(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第6345237号
(P6345237)
(45)発行日　平成30年6月20日(2018.6.20)　(24)登録日　平成30年6月1日(2018.6.1)

(51)Int.Cl.　F I
G09C 1/00 (2006.01)　G09C 1/00 610A

請求項の数 15 (全 18 頁)

(21)出願番号　特願2016-523807(P2016-523807)
(86)(22)出願日　平成26年6月19日(2014.6.19)
(65)公表番号　特表2016-523391(P2016-523391A)
(43)公表日　平成28年8月8日(2016.8.8)
(86)国際出願番号　PCT/US2014/043169
(87)国際公開番号　WO2015/047487
(87)国際公開日　平成27年4月2日(2015.4.2)
審査請求日　平成29年6月2日(2017.6.2)
(31)優先権主張番号　13/929,589
(32)優先日　平成25年6月27日(2013.6.27)
(33)優先権主張国　米国(US)

早期審査対象出願

(73)特許権者　507364838
クアルコム，インコーポレイテッド
アメリカ合衆国　カリフォルニア　９２１２１　サン　ディエゴ　モアハウス　ドライブ　５７７５
(74)代理人　100108453
弁理士　村山　靖彦
(74)代理人　100163522
弁理士　黒田　晋平
(72)発明者　ロベルト・アヴァンジ
アメリカ合衆国・カリフォルニア・９２１２１・サン・ディエゴ・モアハウス・ドライヴ・５７７５

審査官　青木　重徳

最終頁に続く

(54)【発明の名称】平文データを暗号化するための方法および装置

(57)【特許請求の範囲】
【請求項１】
平文データを暗号化するための方法であって、
プロセッサによって、少なくとも1つの平文データ入力を受信するステップと、
前記プロセッサによって、ナンスが適用される前に、暗号化関数を模擬するラウンド関数の第1のシーケンスによって、前記少なくとも1つの平文データ入力を暗号化することで前記暗号化関数を適用し、その後、ナンスデータ出力を作成するために、関数を通じて前記ナンスを適用するステップであって、前記ナンスデータ出力は、暗号化された出力データを作成するために、前記暗号化関数を模擬するラウンド関数の第2のシーケンスによって暗号化される、ステップと、
前記プロセッサによって、前記暗号化された出力データをメモリへ送信するステップとを備える方法。
【請求項２】
前記ナンスは、暗号化されて記憶される、請求項1に記載の方法。
【請求項３】
前記ナンスは、暗号化されずに記憶される、請求項1に記載の方法。
【請求項４】
前記関数は、XOR関数を含む、請求項1に記載の方法。
【請求項５】
前記関数は、前記ナンスから値を導出する数理的関数である、請求項1に記載の方法。

【請求項６】
前記関数は、バイナリもしくは算術加算定数、循環回転、または前記入力を表現するビットの前記関数への任意の転置を含む、請求項5に記載の方法。
【請求項７】
前記プロセッサによって、メモリからの前記暗号化された出力データを復号するステップをさらに備える、請求項1に記載の方法。
【請求項８】
前記プロセッサによって実行されると、前記プロセッサに請求項1乃至7のいずれか1項に記載の方法を実行させるためのコードを記録したコンピュータ可読記録媒体。
【請求項９】
平文データを暗号化するためのデバイスであって、
少なくとも1つの平文データ入力を受信するための手段と、
ナンスが適用される前に、暗号化関数を模擬するラウンド関数の第1のシーケンスによって、前記少なくとも1つの平文データ入力を暗号化することで前記暗号化関数を適用し、その後、ナンスデータ出力を作成するために、関数を通じて前記ナンスを適用するための手段であって、前記ナンスデータ出力は、暗号化された出力データを作成するために、前記暗号化関数を模擬するラウンド関数の第2のシーケンスによって暗号化される、手段と、
前記暗号化された出力データをメモリへ送信するための手段と
を備えるデバイス。
【請求項１０】
前記ナンスは、暗号化されて記憶される、請求項9に記載のデバイス。
【請求項１１】
前記ナンスは、暗号化されずに記憶される、請求項9に記載のデバイス。
【請求項１２】
前記関数は、XOR関数を含む、請求項9に記載のデバイス。
【請求項１３】
前記関数は、前記ナンスから値を導出する数理的関数である、請求項9に記載のデバイス。
【請求項１４】
前記関数は、バイナリもしくは算術加算定数、循環回転、または前記入力を表現するビットの前記関数への任意の転置を含む、請求項13に記載のデバイス。
【請求項１５】
メモリからの前記暗号化された出力データを復号するための手段をさらに備える、請求項9に記載のデバイス。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、平文データを暗号化し、対応する暗号文データを復号するための方法および装置に関する。
【背景技術】
【０００２】
メモリ分析器の使用は、内容を配信することの完全性および秘密性への大きい脅威を意味する。たとえコード内に含まれるデータを保護するために大変な注意が注がれていても、メモリの内容は、バスのスニッフィングによって取り込まれ得る。たとえば、このことは、たとえ内容が暗号化された形態で配信されても、表示にとって安全な環境において復号された後、未加工の内容を漏洩するために使用され得る。このことは、メモリへの書込みに対応する電気信号を「読み取ること」によって成し遂げられ得る。他のより精巧な攻撃は、プロセッサをだまして、攻撃者によって選ばれたデータを読み取り、処理させるために、これらの信号をリプレイすることさえできる。

【０００３】
内容の提供者は、しばしば、未加工の内容を処理するための特定の要件を有する。最低水準では、内容は、決して暗号化されずにメモリ内に記憶されてはならない。たいていの場合、少なくとも、メモリのスクランブリングすなわち暗号化のいくつかの形態が、物理的な攻撃を防ぐためにすべてのメモリ記録に適用されるべきであるという要件が存在する。例として、特有のアドレスに書き込まれたデータは、通常、暗号化されていないデータ、アドレス、およびマスターキーの関数である。このことは、同じデータが異なるアドレスに書き込まれる場合、異なる符号化を有することを保証する。平文データの暗号化をランダム化するためにナンスを使用することは、これらのナンスが安全な方法で記憶かつ取得される場合、リプレイ攻撃を防ぐために使用され得る。

【０００４】
その上、安全な通信のためのスループット要件は、現在のストリーム暗号およびブロック暗号にテストを受けさせており、電力要件およびエリア要件を同時に制御しながらスループットを高くするための新規の構築が、望まれる。

【０００５】
あいにく、現在の技法はしばしば非効率であり、より強固なレベルの保護、同じセキュリティレベルにおけるより高いスループット、ならびに電力要件およびハードウェア実装の場合にはエリア要件の著しい増加を伴わないことが、望まれ得る。

【発明の概要】
【課題を解決するための手段】

【０００６】
本発明の態様は、平文データを暗号化するための装置および方法に関し得る。方法は、少なくとも1つの平文データ入力を受信することと、関数を通じてナンスを、平文データ出力を作成するために少なくとも1つの平文データ入力に、および/または中間的なナンスデータ出力(Nonced data output)を作成するために、少なくとも1つの平文データ入力に適用された暗号化関数の一部分の中間的な値に、適用することと、暗号化された出力データを作成するために、ナンス平文データ出力(Nonced plaintext data output)および/または中間的なナンスデータ出力のうちの少なくとも1つに、暗号化関数を適用することと、を含む。暗号化された出力データは、次いで、メモリへ送信される。

