。

・真性乱数：無限ビット列であり、任意の連続する部分列が与えられても、次の1ビットを推定することができないような列のこと。

・共通鍵暗号(symmetric-key encryption)：暗号化処理と復号化処理に同じ鍵を用いる暗号化技術。

・鍵：暗号化処理の際に用いる秘匿パラメータ。

・平文(plaintext)：暗号化処理前、または復号化処理後のデータであり、ディジタル化されたマルチメディアデータも含まれる。

・暗号文(ciphertext)：暗号化処理されたデータ。

・ブロック暗号：入力データを一定長のデータごとに区切り(区切られた一定長データをブロックという)、鍵と共に攪拌処理を行うことで暗号化処理(encryption)または復号化処理(decryption)を行う暗号技術。

・ストリーム暗号：疑似乱数生成装置に乱数列を決定する情報を与えて乱数列を生成し、この乱数列と平文を攪拌することで暗号文を生成する暗号技術。

・非線形変換：状態更新関数のうち、線形変換でないもの。

・Sボックス(S箱)：3～10ビット程度の置換表。高い非線形性と攪拌性を伴った変換を表参照で行えるうえ、簡単な構成による実現が可能であることから、暗号の実装に多く用いられる。

【実施例】

【００１６】

以下、本発明の実施例を図面を用いて説明する。

【００１７】

図１は本実施例における疑似乱数生成装置(疑似乱数生成ユニット)の機能構成を表す概略図である。以下、図1に従って疑似乱数生成装置の構成を説明する。

【００１８】

疑似乱数生成装置(101)は、外部入力として、パラメータ入力制御ユニット(105)を介し

て鍵情報(121)及び乱数列番号(122)を、さらに出力長(123)からなる入力データ (102)を受け取る。パラメータ入力制御ユニット(105)は、入力が不正であるかどうかをチェックし、鍵情報(121)及び乱数列番号(122)を初期化装置(111)に入力する。出力長(123)は制御ユニット(116)に入力される。パラメータ入力制御ユニット(105)は、例えば入力された鍵情報(121)、乱数列番号(122)が大きすぎるなど不正な値であった場合には、これらのパラメータを疑似乱数生成装置(101)に入力せずに処理を終了する。上記入力は、疑似乱数生成装置(101)のユーザ、もしくは上位から疑似乱数生成装置(101)を呼び出すシステムが入力する値である。さらに、回路を動作させるタイミングを制御するクロック信号をクロック生成ユニット(103)から受け取る。これらの情報を入力とし、疑似乱数生成装置(101)は任意長のビット列(104)を出力する。

【００１９】

疑似乱数生成装置(101)は、初期化ユニット(111)、レジスタ(112)、データ攪拌ユニット(113)、データ出力ユニット(114)、カウンタ(115)、制御ユニット(116)、および初期化ユニット(111)からレジスタ(112)への入力を制御するスイッチ(117)とデータ出力ユニット(114)への入力を制御するスイッチ(118)から構成される。

【００２０】

スイッチ(117)(118)の接続／切断の切り替えは、制御ユニット(116)が行う。制御ユニット(116)はクロック生成ユニット(103)から、ステップカウンタとして動作するカウンタ(115)を介して信号を受信すると、制御ユニット(116)はスイッチ(118)を接続し、レジスタ(112)の保持している値をデータ出力ユニット(114)に入力し、データ出力ユニット(114)は与えられた入力からビット列(104)を出力する。また、制御ユニット(116)はクロック信号を受信すると、データ攪拌ユニット(113)を動作させ、レジスタ(112)の値を更新する。

【００２１】

上記構成は、ハードウェア、ソフトウェア、または、これらの組み合わせにて実現することが可能である。

【００２２】

図2は図1に示す疑似乱数生成装置(101)の処理手順を表すフローチャートである。以下、図2に従って本実施例における疑似乱数生成装置の処理手順を説明する。

【００２３】

ステップ１(202)：疑似乱数生成装置(101)は、パラメータ入力制御ユニット(105)を介して鍵情報K(112)、乱数列番号I(122)を受信し、初期化ユニット(111)に入力する。初期化ユニット(111)は、レジスタ長のビット列をレジスタ(112)の初期状態として生成する。制御ユニット(116)は、スイッチ(117)を接続状態とし、初期化ユニット(111)が生成したビット列をレジスタ(112)にセットする。レジスタ(112)への値がセットされたら、制御ユニット(116)はスイッチ(117)の接続を切る。

ステップ２(203)：疑似乱数生成装置(101)は、カウンタ(115)の値を1にセットする。また、出力長Nを制御ユニット(116)にセットする。

【００２４】

ステップ３(204、208)：疑似乱数生成装置(101)は、クロック生成ユニット(103)からの信号を受け取り、カウンタ(115)の値がN以下であれば以下のステップを繰り返す。カウンタ(115)の値がNを超えている場合には、処理を終了する。

ステップ４(205)：疑似乱数生成装置(101)はスイッチ(118)を接続状態にし、レジスタ(112)の値をデータ出力ユニット(114)に入力する。データ出力ユニット(114)は入力を受け取ると、一定長のビット列(104)を出力する。

ステップ5(206)：データ攪拌ユニット(113)はレジスタ(112)の値を入力としてデータ攪拌処理を行う。データ攪拌ユニット(113)の出力は再びレジスタ(112)に格納される。

