(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2010-185982
(P2010-185982A)
(43)公開日　平成22年8月26日(2010.8.26)

(51)Int.Cl.		F I			テーマコード（参考）
G09C 1/00 (2006.01)		G09C	1/00	610A	5J104

審査請求 未請求　請求項の数 6　ＯＬ　（全 19 頁）

(21)出願番号　特願2009-29022(P2009-29022)
(22)出願日　平成21年2月10日(2009.2.10)

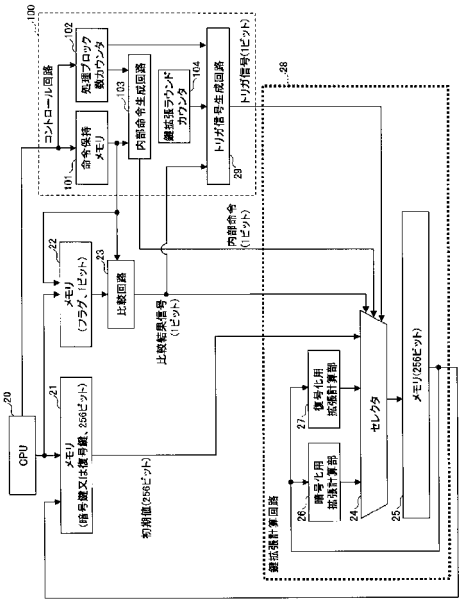
(71)出願人　309033264
東芝ストレージデバイス株式会社
東京都港区芝浦一丁目１番１号
(74)代理人　100089118
弁理士　酒井　宏明
(72)発明者　古橋　佳奈
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内

Ｆターム(参考)　5J104 AA18　JA05　PA07

(54)【発明の名称】暗号化装置、復号化装置及び記憶装置

(57)【要約】　（修正有）
【課題】暗号鍵、復号鍵保持に要するメモリ容量を低減し、暗号鍵又は復号鍵の初期値設定の際のオーバーヘッドを低減する。
【解決手段】１つの暗号鍵を拡張して得られるＮ個の拡張鍵を順次データ処理に使用する暗号化装置において、鍵の初期値と対応するフラグを保持する第１のメモリと、命令が暗号化命令でありフラグが暗号鍵を示すと比較結果信号を出力する比較回路と、比較結果信号が入力されると暗号化命令及びトリガ信号に基づいて第１のメモリに保持された鍵を初期値として第２のメモリにロードするセレクタと、第２のメモリに保持された鍵に基づいて順次拡張鍵を計算してセレクタに入力する暗号化用拡張計算部を備え、セレクタは、第２のメモリへ鍵の初期値をロードする時以外は、暗号化命令に基づいて暗号化用拡張計算部で計算された拡張鍵を第２のメモリにロードすることで暗号鍵を第１の拡張鍵～第Ｎの拡張鍵の順に拡張するように構成する。
【選択図】図４

本発明の一実施例における暗号化及び復号化装置を説明する図



【特許請求の範囲】
【請求項１】
１つの暗号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用する暗号化装置であって、
鍵の初期値と対応するフラグを保持する第１のメモリと、
命令と、前記第１のメモリに保持された前記フラグが示す鍵が共に暗号化に関するもので、前記命令が暗号化命令であり前記フラグが暗号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、
第２のメモリと、
前記比較結果信号が入力されると、前記暗号化命令及びトリガ信号に基づいて前記第１のメモリに保持された前記鍵を初期値として前記第２のメモリにロードする第１のセレクタと、
前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する暗号化用拡張計算部を備え、
前記第１のセレクタは、前記第２のメモリへ鍵の初期値をロードする時以外は、前記暗号化命令に基づいて前記暗号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記暗号鍵を第１の拡張鍵～第Ｎの拡張鍵の順に拡張する、暗号化装置。
【請求項２】
１つの復号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用する復号化装置であって、
鍵の初期値と対応するフラグを保持する第１のメモリと、
命令と、前記第１のメモリに保持された前記フラグが示す鍵が共に復号化に関するもので、前記命令が復号化命令であり前記フラグが復号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、
第２のメモリと、
前記比較結果信号が入力されると、前記復号化命令及びトリガ信号に基づいて前記第１のメモリに保持された前記鍵を初期値として前記第２のメモリにロードする第１のセレクタと、
前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する復号化用拡張計算部を備え、
前記第１のセレクタは、前記第２のメモリへ鍵の初期値をロードする時以外は、前記復号化命令に基づいて前記復号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記暗号鍵を第Ｎの拡張鍵～第１の拡張鍵の順に拡張する、復号化装置。
【請求項３】
データを記憶装置に記録し、前記記憶装置からデータを再生する制御を行う制御部と、
１つの暗号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用し、前記記憶装置に記録するデータを暗号化し、前記記憶装置から再生されたデータを復号化する暗号化及び復号化装置を備え、
前記暗号化及び復号化装置は、
鍵の初期値と対応するフラグを保持する第１のメモリと、
命令と、前記第１のメモリに保持された前記フラグが示す鍵が共に暗号化に関するもので前記命令が暗号化命令であり前記フラグが暗号鍵を示すか、或いは、共に復号化に関するもので前記命令が復号化命令であり前記フラグが復号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、
第２のメモリと、
前記比較結果信号が入力されると、前記命令及びトリガ信号に基づいて前記第１のメモリに保持された前記鍵を初期値として前記第２のメモリにロードする第１のセレクタと、
前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する暗号化用拡張計算部と、
前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタ

に入力する復号化用拡張計算部を有し、
前記第１のセレクタは、前記第２のメモリへ鍵の初期値をロードする時以外は、前記暗号化命令に基づいて前記暗号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記暗号鍵を第１の拡張鍵～第Ｎの拡張鍵の順に拡張すると共に、前記復号化命令に基づいて前記復号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記復号鍵を第Ｎの拡張鍵～第１の拡張鍵の順に拡張する、記憶装置。
【請求項４】
前記暗号化及び復号化装置は、前記比較結果信号に基づいて前記トリガ信号を生成するトリガ信号生成回路を更に有する、請求項３記載の記憶装置。
【請求項５】
前記命令を発行すると共に、鍵の初期値と対応するフラグの対を複数第３のメモリに設定するプロセッサと、
前記プロセッサが発行する鍵選択信号に基づいて鍵の初期値と対応するフラグの対を１つ前記第１のメモリに設定する第２のセレクタを更に備えた、請求項３又は４記載の記憶装置。
【請求項６】
前記比較回路での比較結果が不一致であると、前記第１のメモリに保存されている鍵の初期値を前記第２のメモリに設定し、前記暗号化用拡張計算部又は前記復号化用拡張計算部において鍵拡張を行って得られた前記第２のメモリ内の鍵の初期値を前記第１のメモリに設定し、前記第１のメモリに設定された鍵の初期値に対応するフラグを前記第１のメモリに設定することで前記第１のメモリの内容を更新する、請求項５記載の記憶装置。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、暗号化装置、復号化装置及び記憶装置に係り、特にＡＥＳ（Advanced Encryption Standard）方式を用いる暗号化装置、復号化装置及び記憶装置に関する。
【背景技術】
【０００２】
図１は、ＡＥＳ方式を用いる暗号化処理を説明する図である。ＡＥＳ方式を用いる暗号化処理では、暗号鍵スケジュールと呼ばれる、１つの暗号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用する。例えば、暗号鍵長が２５６ビットの場合、Ｎ＝１５個の拡張鍵を順次データ処理に使用する。暗号化を行う際には、このＮ個の拡張鍵を１番目からＮ番目までの順で使用する。
【０００３】
例えば、１２８ビットの平文又は暗号文に対する暗号化処理は、次のようなシーケンスで行われる。
【０００４】
Ｅ１：　２５６ビットの暗号鍵から１２８ビットの拡張鍵１を求める。
【０００５】
Ｅ２：　１２８ビットの平文と１２８ビットの拡張鍵１から１ラウンド（Round）完了時の１２８ビットのデータを計算する。
【０００６】
Ｅ３：　２５６ビットの暗号鍵若しくは１２８ビットの拡張鍵１から１２８ビットの拡張鍵２を求める。拡張鍵１及び拡張鍵２は、暗号鍵に相当する。
【０００７】
Ｅ４：　１ラウンド完了時の１２８ビットのデータと１２８ビットの拡張鍵２から２ラウンド目の１２８ビットのデータを計算する。
【０００８】
Ｅ５：　以下同様の処理をＮラウンドが完了するまで繰り返し、Ｎラウンドが完了した時点の１２８ビットのデータが暗号文となる。