【図面の簡単な説明】

【０００７】
【図１Ａ】平文データ入力の一連のブロックが、暗号化関数およびナンスを使用して暗号化されるプロセスを示すフロー図である。
【図１Ｂ】図1Aの逆方向の復号処理を示すフロー図である。
【図２】ラウンドと呼ばれる類似の計算ブロックの反復に基づく、ブロック暗号の共通の構造を示すフロー図である。
【図３】ラウンド関数の第1および第2のセットを使用するとともに、ナンスを用いる暗号化処理の中間的なステップを増大させて、拡大された形態でデータ入力を暗号化するためのプロセスを示すフロー図である。
【図４Ａ】個々のブロックの暗号化処理を異なる方法で修正するために、同一のキーおよびナンスもしくはナンスから導出される値を使用して、平文データ入力の一連のブロックを暗号化するためのプロセスを示すフロー図である。
【図４Ｂ】図4Aの逆方向の復号処理を示すフロー図である。
【図５】共通のナンスから導出された異なる値を暗号化処理の中間的なステップに適用することによって、いくつかの異なる出力を取得する、データ入力を暗号化するためのプロセスを示すフロー図である。
【図６】復号かつ再暗号化する必要なく、暗号化されたメモリの大容量記憶装置への保存および復元を可能にする目的で、データ暗号化技法を実施するための例示的なコンピュータハードウェアシステムの図である。

【発明を実施するための形態】

【０００８】
「例示的」または「例」という単語は、本明細書では「例、事例、または例示の働きをすること」を意味するために使用される。「例示的」または「例」として本明細書に記載される任意の態様または実施形態は、他の態様または実施形態に比べて好ましいか、または有利であると必ずしも解釈されるべきではない。
【０００９】
本発明の実施形態は、メモリ内に記憶されるデータの保護のための、強化型メカニズムを提供するための技法に関する。詳細には、メモリ暗号化を強化するために、ブロック暗号の機能性を拡張する方法およびプロセスが記載される。さらに、これらの技法は、また、以下で説明されるように、性能、スループット、および電力消費を改善し得る。これらの技法は、また、(有線またはワイヤレスの)ネットワークを介する伝送の安全なデータ記憶の目的で、性能、スループット、および電力消費を改善するために使用され得る。
【００１０】
一実施形態では、暗号化方式は、平文データ入力の一連のL個のブロックが暗号化関数(たとえば、ブロック暗号)を使用して暗号化されて、利用される。ブロック暗号を用いる暗号化の前に、ナンスは、関数を通じて平文データ入力に適用される。詳細には、複数の平文データ入力を受信することと、ナンス平文データ出力を作成するために、関数を通じてナンスを複数の平文データ入力に適用することと、暗号化された出力データを作成するために、ブロック暗号などの暗号化関数をナンス平文データ出力に適用することと、暗号化された出力データをメモリへ送信することとを含む、平文データを暗号化する方法またはプロセスが開示される。
【００１１】
一実施形態では、以下でより詳細に説明されるように、方法は、複数の平文データ入力を受信することと、関数を通じてナンスを、平文データ出力を作成するために複数の平文データ入力に、および/または中間的なナンスデータ出力を作成するために、複数の平文データ入力に適用された暗号化関数の一部分の中間的な値に、適用することと、暗号化された出力データを作成するために、ナンス平文データ出力および/または中間的なナンスデータ出力のうちの少なくとも1つに、暗号化関数を適用することと、を含み得る。暗号化された出力データは、次いで、メモリへ送信される。
【００１２】
平文データ入力のL個のブロックの暗号化をランダム化すること
図1Aに見られ得るように、一実施形態では、方法またはプロセス100は、複数の平文データ入力(入力1～入力L)110が受信されて、実行される。ナンス120は、関数122を通じて、平文データ入力(入力1～入力L)110に適用される。ナンス120は、平文データ(入力1～入力L)110のL個のブロックの暗号化をランダム化するために使用され得る。図1Aに見られ得るように、平文データ(入力1～入力L)110のL個のブロックは、受信され、ナンス120は、ナンス平文データ出力を作成するために、関数(f1、f2、...、fL)122によって適用され得る。一実施形態では、以下でより詳細に説明されるように、ナンス120に適用するための関数は、XOR関数を含み得る。別の実施形態では、XOR関数の代わりに、モジュラー加算関数が使用され得る。暗号化関数130(たとえば、ブロック暗号)は、次いで、暗号化された出力データ(出力1～出力L)140がメモリへ出力されるように、ナンス平文データ出力に適用され得る。
【００１３】
平文の相等しいブロックが(同時に処理されるL個のブロック110の中で)同一の暗号化を有することを回避するために、ナンス120がいくつかの変形を受けてもよいことを諒解されたい。同様に、L個の平行な暗号化において使用される暗号化関数130で使用される暗号のキーが同一であってもよいので、キースケジュールは、L回やり直される必要がない。
【００１４】
さらに、ナンス120は、以下でより詳細に説明されるように、ユースケースに応じて、暗号化されない方法または暗号化される方法のいずれかで、メモリの小さい内部の保護さ

れたエリアの中に記憶され得るか、またはメインメモリの中に記憶され得るかのいずれかである。

【００１５】

同様に、簡単のために、暗号化関数130で使用される特定の暗号化キーは表されない。しかしながら、暗号化関数が、暗号化処理における暗号化関数によって使用されるキーである、さらなる入力を取ることを諒解されたい。その上、暗号化関数が同一の暗号化キーを使用して反復されるブロック暗号の場合には、様々な垂直のパイプラインが同一のキースケジュールを共有し得、ここで、いくつかの固定したビット転置(回転などの)が、暗号化関数130において使用される前に、ラウンドに適用され得ることを諒解されたい。ハードウェア実装では、これらの転置は、シリコン内の異なる配線に相当するだけなので、性能の影響を有するはずがない。

【００１６】

関数(f1、f2、...、fL)122は、ナンス平文データ出力の計算を、攻撃者にとって予測不可能な方法で擾乱させるために、ナンス120から値を導出する数理的関数であり得る。これらは、選ばれた暗号化関数130に関係し得る、定数、異なる循環回転、または他の関数を用いたマスキングであり得る。ナンス120のサイズが暗号ブロック長よりも大きい場合、関数は、単にナンスの分割部分の抽出であってもよい。

【００１７】

さらに、方法100は、同一の暗号化関数130のL個またはL+1個の実装を利用して、並列化可能であり得る(または、異なる暗号化関数が利用され得る)。図1Aに見られ得るように、破線150では、暗号文の拡張がL+1番目の実装として示され、これは1つのさらなるブロックを出力する。同様に、暗号化関数130に対する同一の暗号化キーが各ブロックに対して使用され得るので、サブキーの導出は一度だけ実行されればよく、したがって、ハードウェアリソースを節約する。

【００１８】

いくつかの実施形態では、ナンス120が初期設定ベクトルの役割と類似の役割を果たすので、十分なセキュリティが、ナンス120を、アクセス可能なメモリエリア内に暗号化しないで記憶することによって提供され得る。この手法の利点は、ナンス120はブロックサイズよりも短くてもよく、したがって、関数動作122において、入力ブロック110の選択されたビットフィールドのみに適用され得ることである。この方式は、メモリ暗号化にとって有用であり得る。例として、ブロック暗号が128ビットのブロックサイズを有し、キャッシュラインが128バイト長である場合、L=8を設定することによって、全体のキャッシュラインは、キャッシュの最後のレベルから吐き出されるとき、直ちに暗号化され得る。