ステップ6(207)：カウンタ(115)の値をインクリメントする。

【００２５】

図3は本実施例における疑似乱数生成装置(101)のデータ攪拌ユニット(113)の構成を表

す概略図である。

【００２６】

図3では、便宜上、レジスタ(112)を時刻tにおけるレジスタ(302)と時刻t+1におけるレジスタ(303)に分けて表示している。データ攪拌ユニット(301)は時刻tのレジスタ(302)の状態から時刻t+1のレジスタ(303)の状態を生成する。本実施例におけるレジスタ(112)は2つの小レジスタから構成されている。図中ではこの2つの小レジスタをバッファ(311)、ステート(312)と表記する。ステート(312)は2ワードで構成される。ここで1ワードとは、4ビットの倍数からなるデータブロックのサイズを表す。典型的なワードの単位としては、4ビット、1バイト（8ビット）、2バイト、4バイトなどを使用する。また、鍵情報K(121)の大きさをn(K)ビットとするとき、バッファ(311)は2×n(K)ビット以上であれば良い。バッファ(311)はMワードで構成されているものとする。以下、本実施例の中でパラメータの値を特定するときには、1ワードが1バイト、n(K)＝80を基準とする。このとき、バッファ(311)は20ワードで構成される。

【００２７】

以下、本実施例では1ワードは1バイトとし、M=20の場合に本実施例による疑似乱数生成装置の構成例を説明する。バッファ(311)の値を上位ワードから順にB(0), B(1), …, B(19)と表す。同様にステート(312)の値を上位ワードA(0)、A(1)と表す。

【００２８】

データ攪拌ユニット(301)は、非線形変換ユニット(321)、線形変換ユニット１(322)、線形変換ユニット２(323)で構成されている。非線形変換ユニット(321)は、バッファ(311)の一部の値を入力とし、その出力を線形変換ユニット１(322)に入力する。線形変換ユニット(322)はステート(312)の状態更新を行う関数であり、時刻tにおけるステートの値と非線形変換ユニット(321)の出力を入力とし、時刻t+1におけるステートの値を生成する。線形変換ユニット２(323)はバッファ(311)の状態更新を行う関数であり、時刻tにおけるバッファの値とステートの一部の値を入力とし、時刻t+1におけるバッファの値を生成する。

【００２９】

図4は本実施例におけるデータ攪拌ユニット(301)における非線形変換ユニット(321)の構成を表す概略図である。

【００３０】

非線形変換ユニット(321)は時刻tのレジスタ(302)のうち、バッファ(311)から4ワードを入力し、4ワードを出力する。バッファ(311)から取り出す4ワードをB(i1), B(i2), B(i3),およびB(i4)とする(i1<i2<i3<i4)。本実施例では、i2－i1, i3－i2, i4－i3, i4－i1は互いに異なる数値となるように取り出すワード位置を定める。このような数値を取ることにより、状態更新関数はより高い攪拌を行うことができる。上記の値として、例えばi1＝2、i2＝5、i3＝9、i4＝17を取ればよい。このとき、i2－i1＝3、i3－i2＝4、i4－i3＝8、i4－i1＝15となる。また、i1～14の別例として、i1＝1、i2＝4、i3＝6、i4＝16を取ってもよい。

【００３１】

非線形変換ユニット(321)は、4ワードの入力を受け取ると、それぞれのワードを小型非線形置換(404)を用いて変換し、その出力する値を連結して出力する。ここで、小型非線形置換(404)として、例えば8ビット入力8ビット出力の置換表Sボックスを用いることができる。Sボックスとしては、たとえば

Federal Information Processing Standards Publications (FIPS PUBS)、Advanced Encryption Standard (AES)、NIST、２００１年１１月２６日、Ｐ．１６、インターネット＜http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf＞で使われている以下の変換表を使うことが可能である。

【００３２】

S[256]=

{ 82, 9, 106, 213, 48, 54, 165, 56,

191, 64, 163, 158, 129, 243, 215, 251,
124, 227, 57, 130, 155, 47, 255, 135,
52, 142, 67, 68, 196, 222, 233, 203,
84, 123, 148, 50, 166, 194, 35, 61,
238, 76, 149, 11, 66, 250, 195, 78,
8, 46, 161, 102, 40, 217, 36, 178,
118, 91, 162, 73, 109, 139, 209, 37,
114, 248, 246, 100, 134, 104, 152, 22,
212, 164, 92, 204, 93, 101, 182, 146,
108, 112, 72, 80, 253, 237, 185, 218,
94, 21, 70, 87, 167, 141, 157, 132,
144, 216, 171, 0, 140, 188, 211, 10,
247, 228, 88, 5, 184, 179, 69, 6,
208, 44, 30, 143, 202, 63, 15, 2,
193, 175, 189, 3, 1, 19, 138, 107,
58, 145, 17, 65, 79, 103, 220, 234,
151, 242, 207, 206, 240, 180, 230, 115,
150, 172, 116, 34, 231, 173, 53, 133,
226, 249, 55, 232, 28, 117, 223, 110,
71, 241, 26, 113, 29, 41, 197, 137,
111, 183, 98, 14, 170, 24, 190, 27,
252, 86, 62, 75, 198, 210, 121, 32,
154, 219, 192, 254, 120, 205, 90, 244,
31, 221, 168, 51, 136, 7, 199, 49,
177, 18, 16, 89, 39, 128, 236, 95,
96, 81, 127, 169, 25, 181, 74, 13,
45, 229, 122, 159, 147, 201, 156, 239,
160, 224, 59, 77, 174, 42, 245, 176,
200, 235, 187, 60, 131, 83, 153, 97,
23, 43, 4, 126, 186, 119, 214, 38,
225, 105, 20, 99, 85, 33, 12, 125};