【０００９】
図２は、ＡＥＳ方式を用いる復号化処理を説明する図である。復号化を行う際には、暗号化の際とは逆にＮ個の拡張鍵をＮ番目から１番目の順で使用する。
【００１０】
例えば、１２８ビットの暗号文に対する復号化処理は、次のようなシーケンスで行われる。
【００１１】
Ｄ１：　２５６ビットの復号鍵、即ち、１２８ビットの拡張鍵Ｎと１２８ビットの拡張鍵Ｎ－１を求める。拡張鍵Ｎ及び拡張鍵Ｎ－１は、復号鍵に相当する。
【００１２】
Ｄ２：　１２８ビットの暗号文と１２８ビットの拡張鍵ＮからＮ－１ラウンド完了時の１２８ビットのデータを計算する。
【００１３】
Ｄ３：　１２８ビットの拡張鍵Ｎから１２８ビットの拡張鍵Ｎ－１を求める。
【００１４】
Ｄ４：　Ｎ－１ラウンド完了時の１２８ビットのデータと１２８ビットの拡張鍵Ｎ－１からＮ－２ラウンド目の１２８ビットのデータを計算する。
【００１５】
Ｄ５：　以下同様の処理を０ラウンドが完了するまで繰り返し、０ラウンドが完了した時点の１２８ビットのデータが平文（復号文）となる。
【００１６】
このように、暗号化処理と復号化処理とでは、拡張鍵の初期値が異なる。このため、暗号化処理を実行しながら鍵拡張（所謂オン・ザ・フライ（On-The-Fly）鍵拡張）を行う際には、暗号化処理の前に拡張鍵を一旦初期化する必要がある。同様に、復号化処理を実行しながら鍵拡張を行う際には、復号化処理の前に拡張鍵を一旦初期化する必要がある。
【００１７】
一方、ＡＥＳ方式の暗号化及び復号化の処理方向は頻繁に入れ替わるため、ＣＰＵ（Central Processing Unit）等のプロセッサが、暗号化、復号化の切り替わりの度に、初期値を設定する際のオーバーヘッドがないことが望ましい。又、暗号化のモード（Cipher Modes of Operation）によっては、暗号化や復号化に使用する初期ベクトル生成の際に暗号処理を行うため、１回の起動内で復号化、暗号化、復号化といった具合に連続して処理方向が入れ替わる場合もあり、この場合であっても初期値を設定する際のオーバーヘッドがないことが望ましい。
【００１８】
図３は、従来の暗号化及び復号化装置の一例を説明する図である。ここでは説明の便宜上、暗号用鍵及び復号鍵の鍵長が２５６ビットであるものとする。暗号化及び復号化装置は、図３に示す如き鍵拡張回路と、ＡＥＳ方式の暗号化処理及び復号化処理を行うＡＥＳ方式のエンジン（図示せず）を有する。
【００１９】
先ず、ＣＰＵ１０が拡張鍵１と拡張鍵２をメモリ１１に設定すると共に、拡張鍵Ｎ－１と拡張鍵Ｎをメモリ１２に設定する。セレクタ１３は、暗号化命令又は復号化命令に基づいてメモリ１１又は１２内の拡張鍵をセレクタ１４へ選択出力する。セレクタ１３は、暗号化処理を行う時にはＣＰＵ１０からの暗号化命令に基づいて拡張鍵１と拡張鍵２（即ち、暗号鍵）をセレクタ１４へ選択出力し、復号化処理を行う時にはＣＰＵ１０からの復号化命令に基づいて拡張鍵Ｎ－１と拡張鍵Ｎ（即ち、復号鍵）をセレクタ１４へ選択出力する。
【００２０】
セレクタ１４は、暗号化命令及びメモリ１５へ初期値のロードを指示するトリガ信号に基づいて暗号鍵を初期値としてメモリ１５にロードし、復号化命令及びトリガ信号に基づいて復号鍵を初期値としてメモリ１５にロードする。暗号化用拡張計算部１６は、暗号化