【００１９】

したがって、前に説明したように、プロセス100は、メモリ暗号化を強化するために、ブロック暗号の機能性を拡張する。詳細には、暗号化方式100は、暗号化関数130を使用してそれぞれ暗号化される平文データ入力(入力1～入力L)110の一連のL個のブロックを利用し、ここで、暗号化関数130を用いる暗号化の前に、ナンス120が、関数122を通じて平文データ入力100に適用される。暗号化関数130は、暗号化された出力データ(出力1～出力L)140がメモリへ出力されるように、ナンス平文データ出力に適用され得る。

復号は、逆に働く。たとえば、図1Bを参照すると、対応する復号プリミティブである暗号化関数130の逆関数は、メモリからの暗号化された出力データ(入力140として示す)に適用され得、入力$_i$およびナンスの合成物、たとえば、i=1、2、...、Lおよびナンスに対して、入力$_i$

【００２０】

【数１】

⊕

【００２１】

ナンスを計算するために使用され得、そこから元の入力が復元可能である(出力110として示す)。

【００２２】

ブロック暗号をランダム化すること

以下で説明されるように、平文データ入力110は、ナンス120が適用される前に、(選ばれた暗号化関数である)ブロック暗号を組み立てるラウンド関数の第1のシーケンスによって最初に暗号化され得、その後、ナンスは、ナンスデータ出力を作成するために適用される。ナンスデータ出力は、次いで、メモリへ出力される暗号化された出力データを作成するために、(選ばれた暗号化関数である)ブロック暗号を模擬するラウンド関数の第2のシーケンスによって暗号化され得る。

【００２３】

暗号化関数130(たとえば、ブロック暗号)を模擬するために、様々な構築が使用され得る。たとえば、Luby-Rackoff構築のような構築、たとえば、Feistelネットワーク(データ暗号化標準(DES:Data Encryption Standard)のような)、および置換-転置(SP:Substitution-Permutation)ネットワーク(高度暗号化標準(AES:Advanced Encryption Standard)のような)が、使用され得る。両方の場合において、1つのパラメータ化された非線形関数が、繰り返して入力に適用される。この関数の各適用は、「ラウンド」または「ラウンド関数」と呼ばれる場合がある。ラウンドの出力は、次のラウンドの入力である。平文は最初のラウンドの入力であり、暗号文は最後のラウンドの出力である。ラウンド関数は、ラウンドキーと呼ばれるさらなるパラメータを取り、ラウンドキーは、暗号化/復号キー(たとえば、暗号キー)から導出される。

【００２４】

図2を参照すると、ラウンド関数に基づいてブロック暗号を生成するためのプロセス200の一例が示される。図2に示すように、平文データ入力202は、ブロック暗号を模擬するラウンド関数の複数のN個のラウンド204に入力される。したがって、ブロック暗号は、ラウンド関数の複数のN個のラウンド204によって模擬され、ここで、k1、k2、...、kNは、それぞれラウンド1、2、...、Nに対するラウンドキーである。出力206は、平文データ入力202に適用される(ブロック暗号を模擬する)ラウンド関数によって暗号化される、暗号化された平文データ入力202である。復号は逆順序で正確な同一のプロセスのはずであることを諒解されたい。

【００２５】

例示的な実装形態は、以下で説明される。たとえば、この方式の性能効率のよい実装は、場合によってはラウンドキーを共有する、同一のブロック暗号の2つの平行な実装を必要とし得る。ハードウェア実装コストを低減するために、ナンスは、暗号の途中で適用され得る。このことによって、ナンスの適用の前の暗号の部分は1回だけ実施されなければならず、ナンスの適用の後に続く暗号の部分は2回実施される。

【００２６】

例として、プロセス300を示す図3を参照すると、平文データ入力302は、N個のラウンドのうちのM個(1≦M<N)を通じて暗号化され得、たとえば、M個のラウンド304は、M個のラウンドキー(k1、k2、... kM)を使用してパラメータ化される。次に、ナンス(v)306が適用され、-たとえば、M番目のラウンドの出力Xにそれを XORし、-XORされた出力およびナンスは、さらに、独立に(分離ブロック308)暗号化され、-(M+1)番目のラウンドでプロセスを再開する。図3に見られ得るように、N-M個のラウンド310に対するラウンドキーk'およびk''の次のラウンドは、ラウンドキーの同一のセットであってもよく、または異なる回転、もしくは異なる秘密の定数でマスクされることなどの、互いのわずかな変種であってもよい。さらに、出力が連結されて(ブロック314)、出力316をもたらし得る。

【００２７】

わずかに異なる実装形態の別の例は、Xおよびナンスのビットの転置から成り得る。たとえば、Xが、$=X_{hi}||X_{lo}$(長さの等しい2つのビットストリングの連結としての分解)に設定され、v(ナンス)が$v=v_{hi}||v_{lo}$に設定される場合、Aは$A=X_{hi}||v_{lo}$であることになり、B

はB=$X_{lo}||v_{hi}$であることになる。したがって、最後のN-M個のラウンドが十分な発散を有する場合、Xとナンスの両方の、出力の半分CとDの両方への十分な作用がある。これは単に一例であり、他のビット転置が可能であることを諒解されたい。しかしながら、ブロックサイズが十分に大きい場合、方式は、同一の平文に対して同一の暗号文の頻繁な(部分的な)繰返しをもたらし得ない。したがって、

【００２８】

【数２】

$$A = (X_{hi} \oplus v_{hi}) \| v_{lo}$$

【００２９】

および

【００３０】

【数３】

$$B = (X_{lo} \oplus v_{lo}) \| v_{hi}.$$

【００３１】

のような式のように、ナンスを次の入力の全体に作用させることが推奨され得る。ここでの重要性が、-ひとたび復号処理がN-M個のラウンドを実行すればナンスが復元され得るように、プロセスが容易に逆にできることであることを諒解されたい。さらに、連結関数314は、最後の平行なラウンドの2つの出力の連結であり得るが、-2つの入力の任意の別のビット転置がここで使用され得る。最初のM個のラウンドのハードウェア実装は二重にされなくてもよいが、-最後のN-M個のラウンドに対してのみ二重にされる点において、プロセスは有益である。復号はまた、この場合、逆に働く。出力の2つの「側部」CおよびDは、ナンスvが復元されるまで最後のN-M個のラウンドに対して平行に復号され、分離動作が逆にされ、次いで、入力の復号はM個のラウンドの中で完了される。

【００３２】

図4Aを参照すると、以前の技法を一般化する、平文データ入力(入力1～入力L)410のL個のブロックを同時に暗号化するためのプロセス400の一例が示され、ここで、ナンス420は、適切に変形された後、様々なラウンドの使用と一緒に各ブロックに付加される。詳細には、図4Aのプロセス400は、平文データ入力410が、ナンス120が適用される前にラウンド関数の第1のシーケンス(M個のラウンド404)によって暗号化され得、その後、ナンスデータ出力を作成するためにナンス420が適用されることを示す。図4Aに見られ得るように、平文データ(入力1～入力L)410のL個のブロックが受信され、関数(f1、f2、...、fL)422が、異なるナンスデータ出力を作成するために、ナンス420に適用される。一実施形態では、ナンス420に適用するための関数は、XOR関数を含み得る。あるいは、モジュラー加算または減算などの容易に逆にできる他の関数が、ナンス(から導出される値)を適用するために使用され得る。ナンスデータ出力は、次いで、メモリへ出力される暗号化された出力データ440を作成するために、ラウンド関数の第2のシーケンス(N-M個のラウンド406)によって暗号化され得る。M個のラウンド404およびN-M個のラウンド406を利用することによって、完全な暗号化関数が、それによって模擬および適用されることを諒解されたい。さらに、方法400は、メモリへ出力される暗号化された出力データを作成するために、ラウンド関数404および406の、L個(ナンスが暗号化されない場合)またはL+1個(ナンスが暗号化される場合)の実装を利用して、並列化可能であり得る。