また、4ビット入力4ビット出力のSボックスTから8ビット入力8ビット出力のSボックスSを構成し、小型非線形置換(404)として用いても良い。図12は、本実施例における4ビット入力4ビット出力のSボックスTを使った8ビット入力8ビット出力のSボックスSの構成を表す概略図である。小型非線形置換(404)への入力(1301)をE，出力(1302)をFとする。図12の構成では、まず、入力(1301)を上位4ビットと下位4ビットに分割する。上位4ビットをE1，下位4ビットをE2と表すと、次のようになる。

E1 || E2 ← E,
EA1 ← T1[E1],
EA2 ← T2[E2],
EA ← EA1 || EA2,
EB ← L(EA),
EB1 || EB2 ← EB,
EC1 ← T3[EB1],
EC2 ← T3[EB2],
F ← EC1 || EC2.

ただし、ここでx||yはxとyの連結を表す。また、Lはアフィン変換ユニット(1312)の変換である。また、T1～T4は4ビット入力4ビット出力のSボックスである。T1～T4は同じSボックスを用いても良い。4ビット入力4ビット出力のSボックスとしては、例えば次で定義される置換表を用いることができる。

T[16] = {1, 3, 9, 10, 5, 14, 7, 2, 13, 0, 12, 15, 4, 8, 6, 11}

非線形変換ユニットの出力X1, X2, X3, X4は、8ビット入力8ビット出力のSボックスSを用いて、以下の式のように構成することができる。

X1 ← S(B(i1)),
X2 ← S(B(i2)),
X3 ← S(B(i3)),
X4 ← S(B(i4))

図5は本実施例における疑似乱数生成装置(101)のステート変換部（図3における、時刻tにおけるステート（312）から時刻t＋1におけるステート（312）への変換）の構成を表す概略図である。

【００３３】

ステート変換部は非線形変換ユニット(321)と線形変換ユニット１(322)の組み合わせである。線形変換ユニット１(322)は、非線形変換ユニット(321)の出力を補助入力とし、時刻tのステート(312)の値を入力として時刻t+1のステートの値を出力する。非線形変換ユニット(321)の出力をX1, X2, X3, X4と表すと、線形変換ユニット１(322)は以下の処理を行う。

【００３４】

P(0) ← A(0) XOR X1,
P(1) ← A(1) XOR X2,
Q(0) ← P(0) XOR C1・P(1),
Q(1) ← P(1) XOR C2・P(0),
R(0) ← Q(0) XOR X3,
R(1) ← Q(1) XOR X4

ただし、ここでCj・P(i)は1ワードをn(W)ビットとするとき、2n(W)個の元を持つ有限体上で定義される乗算を表す。また、定数C1、C2は0、1でない値を用いる。処理の最終結果R(0)、R(1)をそれぞれ時刻t+1のステートA(0)、A(1)にセットする。

【００３５】

このような線形変換をおこなうことにより、攪拌処理における非線形変換ユニット(321)の働きを最大限に高めることができる。

【００３６】

図6は本実施例における疑似乱数生成装置(101)のバッファ変換部（図3における、時刻tにおけるバッファ（311）から時刻t＋1におけるバッファ（311）への変換）の構成を表す線形変換ユニット２(323)の概略図である。

【００３７】

線形変換ユニット２(323)は、時刻tにおけるステート(312)を補助入力とし、時刻tのバッファ(311)の値を入力として時刻t+1のバッファ(311)の値を出力する。線形変換ユニット２(323)はワード単位の巡回置換と３個のフィードバックで構成される。フィードバック処理はワード単位で行う。線形変換ユニット２(323)は以下の処理を行う。

【００３８】

Y(i) ← B(i) (0≦i ＜ 20),
Y(2) ← Y(2) XOR Y(4),
Y(5) ← Y(5) XOR Y(8),
Y(9) ← Y(9) XOR Y(16),
Y(M-1) ← Y(M-1) XOR A(0),
W(i+1) ← Y(i) (0≦i ＜19),
W(0) ← Y(19)

処理の最終結果W(i)をそれぞれ時刻t+1のバッファB(i)にセットする。

【００３９】

フィードバック「Y(p(i)) ← Y(p(i)) XOR Y(q(i))」は上記の構成に限らず、p(i)－q(i)がすべて違う値を取る組み合わせから選んでも良い。また、ソフトウェアでの実装にお

いて、Y(p(i))，Y(q(i))として、非線形変換ユニット(321)への入力となるバッファ(311)から取り出すワードB(i1)，B(i2)，B(i3)，B(i4)と同一の値をとってもよい。この場合、データ攪拌ユニット(113)の実行プログラムがレジスタ(302)にアクセスする回数を減らすことができ、プログラムの実行速度が向上することが期待できる。

【００４０】

図7は本実施例における疑似乱数生成装置(101)の乱数出力装置（データ出力ユニット(114)）の構成を表す概略図である。データ出力ユニット(114)は時刻tのレジスタ(302)の値からステート(312)の下位ワードA(1)を乱数ビット列(104)として出力する。