【００３３】

関数(f1、f2、...、fL)422は、図1Aを参照して記載されるものと実質的に同一の役割を実行することを諒解されたい。しかしながら、その下にあるブロック暗号の(M+1)番目のラウンド406まで関数が実施されないという事実により、ナンスからのより複雑な導出が可能となる。AES実装の場合において、AESのキースケジューリング手順のいくつかの改変

が、関数を生成するために採用され得る。一実施形態では、関数は、ブロック暗号の最初のMラウンド404を用いて、平行に計算され得る。すべての同一のラウンドキーを様々なラウンドに与えることでなく、各垂直のパイプラインにとって固有のいくつかの固定した転置および/またはマスキングを、それらに同様に適用することが有益であり得る。同様に、ユースケース要件に応じて、ナンス420を、アクセス可能なメモリエリア内または保護されたメモリエリア内に、暗号化しないで単に記憶することは有益であり得る。というのも、ナンス420は初期設定ベクトルの役割と類似の役割を果たし、依然として十分に安全であり得るからである。

【００３４】

復号は、逆に働く。たとえば、図4Bを参照すると、入力440は図4Aの暗号化の出力であり、出力410は元の入力(すなわち、元の平文入力)に相当するはずである。

【００３５】

リソースの節約

以前の方式のすべてが、平文が暗号化関数によって直接暗号化されるという考えに基づいていたことを諒解されたい。しかしながら、ブロック暗号のためのいくつかの動作モードは、暗号文を導出するために平文にXORされるキーストリームを生成するための暗号化プリミティブ、-たとえば、カウンタ(CTR)モードを使用する。このタイプの暗号化の例は、以下で説明される。キーストリーム生成のためのリソースを節約しようと試みる場合、セキュリティを犠牲にしてあまりにも多くの節約が起きていないことが確実にされる必要がある-すなわち、様々なキーストリームブロックは、互いに相互関係がないように見えなければならない。たとえば、いくつかの入力ブロックを暗号化するために、「キーストリーム」からのブロックを再使用することは魅力的であり得る-メモリ暗号化のシナリオでは、このことは、メモリ暗号化回路のエリアの問題を容易に解決し得る。しかしながら、平文の2つのブロックP1およびP2が両方とも同一のパッドπでXORされる場合、暗号文ブロックは、

【００３６】

【数４】

$$C1 = P1 \oplus \pi$$

【００３７】

および

【００３８】

【数５】

$$C2 = P2 \oplus \pi$$

【００３９】

であることになり、それらは、

【００４０】

【数６】

$$P1 \oplus P2 = C1 \oplus C2$$

【００４１】

を満たす。このことは、平文に関する重要な情報をあらわにするおそれがあり、したがって、重大な情報を記憶するために不適切である。しかしながら、共通のハードウェアを使用してキーストリームの2つ以上のブロックの最初のラウンドだけを計算し、次いで、最後のラウンドを別個に実行することは有益であり得る。そのような方法のセキュリティは、使用済み暗号の、ラウンドが低減されたバージョンの暗号解読法と、いくつかのラウン

ドの後の中間的な値の予測可能性とに、依存する。

【００４２】

このことの一例は、図5を参照して表示される。この例示的な実施形態プロセス500では、入力502およびナンス(v)520は、L個のキーストリームブロックを生成するために使用される値である。入力502は、平文でない。同様に、出力1、出力2、...出力L540は暗号文でないが、暗号文のL個のブロックは、CTR動作モードにおけるようにこれらの値にXORされる(または、ブロック暗号の暗号化プリミティブのみを使用する他の暗号化モードのいくつかの変種では、より複雑な方法で使用される)。他の態様では、図5は図4Aと類似であり、ナンス520が適用される前にラウンドキーの第1のラウンド(M個のラウンド504)を含み、その後、ナンスデータ出力を作成するために、ナンス520が適用される。ナンス520は、ナンス出力(Nonced output)を作成するために、関数(f1、f2、...、fL)522によって適用され得る。ナンス520を適用するための関数は、XOR関数を含み得る。ナンス出力は、次いで、暗号化された出力540を作成するために、ラウンドキーの第2のラウンド(N-M個のラウンド506)によって暗号化され得る。

AES(たとえば、AES-128)がブロック暗号として選ばれる場合、現在の暗号解読に関する成果を考慮すると、M=3または4が使用され得る。6または7ラウンドに低減されたAES-128が、やはり攻撃するのに相当困難であることが論理的根拠であり、次いで、攻撃者が入力を制御できる場合に限り-それは、この状況の中で起こり得ない。たとえば、ユースケースがメモリ暗号化であり、ここで、全体のキャッシュラインが暗号化され、これらが128バイトであり、そのため8ブロック(L=8)が必要とされると仮定する。このことは、M=3に対して、合計で3+8*7=59ラウンドのAESがHWで実装される必要があり、80の代わりに約26%のエリアおよび電力の節約をもたらすことを意味する。M=4に対して、実装されるAESのラウンドの数は4+8*6=52であり、約35%の節約が得られる。最後のN-M個のラウンドのためのキースケジュールがすべてのパイプラインに共通である場合、-おそらくは、平行なパイプラインの中で、ラウンドキーの単なるいくつかの固定したビット転置を用いて、節約はもう少し大きいかもしれないが、たぶん、せいぜいそれにとどまる-というのも、これは、(M+1)番目のラウンドへの入力にXORされるべき異なる値をナンスから導出するための論理による相殺よりも、大きいはずだからである。

【００４３】

ナンスの計算

一実施形態では、新しいブロック(または、L個のブロックのセット)がメモリに書き込まれる必要があるたびに、ナンスは更新され得る。ブロック暗号が十分な発散を有する(または、ブロック暗号が最後のN-M個のラウンドにおいて十分な発散を有する)場合、ナンスを、たとえばsビットだけ単にシフトし、次いで、s個の新しい新規のランダムなビットをナンスに追加すれば十分であり得る。たとえば、これは、ナンス(v)に対して、

【００４４】

【数７】

$$v \leftarrow (v << s) \oplus r$$

【００４５】

として計算され得、ここで、rはsビットのストリングである。さらに、新規のビットは、最上位の位置からシフトインされ得、または、vは、独立にシフトかつ更新される様々なサブレジスタの中で区分され得る。しかしながら、この方策が使用される場合、暗号化されないでナンスを記憶することは、場合によっては今後のナンスを部分的に予測可能にする場合があり、それによって、場合によっては暗号解読の助けとなるので、ナンスは、暗号化されないで記憶されるべきでなく、暗号化されるべきである。さらに、ナンスが、(a)データがそこに記憶される物理的なメモリアドレスとは無関係となる値である、または(b)そのアドレスに依存する、のいずれかであり得ることに留意されたい。後者の場合に対して、ナンスは、(i)物理的なメモリアドレス、および(ii)ランダムな値、(暗号化された

)カウンタ、または上述された方法によって、もしくは違った方法によって計算された値の連結であり得る。

【００４６】

例示的なハードウェア

前に説明した方法およびプロセスを実施し得る例示的なコンピュータハードウェア600を、図6に示す。コンピュータシステム600は、バスを介して電気的に結合され得る(または適宜、他の方法で通信できる)ハードウェア要素を備えて示されている。ハードウェア要素は、少なくとも1つのメインプロセッサ602(たとえば、中央処理装置(CPU))および他のプロセッサ604を含み得る。これらのプロセッサは、汎用プロセッサおよび/または1つもしくは複数の専用プロセッサ(デジタル信号処理チップ、グラフィックスアクセラレーションプロセッサなどのような)であってよいことを諒解されたい。プロセッサは、それぞれのメモリ管理ユニット(MMU)610に結合され得、メモリ管理ユニット610は、キャッシュ612(たとえば、キャッシュは、存在してもしなくてもよく、および/または別個であっても他の要素の中に組み込まれてもよい)(破線によって囲まれる)を通じて、エンクリプタ処理ユニット620に、ならびに/またはメモリ630および/もしくは記憶デバイス640に結合され得る。以下で説明されるように、エンクリプタ620は、メモリ内に記憶されるべきデータに対してメモリ暗号化を強化するために、暗号ブロックの機能性を拡張するための前に説明した方法およびプロセスを利用し得る。

【００４７】

コンピュータ600は、入力デバイス(たとえば、キーボード、マウス、キーパッド、マイクロフォン、カメラなど)、および出力デバイス(たとえば、表示デバイス、モニタ、スピーカ、プリンタなど)のような、他のデバイス(図示せず)を含み得ることを諒解されたい。コンピュータ600は、さらに、1つまたは複数のメモリ要素、記憶デバイス630、640を含んでもよく(および/またはそれらと通信してもよく)、1つまたは複数のメモリ要素、記憶デバイス630、640は、ローカルおよび/もしくはネットワークアクセス可能な記憶装置を備えてもよく、ならびに/または、それだけに限らないが、プログラム可能、フラッシュ更新可能などとすることができる、ディスクドライブ、ドライブアレイ、光記憶デバイス、ランダムアクセスメモリ(「RAM」)および/もしくは読取り専用メモリ(「ROM」)などのソリッドステート記憶デバイスを含むことができる。コンピュータ600は、また、通信サブシステムを含んでもよく、通信サブシステムは、モデム、ネットワークカード(ワイヤレスまたは有線の)、赤外線通信デバイス、(Bluetooth(登録商標)デバイス、802.11デバイス、Wi-Fiデバイス、WiMaxデバイス、セルラー通信デバイスなどの)ワイヤレス通信デバイスおよび/またはチップセットなどを含んでもよい。通信サブシステムは、本明細書で説明されるネットワーク、他のコンピュータシステム、および/または任意の他のデバイスとデータを交換することを可能にし得る。コンピュータ600は、モバイルデバイス、非モバイルデバイス、ワイヤレスデバイス、有線のデバイスなどであってもよく、ワイヤレスおよび/または有線の接続を有してもよく、任意のタイプの電子デバイスまたはコンピューティングデバイスであってもよいことを諒解されたい。

【００４８】

一実施形態では、データが、暗号化される場所に記憶されるべき場合(決定ブロック650)、エンクリプタ620(たとえば、データを暗号化するためのデバイス)は、複数の平文データ入力(入力1～入力L)110を受信することと、ランダム化されたナンス平文データ出力を作成するために、関数(f1、f2、...、fL)122を通じてナンス122を適用することと、暗号化された出力データ(出力1～出力L)140がメモリ630へ出力されるように暗号化関数130をナンス平文データ出力に適用することとを含む、(図1Aをさらに参照して)前に説明したプロセスを実施し得る。このデータは、さらに、記憶装置640に記憶され得る。他の実施形態では、前に説明したように、暗号化関数を適用するために、エンクリプタ620は、ナンスが適用される前に、暗号化関数を模擬するラウンド関数の第1のシーケンスを利用して平文データ入力を暗号化し得る。その後で、ナンスデータ出力を作成するために、このナンスが適用される。ナンスデータ出力は、次いで、メモリ630へ出力される暗号化された

出力データを作成するために、暗号化関数を模擬するラウンド関数の第2のシーケンスによって暗号化され得る。これらの実装形態の例は、前に詳細に説明したように、図2～図5に示される。

【００４９】

しかしながら、決定ブロック650において、データが、暗号化される場所に記憶されないと決定される場合、データは、通常、メモリ630に記憶され得、および/または通常のメモリマッピング入力/出力および制御装置655が、記憶装置640へのダイレクトメモリアクセス(DMA)制御を実施するために利用され得る。

【００５０】

一般に、メモリ暗号化が利用できる場合、その内容は、それらが仮想メモリシステム内の記憶デバイスに書き込まれる前に復号される必要がある。しかしながら、このことに対応するため、本発明の実施形態によれば、DMAデータ転送チャネルは、メモリ630(たとえば、RAM、DDR RAMなど)の実際の暗号化された内容を読み取るために使用され得、それらを記憶デバイス640(たとえば、ハードドライブまたはフラッシュメモリ)のセクタに書き込むために、ならびに内容をセクタから読み取り、メモリ630内へ直接配置するために、使用され得る。したがって、これらのメモリ暗号化方法は、物理的なアドレスとは無関係となり得、ページは、さらなる暗号化/復号のオーバヘッドなしに、スワップアウトおよびバックインされ得る。

【００５１】

前に説明したシステムの利点は、メモリの内容が、それらがスワップファイルへ移動されメモリへ戻されるたびに復号かつ再暗号化される必要がないことであり、このことは、著しい電力の節約および時間の節約という結果をもたらす。さらに、本明細書で説明される技法は、物理的または電気的なメモリ攻撃に対する、-すなわち、メモリの直接の読取りに対する-、良好な直接の保護を提供するだけでなく、同一データまたは相互関係のあるデータの同じ場所への繰り返される書込みが効果的にランダム化されるので、バスのトラフィックをサイドチャネルとして使用する攻撃に対する耐性も提供する。さらに、本明細書で説明される技法は、比較的小さい追加のハードウェア実装で済む。同様に、本明細書で説明される技法は、それらが本質的に任意の普通に使われるブロック暗号に適用され得るように、十分に一般的である。さらに、各ラウンドの入力サイズおよび出力サイズは、すべてが相等しい必要はなく、マスキング操作は、これらの場合、ただ最低限に適合されればよい。その上、暗号化されるメモリを節約するための直接のDMAチャネルは、また、電力消費および時間の著しい節約を導き得る。

【００５２】

さらに、前に説明したように、ナンスは、実装形態に応じて、暗号化されない方法または暗号化される方法のいずれかで、メインメモリ630内に記憶され得る。あるいは、前に説明したように、ナンスは、専用のメモリの小さい保護されたエリア内に記憶され得る。

【００５３】

同様に、1つの例では、固定したキーがデバイスブートにおいてランダムに選ばれる場合、対応するキースケジュールは、その時点で事前計算され得ることを諒解されたい。特定の例として、マスターキー、または必要であればキーの中に配置され得るメモリアドレスへの依存関係が存在し得る。さらなる例として、ナンスは、固定値(その場合、関数(f1、f2、...、fL)の出力などのすべての導出される定数は、事前計算され得る)、ページごとの値であり得、または物理的なメモリアドレスに依存し得る。これらの例示的な方式は、簡略化の目的のために使用され得る。