【００４１】

図8は本実施例における疑似乱数生成装置(101)の初期化ユニット(111)の構成を表す概略図である。

【００４２】

80ビット(10ワード)の鍵情報K(121)を上位ワードからK(0)、K(1)，…，K(9)と表す。また、64ビット(8ワード)の乱数列番号(122)を上位ワードからI(0)，I(1)，…，I(7)と表す。初期化ユニット(111)は、パラメータ入力制御ユニット(105)を介して上記鍵情報(121)と乱数列番号(122)を受け取り、以下の処理に従ってレジスタ(824)に値をセットする。

【００４３】

B(i) ← K(i)，(0 ≦ i ＜10)，
B(i) ← I(i－10)，(10 ≦ i ＜16)，
B(i) ← C3(i－16)，(16 ≦ i ＜20)，
A(0) ← C3(4)，
A(1) ← C3(5)

ここで、C3(i) (0≦i＜6)は任意の定数である。次に、データ攪拌ユニット(825)を繰り返し用いてレジスタの値を攪拌する。ここでデータ攪拌ユニット(825)は図３に記載のデータ攪拌ユニット(301)を用いて良い。この場合、40回以上の繰り返しを行うことが望ましい。

【００４４】

初期化ユニット(111)は規定の処理を行った後、レジスタ(824)の値をレジスタ(112)にセットする。

【００４５】

図13は本実施例における疑似乱数生成装置(101)の初期化ユニット(111)の別の構成を表す概略図である。図13の表す初期化ユニット(111)は、図8の表す初期化ユニット(111)に、さらにワード攪拌ユニット(1411)を加えたものである。80ビット(10ワード)の鍵情報K(121)を上位ワードからK(0)、K(1)，…，K(9)と表す。また、64ビット(8ワード)の乱数列番号(122)を上位ワードからI(0)，I(1)，…，I(7)と表す。初期化ユニット(111)は、まず定数C4をレジスタの値にセットする。

【００４６】

B(i) ← C4(i)，(0 ≦ i ＜ 20)，
A(i) ← C4(i+20)，(i = 0，1)．

ここで、C4(i) (0 ≦ i ＜22)は任意の定数である。次に、初期化ユニット(111)は、パラメータ入力制御ユニットを介して入力されるデータを、以下のステップに従って攪拌し、レジスタ(112)の初期状態を定める。

【００４７】

ステップ１．パラメータ入力制御ユニット(105)は、鍵情報K(121)、乱数列番号(122)を1ワードずつ、順にワード攪拌ユニット(1411)に入力する。
ステップ２．ワード攪拌ユニット(1411)は、パラメータ入力制御ユニット(105)から入力された1ワードのデータと、レジスタ(824)に格納されているデータの1ワードを排他的論理和し、計算結果をレジスタ(824)の元の位置に戻す。
ステップ３．データ攪拌ユニット(825)でレジスタ（824）の値を攪拌する。
ステップ１～３を、パラメータ入力制御ユニット(105)がすべての鍵情報K(121)、乱数列

番号(122)を入力し終えるまで繰り返す。
ステップ４．データ攪拌ユニット(825)を繰り返し用いてレジスタ（824）の値を攪拌する。

【００４８】
ここで、データ攪拌ユニット(825)は図３に記載のデータ攪拌ユニット(301)を用いて良い。また、ステップ４では、20回以上の繰り返しを行うことが望ましい。

【００４９】
初期化ユニット(111)は規定の処理を行った後、レジスタ(824)の値をレジスタ(112)にセットする。

【００５０】
次に、本実施例の、鍵情報K(121)の大きさn(K)ビットが128ビットである場合のパラメータの選び方について説明する。n(K)=128の場合も、機器の構成、処理の手順はn(K)＝80の場合と同様に行うことができる。n(K)=128の場合、バッファ(311)は32ワードで構成され、バッファ(822)の値を上位ワードから順にB(0)，B(1)，…，B(19)と表す。ステート(823)の値は上位ワードA(0)、A(1)と表す。

【００５１】
非線形変換ユニット(321)は時刻tのレジスタ(302)のうち、バッファ(311)から4ワードを入力し、4ワードを出力する。バッファ(311)から取り出すワードB(i1)，B(i2)，B(i3)，B(i4)とする(i1＜i2＜i3＜i4)。n(K)＝128の場合、例えばi1＝1、i2＝5、i3＝15、i4＝29を取ればよい。

【００５２】
また、線形変換ユニット２(323)はワード単位の巡回置換と３個のフィードバックで構成される。フィードバック処理はワード単位で行う。n(K)=128の場合、例えば線形変換ユニット２(323)は以下の処理を行う。

【００５３】
Y(i) ← B(i)　(0 ≦ i ＜32),
Y(1) ← Y(1) XOR Y(4),
Y(5) ← Y(5) XOR Y(14),
Y(15) ← Y(15) XOR Y(28),
Y(M－1) ← Y(M-1) XOR A(0),
W(I＋1) ← Y(i)　(0≦i＜31),
W(0) ← Y(31)

上記実施例によれば、最大距離分離符号行列を用いた線形変換を小型化でき、疑似乱数生成装置全体としても小型化できる。また安全性を保ったままステートおよびその状態更新関数を最小化することが可能である。

【００５４】
また、安全性の要となるステートの非線形変換を、ステート、バッファの線形状態更新関数の処理と分離して行うことにより、非線形変換の内部処理を並列化することが可能となる。この結果、特にハードウェア実装では、状態更新関数の処理に要する時間が短くなり、高速処理が可能となる。