【００５４】

メモリ内に記憶されるデータの保護のための強化型メカニズムを、ブロック暗号の機能性を拡張することによって提供するための技法は、前に説明したように、ソフトウェア、ファームウェア、ハードウェア、それらの組合せなどとして、実施され得ることを諒解されたい。一実施形態では、前に説明した機能は、あらかじめ所望される機能(たとえば、図1～図5の方法の動作)を実現するために、コンピュータ600の1つまたは複数のプロセッ

サ(たとえば、エンクリプタ620または他のプロセッサ)によって実施され得る。その上、図1～図5を参照して前に説明したように、復号は、単に逆に働く。

【００５５】

前に説明した本発明の態様は、前に説明したように、デバイスのプロセッサによる命令の実行に関連して実施され得ることを諒解されたい。詳細には、限定はしないがプロセッサを含む、デバイスの回路は、プログラム、ルーチンの制御下、または命令の実行下で動作して、本発明の実施形態による方法またはプロセスを実行することができる。たとえば、そのようなプログラムは、(たとえば、メモリおよび/または他のロケーションに記憶された)ファームウェアまたはソフトウェア中に実装され得、デバイスのプロセッサおよび/または他の回路によって実装され得る。さらに、プロセッサ、マイクロプロセッサ、回路、コントローラなどの用語は、論理、コマンド、命令、ソフトウェア、ファームウェア、機能などを実行することが可能な任意のタイプの論理または回路を指すことを諒解されたい。

【００５６】

デバイスがモバイルデバイスまたはワイヤレスデバイスであるとき、デバイスは、任意の適切なワイヤレス通信技術に基づくか、または場合によってはそれをサポートするワイヤレスネットワークにより、1つまたは複数のワイヤレス通信リンクを介して通信することができることを諒解されたい。たとえば、いくつかの態様では、ワイヤレスデバイスおよび他のデバイスは、ワイヤレスネットワークを含むネットワークと関連付けられ得る。いくつかの態様では、ネットワークは、ボディエリアネットワークまたはパーソナルエリアネットワーク(たとえば超広帯域ネットワーク)を備えることができる。いくつかの態様では、ネットワークは、ローカルエリアネットワークまたは広域ネットワークを備えることができる。ワイヤレスデバイスは、様々なワイヤレス通信技術、プロトコル、またはたとえば3G、LTE、Advanced LTE、4G、CDMA、TDMA、OFDM、OFDMA、WiMAXおよびWiFiなどの規格のうちの1つまたは複数をサポートするか、さもなければ使用することができる。同様に、ワイヤレスデバイスは、様々な対応する変調スキームまたは多重化スキームのうちの1つまたは複数をサポートするか、さもなければ使用することができる。したがって、ワイヤレスデバイスは、上記または他のワイヤレス通信技術を使用して、1つまたは複数のワイヤレス通信リンクを確立し、それを介して通信するのに適した構成要素(たとえばエアインターフェース)を含むことができる。たとえば、デバイスは、ワイヤレス媒体を介した通信を容易にする様々な構成要素(たとえば、信号発生器および信号処理器)を含むことができる、関連する送信機および受信機の構成要素(たとえば、送信機および受信機)を有するワイヤレストランシーバを備えることができる。よく知られているように、モバイルワイヤレスデバイスは、したがって、他のモバイルデバイス、携帯電話、他の有線およびワイヤレスのコンピュータ、インターネットウェブサイトなどとワイヤレスに通信することができる。

【００５７】

本明細書における教示は、様々な装置(たとえばデバイス)に組み込むことができる(たとえば様々な装置(たとえばデバイス)の中で実施され、あるいは様々な装置(たとえばデバイス)によって実行される)。たとえば、本明細書で教示する1つまたは複数の態様は、コンピュータ、有線コンピュータ、ワイヤレスコンピュータ、電話(たとえば、セルラー電話)、携帯情報端末(「PDA」)、タブレット、モバイルコンピュータ、モバイルデバイス、非モバイルデバイス、有線デバイス、ワイヤレスデバイス、ラップトップコンピュータ、エンターテインメントデバイス(たとえば、音楽デバイスもしくはビデオデバイス)、ヘッドセット(たとえば、ヘッドフォン、イヤピースなど)、医療デバイス(たとえば、生体センサ、心拍数モニタ、歩数計、EKGデバイスなど)、ユーザI/Oデバイス、固定コンピュータ、デスクトップコンピュータ、サーバ、販売時点(POS)デバイス、エンターテインメントデバイス、セットトップボックス、ATM、または任意の他の適切な電子デバイス/コンピューティングデバイスに組み込むことができる。これらのデバイスは、様々な電力要件およびデータ要件を有し得る。

【００５８】
いくつかの態様では、ワイヤレスデバイスは、通信システムのためのアクセスデバイス(たとえばWi-Fiアクセスポイント)を備えることができる。そのようなアクセスデバイスは、たとえば、有線またはワイヤレスの通信リンクを介した、別のネットワーク(たとえば、インターネットまたはセルラーネットワークなどのワイドエリアネットワーク)への接続を提供することができる。したがってアクセスデバイスは、別のデバイス(たとえばWiFi局)による他のネットワークまたは何らかの他の機能へのアクセスを可能にすることができる。
【００５９】
情報および信号は、任意の様々な異なる技術および技法を使用して表すことができることは当業者には理解されよう。たとえば上記説明全体を通して参照することができるデータ、命令、指令、情報、信号、ビット、記号およびチップは、電圧、電流、電磁波、磁界または粒子、光学場または粒子、あるいはそれらの任意の組合せによって表すことができる。
【００６０】
本明細書で開示された実施形態に関連して記載された様々な例示的な論理ブロック、モジュール、回路、およびアルゴリズムステップが、電子ハードウェア、コンピュータソフトウェア、または両方の組合せとして実装され得ることを、当業者はさらに諒解されよう。ハードウェアおよびソフトウェアのこの互換性を明確に説明するために、様々な例示的な構成要素、ブロック、モジュール、回路、およびステップが、それらの機能の点から一般的に上記で説明されている。そのような機能がハードウェアまたはソフトウェアのどちらとして実施されるのかは、システム全体に課される特定の用途および設計制約に依存する。当業者は、説明された機能性を、各特定の応用例のために様々な形で実施することができるが、そのような実施判断が、本発明の範囲からの逸脱を引き起こすと解釈されてはならない。
【００６１】
本明細書において開示される実施形態に関連して説明される様々な例示的な論理ブロック、モジュール、および回路は、汎用プロセッサ、デジタルシグナルプロセッサ(DSP)、特定用途向け集積回路(ASIC)、フィールドプログラマブルゲートアレイ(FPGA)もしくは他のプログラマブル論理デバイス、個別のゲートもしくはトランジスタ論理、個別のハードウェアコンポーネント、または本明細書において説明される機能を実行するように設計されているそれらの任意の組合せを用いて実現または実行され得る。汎用プロセッサを、マイクロプロセッサとすることができるが、代替案では、プロセッサを、任意の従来のプロセッサ、コントローラ、マイクロコントローラ、または状態機械とすることができる。プロセッサを、コンピューティングデバイスの組合せ、たとえばDSPおよびマイクロプロセッサの組合せ、複数のマイクロプロセッサ、DSPコアに関連する1つもしくは複数のマイクロプロセッサ、または任意の他のそのような構成として実施することもできる。
【００６２】
本明細書で開示される実施形態に関連して説明される方法またはアルゴリズムのステップは、ハードウェア内で直接に、プロセッサによって実行されるソフトウェアモジュール内で、またはこの2つの組合せにおいて、実施され得る。ソフトウェアモジュールは、RAMメモリ、フラッシュメモリ、ROMメモリ、EPROMメモリ、EEPROMメモリ、レジスタ、ハードディスク、取外し可能ディスク、CD-ROM、または当分野で知られている任意の他の形態の記憶媒体に常駐させることができる。例示的記憶媒体は、プロセッサが記憶媒体から情報を読み出し、かつ、記憶媒体に情報を書き込むことができるようにプロセッサに結合される。代替形態として、記憶媒体はプロセッサと一体にすることができる。プロセッサおよび記憶媒体は、ASIC内に存在してもよい。ASICは、ユーザ端末内に存在してもよい。代替で、プロセッサおよび記憶媒体は、ユーザ端末内の個別の構成要素として存在してもよい。
【００６３】

1つまたは複数の例示的実施形態では、説明される機能を、ハードウェア、ソフトウェア、ファームウェア、またはその任意の組合せで実施することができる。ディスク(disk)およびディスク(disc)は、本明細書で使用されるときに、コンパクトディスク(disc)(CD)、レーザディスク(disc)、光ディスク(disc)、デジタル多用途ディスク(disc)(DVD)、フロッピー(登録商標)ディスク(disk)、およびブルーレイディスク(disc)を含み、ディスク(disk)は、通常はデータを磁気的に再生し、ディスク(disc)は、レーザを用いてデータを光学的に再生する。前述の組合せも、コンピュータ可読媒体の範囲内に含まれるべきである。

【００６４】

開示されている実施形態についての以上の説明は、すべての当業者による本発明の構築または使用を可能にするために提供されたものである。これらの実施形態への様々な修正が当業者には容易に明らかになり、本明細書で定義する一般原理は、本発明の趣旨または範囲を逸脱することなしに他の実施形態に適用され得る。したがって、本発明は、本明細書に示される実施形態に限定されるのではなく、本明細書において開示される原理および新規の特徴に矛盾しない最も広い範囲を与えられるべきである。

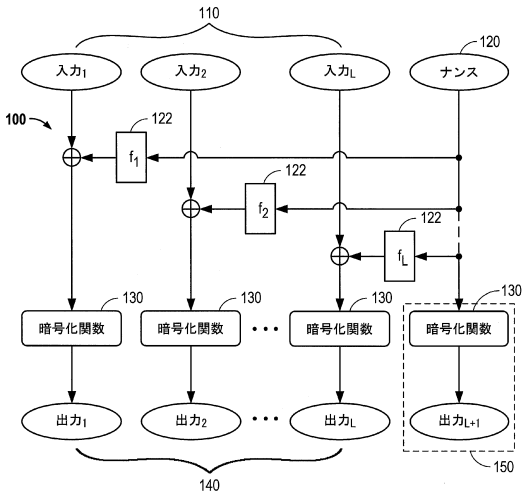
【符号の説明】

【００６５】

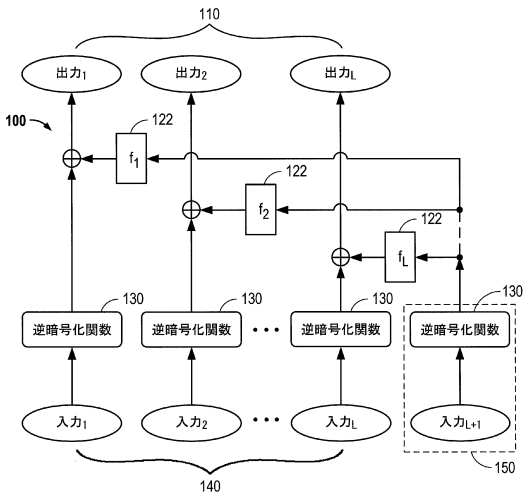
110 平文データ入力
120 ナンス
122 関数
130 暗号化関数
140 暗号化された出力データ
202 平文データ入力
204 N個のラウンド
206 出力
302 平文データ入力
304 M個のラウンド
306 ナンス
308 分離ブロック
310 N-M個のラウンド
314 連結関数
316 出力
404 M個のラウンド
406 N-M個のラウンド
410 平文データ入力
420 ナンス
422 関数
440 暗号化された出力データ
502 入力
504 M個のラウンド
506 N-M個のラウンド
520 ナンス
522 関数
540 暗号化された出力
600 コンピュータ
602 メインプロセッサ
604 他のプロセッサ
610 メモリ管理ユニット
612 キャッシュ
620 エンクリプタ処理ユニット