【００５５】
本実施例の一つの好ましい応用例は、たとえば、携帯電話などの小型電子機器において動画再生を行うなど高速なリアルタイム処理が必要となるシステムである。以下、本実施例を利用したデータの暗号化および配信システムについて説明する。図9は本実施例における疑似乱数生成装置(101)を用いて暗号化処理を行う暗号復号処理装置の構成を表す概略図である。

【００５６】
暗号復号処理装置(901)は疑似乱数生成装置(101)を内蔵しており、暗号化処理を行う場合、鍵情報(911)と乱数列番号(912)、暗号化対象である入力データ(902)を入力し、暗号文である出力データ(903)を出力する。また、暗号復号処理装置(901)は、入力データ(902

)のデータ長を測定する入力長判定装置(923)を持つ。暗号化処理では、入力長判定装置(923)の出力を出力長(123)とし、鍵情報(911)、乱数列番号(912)から疑似乱数生成装置(921)を介して生成されたビット列(922)を入力データ(902)とビットごとに排他的論理和回路(924)によって排他的論理和処理することで出力データ(903)を生成する。図９の暗号復号処理装置(901)においては、暗号化処理と復号処理は等価な処理であり、復号処理では入力データ(902)として暗号文を与えれば、出力データ(903)として復号文が得られる。

【００５７】

図10は上記実施例における各装置(101、901)をソフトウェアを用いて実装する場合の機器構成例である。

【００５８】

上記各装置(101、901)は、記憶装置(1011)、CPU(1012)、メモリ(1013)、入出力インターフェース(1014)が、データバスなどの内部通信線で繋がれている一般的な情報処理装置(1001)上に構成することが可能である。

【００５９】

疑似乱数生成装置(101)は、記憶装置(1011)に処理プログラム(1023)としての疑似乱数生成プログラム(1023)、秘密鍵(1021)、および乱数列番号(1022)が格納されており、疑似乱数生成プログラム(1023)がCPU(1012)により実行されることで、実現される。

【００６０】

なお、処理プログラム(1023)は、あらかじめ、上記記憶装置(1011)に格納されていても良いし、必要なときに、入出力インターフェース(1014)と情報処理装置（1001）が利用可能な媒体を介して、他の装置から上記記憶装置(1011)に導入されてもよい。媒体とは、たとえば、入出力インターフェース(1014)に着脱可能な記憶媒体、または通信媒体（すなわちネットワークまたはネットワークを伝搬する搬送波やディジタル信号）を指す。

【００６１】

乱数生成処理時の入力データ(1031)としては、出力長N(123)があり、入出力インターフェース(1014)を介してメモリ(1013)に記憶される。乱数生成処理を行う場合には、CPU(1012)が疑似乱数生成プログラム(1023)を実行し、以下のステップに従った処理を行う。

ステップ１：疑似乱数生成プログラム(1023)をメモリ(1013)にロードする。

ステップ２：上記秘密鍵(1021)と乱数列番号(1022)を用いて図1におけるレジスタ(112)を実現するメモリ(1013)上の変数配列を初期化する。

ステップ３：CPU(1012)上でデータ攪拌処理とデータ出力処理を繰り返し行い、ビット列を生成する。

【００６２】

乱数生成処理の結果は、同じくメモリ(1013)に記憶し、入出力インターフェース(1014)を介して出力データ(1032)として出力する。

【００６３】

なお、図１０では乱数列番号(1022)はあらかじめ記憶装置(1011)に記憶されているものとしたが、出力長N(1031)と同様に入力データとして与えても良い。また、秘密鍵(1021)も同様に入力データとして与えても良い。

【００６４】

図10に示す装置に、上記実施例における暗号復号処理装置(901)をソフトウェア実装する場合は、以下のように構成する。記憶装置(1011)に、処理プログラム(1023)として暗号復号処理プログラムを格納し、疑似乱数生成装置の場合と同様に、秘密鍵(1021)、乱数列番号(1022)を格納する。そしてCPU(1012)が、暗号復号処理プログラム(1023)を実行することにより、暗号復号処理装置(901)が実現される。

【００６５】

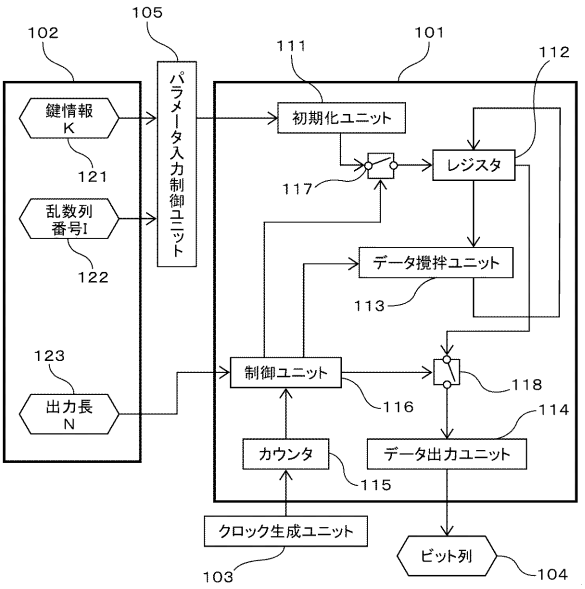
暗号処理を行う場合には、暗号復号処理プログラム(1023)はメモリ(1013)にロードされ、CPU(1012)上で演算処理を行う。暗号処理の入力データ(1031)は入出力インターフェース(1014)を介してメモリ(1013)に記憶される。また、暗号処理の結果は同じくメモリ(1013)に記憶され、入出力インターフェース(1014)を介して出力データ(1032)として出力され

る。
【００６６】
また、疑似乱数生成装置の場合と同様に、乱数列番号(1022)はあらかじめ記憶装置(1011)に記憶されていてもよいし、入力データ(1031)として与えても良い。また、秘密鍵(1021)も同様に入力データ(1031)として与えても良い。
【００６７】
図11は本実施例を応用したコンテンツ配信システムの構成例である。
【００６８】
コンテンツ配信システムは、データ配信サーバ(1201)、受信端末(1202)、およびデータの転送路であるネットワーク(1204)からなる。コンテンツは暗号処理を施され、暗号文(1203)としてデータ配信サーバ(1201)から受信端末(1202)に配信される。ネットワークは有線でも無線でも良い。
【００６９】
データ配信サーバ(1201)は、記憶装置(1211)、CPU(1212)、メモリ(1213)、暗号処理システム(1214)および通信装置(1215)から構成される。コンテンツ(1216)は、記憶装置(1211)に格納されている。受信端末(1102)は、記憶装置(1221)、CPU(1222)、メモリ(1223)、暗号処理システム(1224)、コンテンツの再生装置(1226)から構成されている。
【００７０】
データの配信は、次のステップに従って行う。
ステップ1：データ配信サーバ(1201)と受信端末(1202)は、同じ乱数列を共有できるように、事前に鍵情報(121)を秘密裡に共有し、乱数列番号(122)も、秘密裡である必要はないが、共有する。これらの情報を共有するには、例えば、公開鍵暗号技術を用いた鍵配送方法を用いることができる。
ステップ2：データ配信サーバ(1201)は、上記共有情報を与えた暗号処理システム(1214)を用いてコンテンツ(1216)を暗号化する。
ステップ3：データ配信サーバ(1201)は、暗号文データ(1203)を通信装置(1215)を用いて、ネットワーク(1204)を通じて受信端末(1202)に送信する。
ステップ4：受信端末(1202)は、通信装置(1225)を介して受信した暗号文データ(1203)を、上記共有情報を与えた暗号復号処理装置(1224)を用いて復号する。
ステップ5：受信端末(1202)は、復号化されたコンテンツを再生装置(1226)で再生する。
【００７１】
上記通信方法では、通信が始まる前に乱数列番号(122)を共有しているが、乱数列番号(122)は暗号文(1203)に付随する形で通信中に受信端末(1202)に送付しても良い。この場合には、乱数列番号(122)を含む通信データにメッセージ認証子をつけることで、通信路上でのデータ改ざんによる安全性低下を避けることができる。
【図面の簡単な説明】
【００７２】
【図１】疑似乱数生成装置の概略構成を例示する図である。
【図２】疑似乱数生成装置の処理手順を例示するフローチャートである。
【図３】疑似乱数生成装置のデータ攪拌ユニットの概略構成を例示する図である。
【図４】疑似乱数生成装置の非線形変換ユニットの構成を例示する概略図である。
【図５】疑似乱数生成装置のステート変換部の構成を例示する概略図である。
【図６】疑似乱数生成装置のバッファ変換部の構成を例示する概略図である。
【図７】疑似乱数生成装置の乱数出力装置の構成を例示する概略図である。
【図８】疑似乱数生成装置の初期化ユニットの構成を例示する概略図である。
【図９】疑似乱数生成装置を用いて暗号化処理を行う暗号復号処理装置の構成を例示する概略図である。
【図１０】各装置をソフトウェアで実現する場合に用いる機器構成の概略を例示する図である。
【図１１】疑似乱数生成装置を応用したコンテンツ配信システムの概略を例示する図であ

る。
【図１２】疑似乱数生成装置の小型非線形置換の概略構成を例示する図である。
【図１３】疑似乱数生成装置の初期化ユニットの別の構成を例示する概略図である。
【符号の説明】
【００７３】
１０１：疑似乱数生成装置、１０２：外部入力データ、１０３：クロック生成ユニット、１０４：出力ビット列、１０５：パラメータ入力制御ユニット、１１１：初期化ユニット、１１２：レジスタ、１１３：データ攪拌ユニット、１１４：データ出力ユニット、１１５：ステップカウンタ、１１６：制御ユニット、１１７、１１８：スイッチ、１２１：鍵情報、１２２：乱数列番号、１２３：出力データ長、３１１：バッファ、３１２：ステート、３２１：非線形変換ユニット、３２２、３２３：線形変換ユニット、４０４：置換表（Sボックス）、９２３：入力長判定ユニット。