630　メモリ
640　記憶デバイス
650　決定ブロック
655　メモリマッピング入力/出力および制御装置

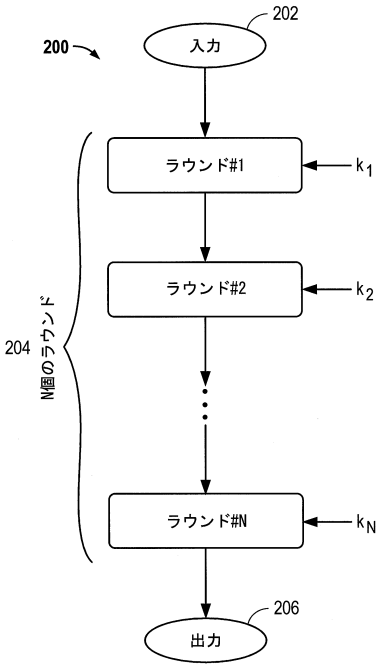
【図１Ａ】



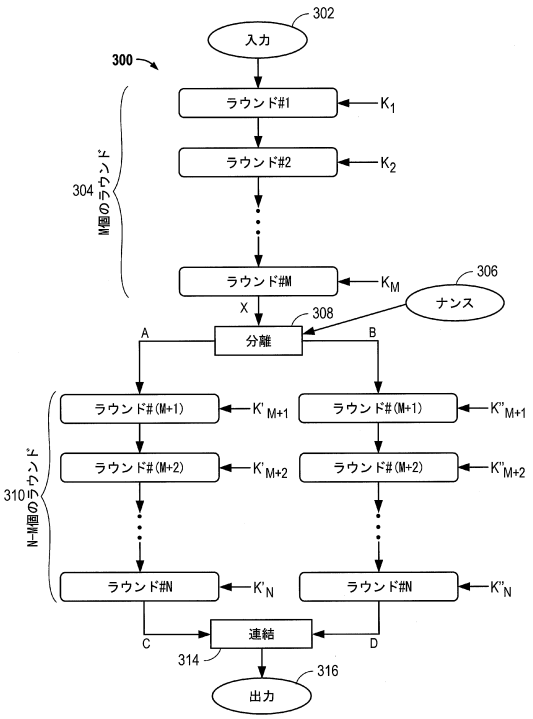
【図１Ｂ】



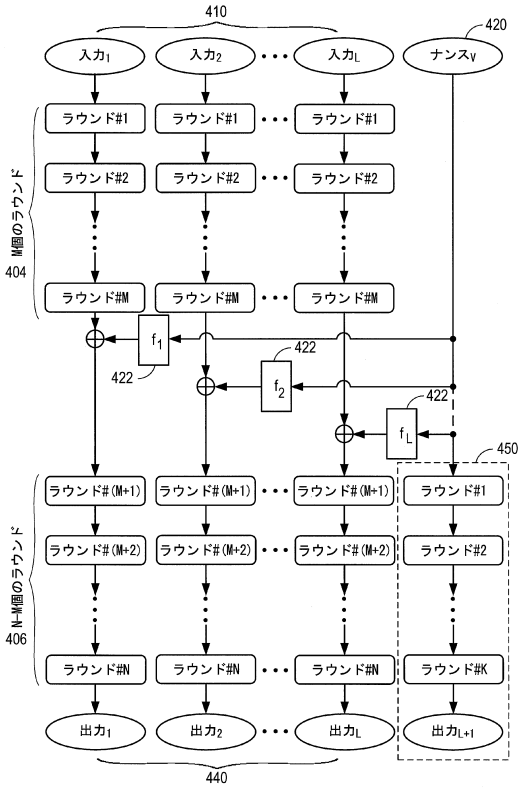
【図２】


200
202
入力
ラウンド#1
ラウンド#2
ラウンド#N
k_1
k_2
k_N
204
N個のラウンド
206
出力

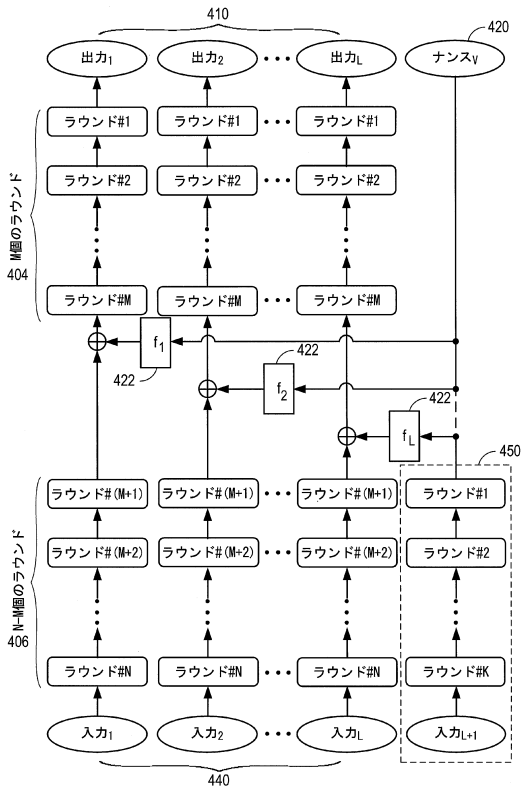
【図３】


300
302
入力
ラウンド#1
ラウンド#2
ラウンド#M
K_1
K_2
K_M
304
M個のラウンド
306
ナンス
X
308
分離
A
B
ラウンド#(M+1)
ラウンド#(M+2)
ラウンド#N
K'_{M+1}
K'_{M+2}
K'_N
K''_{M+1}
K''_{M+2}
K''_N
310
N-M個のラウンド
C
D
連結
314
316
出力

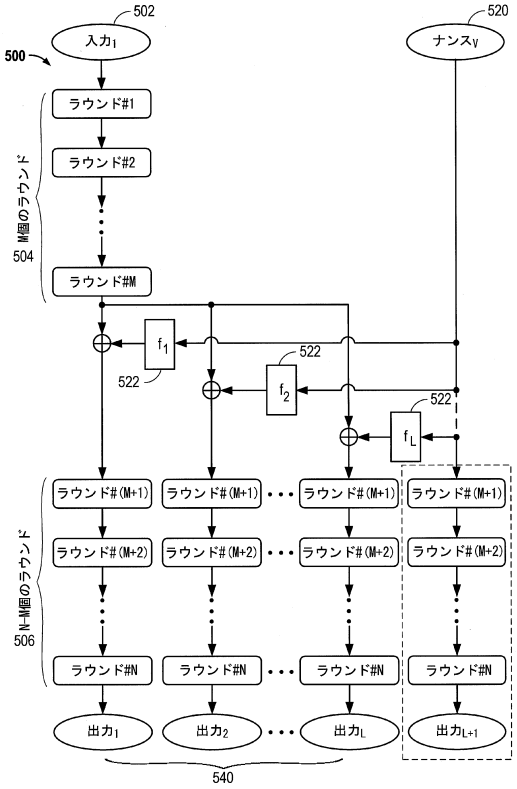
【図４Ａ】


410
420
入力1
入力2
入力L
ナンスV
ラウンド#1
ラウンド#2
ラウンド#M
404
M個のラウンド
f_1
f_2
f_L
422
450
ラウンド#(M+1)
ラウンド#(M+2)
ラウンド#N
ラウンド#K
406
N-M個のラウンド
出力1
出力2
出力L
出力L+1
440

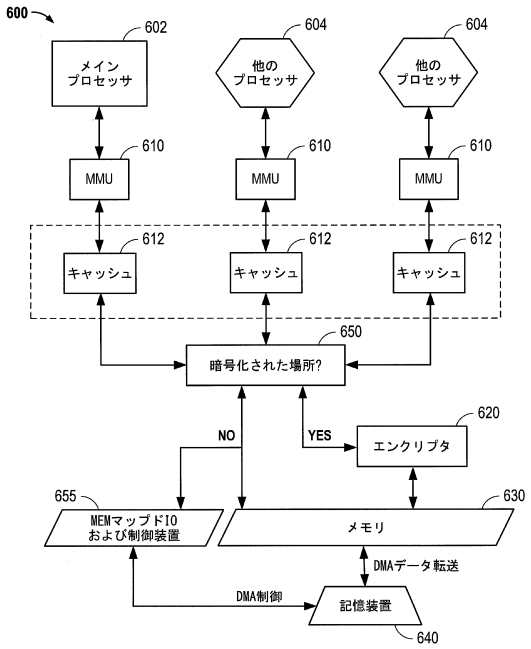
【図４Ｂ】


410
420
出力1
出力2
出力L
ナンスV
ラウンド#1
ラウンド#2
ラウンド#M
404
M個のラウンド
f_1
f_2
f_L
422
450
ラウンド#(M+1)
ラウンド#(M+2)
ラウンド#N
ラウンド#K
406
N-M個のラウンド
入力1
入力2
入力L
入力L+1
440

【図５】


500
502
入力1
520
ナンスV
ラウンド#1
ラウンド#2
ラウンド#M
M個のラウンド
504
f1
f2
fL
522
ラウンド#(M+1)
ラウンド#(M+2)
ラウンド#N
N-M個のラウンド
506
出力1
出力2
出力L
出力L+1
540

【図６】


600
602
メイン
プロセッサ
604
他の
プロセッサ
604
他の
プロセッサ
610
MMU
610
MMU
610
MMU
612
キャッシュ
612
キャッシュ
612
キャッシュ
650
暗号化された場所?
NO
YES
620
エンクリプタ
655
MEMマップドIO
および制御装置
630
メモリ
DMAデータ転送
DMA制御
記憶装置
640

フロントページの続き

(56)参考文献 特開２０１０－２３１７７８（ＪＰ，Ａ）
特開２００４－３２５６７７（ＪＰ，Ａ）
特表２０１２－５０７９４９（ＪＰ，Ａ）
国際公開第２０１２／１０８０１６（WO，Ａ１）
米国特許出願公開第２００９／０１１３２１７（ＵＳ，Ａ１）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０９Ｃ　　１／００