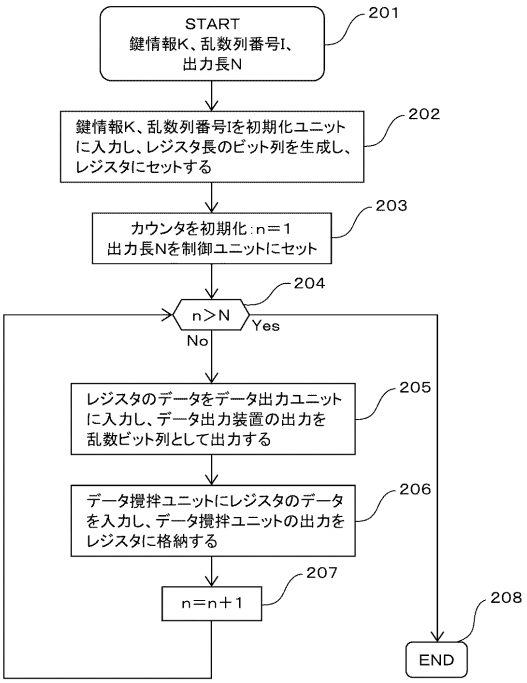
【図１】

図１



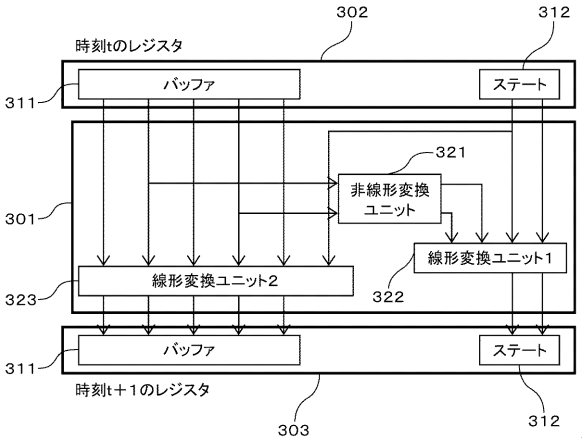
【図２】

図２



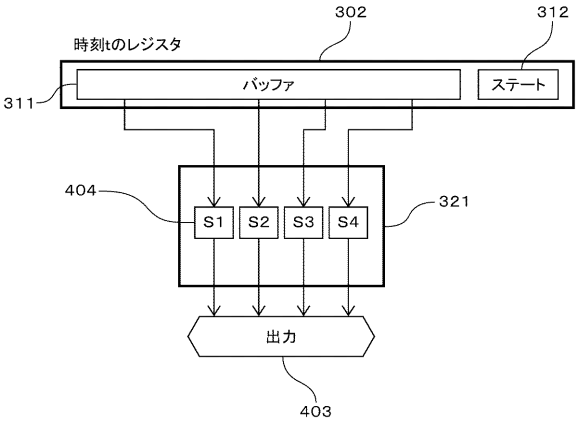
【図３】

図３



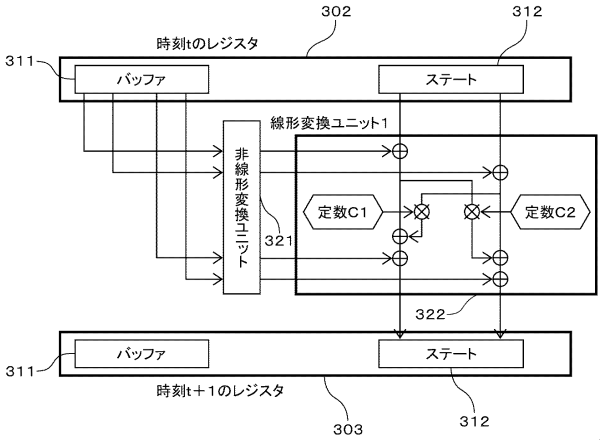
【図４】

図４



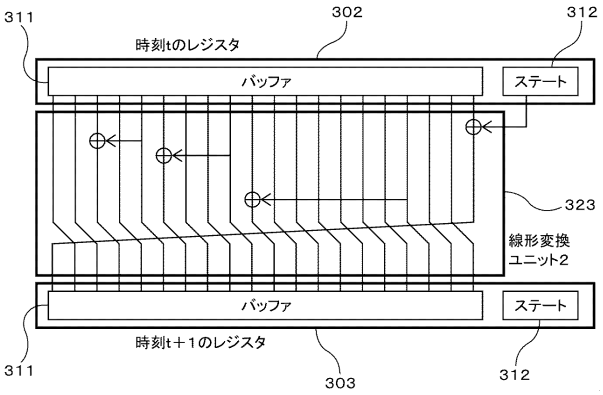
【図５】

図５



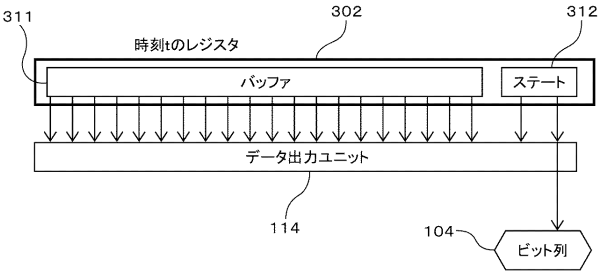
【図６】

図６



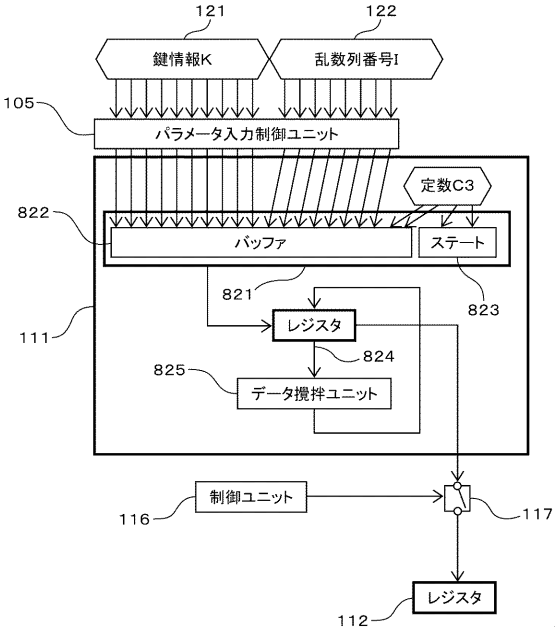
【図７】

図７



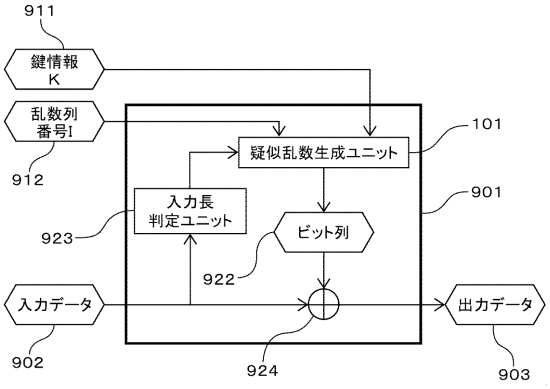
【図８】

図８



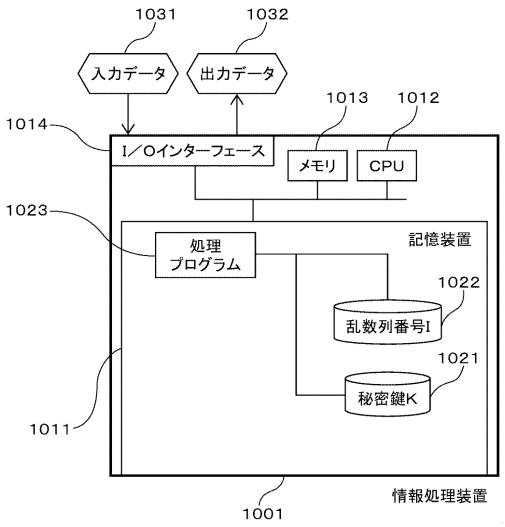
【図９】

図９



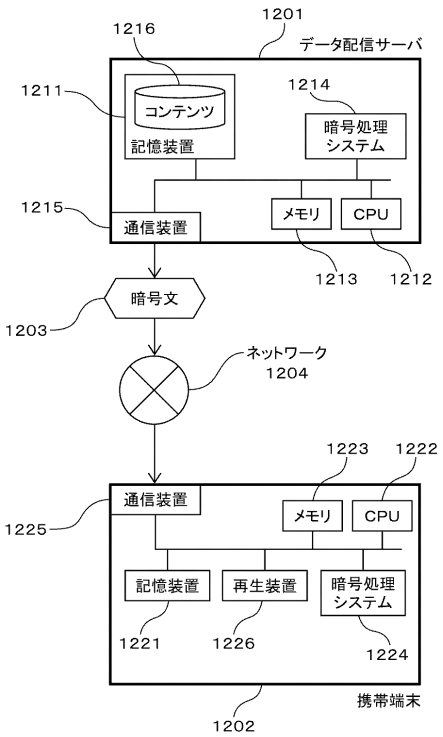
【図１０】

図１０



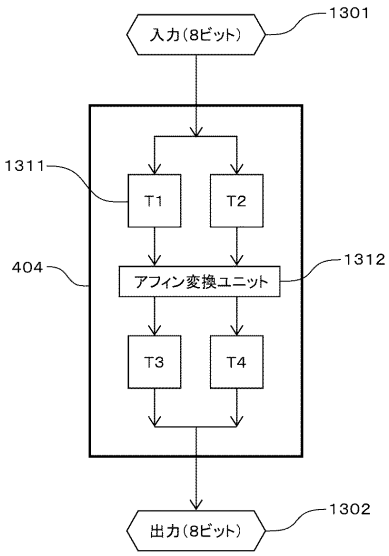
【図１１】

図１１



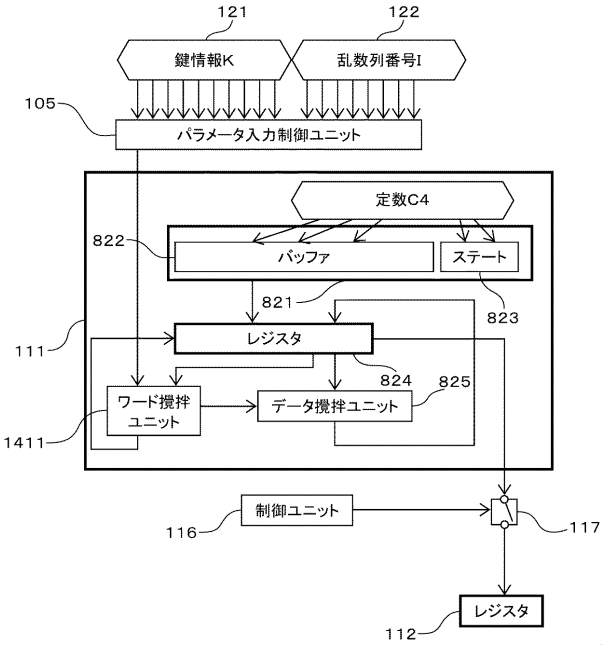
【図１２】

図１２



【図１３】

図１３



フロントページの続き

(56)参考文献　国際公開第２００６／０１９１５２（WO，Ａ１）
特開２００３－０３７４８２（ＪＰ，Ａ）
特開２００７－０４１６２０（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　　７／５８
Ｇ０９Ｃ　　１／００
Ｈ０４Ｌ　　９／００