処理を行う際にはメモリ１５内の暗号鍵に基づいて順次拡張鍵を計算する。復号化用拡張計算部１７は、復号化処理を行う際にはメモリ１５内の復号鍵に基づいて順次拡張鍵を計算する。セレクタ１４は、メモリ１５へ鍵の初期値をロードする時以外は、暗号化命令に基づいて暗号化用拡張計算部１６で計算された拡張鍵をメモリ１５にロードし、復号化命令に基づいて復号化用拡張計算部１７で計算された拡張鍵をメモリ１５にロードする。従って、暗号化処理を行う時には暗号鍵が拡張鍵１～Ｎの順に拡張され、復号化処理を行う時には復号鍵が拡張鍵Ｎ～１の順に拡張される。セレクタ１４、メモリ１５、暗号化用拡張計算部１６及び復号化用拡張計算部１７は、鍵拡張計算回路１８を形成する。

【００２１】

ＡＥＳ方式のエンジン（図示せず）は、暗号化命令に基づいて、平文に対してメモリ１５に格納された拡張鍵を用いて暗号化処理を行い暗号文を生成する。又、ＡＥＳ方式のエンジンは、復号化命令に基づいて、暗号文に対してメモリ１５に格納された拡張鍵を用いて復号化処理を行い平文（復号文）を生成する。

【００２２】

従来の暗号化及び復号化装置では、鍵の初期値を設定する際のオーバーヘッドを小さくするため、ＣＰＵ１０が予め用意された暗号鍵と復号鍵の２つをメモリ１１，１２に設定後、データ暗号化処理又は復号化処理に合わせて鍵スケジュールを初期化し、暗号化処理又は復号化処理を行う。このため、１つの暗号鍵に対して暗号鍵用と復号鍵用の２つのメモリ１１，１２が必要となる。

【００２３】

又、ＣＰＵ１０が予め暗号鍵と復号鍵の２つを用意する方法では、単一のＡＥＳ方式のエンジンを用いて複数の暗号鍵に対して夫々のデータの暗号化処理（又は、復号化処理）を行う暗号化及び復号化装置の場合、ＣＰＵ１０は暗号鍵の数分の暗号鍵及び復号鍵を用意する必要がある。このため、ＣＰＵ１０は、暗号鍵及び復号鍵を保持するのためにメモリ１１，１２等のメモリ中、比較的大きなメモリ容量を占有してしまう。更に、暗号鍵を変更する際には、暗号鍵と復号鍵の両方を変更する必要が生じるため、暗号鍵と復号鍵の初期値を設定する処理に時間がかかり、ＣＰＵ１０のオーバーヘッドが大きくなってしまう。

【特許文献１】特表２００７－５００３７６号公報

【特許文献２】特開２００５－４０４８号公報

【発明の開示】

【発明が解決しようとする課題】

【００２４】

従来は、暗号鍵及び復号鍵を保持するのに要するメモリ容量を低減し、且つ、暗号鍵又は復号鍵の初期値を設定する際のプロセッサのオーバーヘッドを低減することが難しいという問題があった。

【００２５】

そこで、本発明は、暗号鍵及び復号鍵を保持するのに要するメモリ容量を低減し、且つ、暗号鍵又は復号鍵の初期値を設定する際のプロセッサのオーバーヘッドを低減可能な暗号化装置、復号化装置及び記憶装置を提供することを目的とする。

【課題を解決するための手段】

【００２６】

本発明の一観点によれば、１つの暗号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用する暗号化装置であって、鍵の初期値と対応するフラグを保持する第１のメモリと、命令と、前記第１のメモリに保持された前記フラグが示す鍵が共に暗号化に関するもので、前記命令が暗号化命令であり前記フラグが暗号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、第２のメモリと、前記比較結果信号が入力されると、前記暗号化命令及びトリガ信号に基づいて前記第１のメモリに保持された前記鍵を初期値として前記第２のメモリにロードする第１のセレクタと、前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレ

クタに入力する暗号化用拡張計算部を備え、前記第１のセレクタは、前記第２のメモリへ鍵の初期値をロードする時以外は、前記暗号化命令に基づいて前記暗号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記暗号鍵を第１の拡張鍵～第Ｎの拡張鍵の順に拡張する暗号化装置が提供される。

【００２７】

本発明の一観点によれば、１つの復号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用する復号化装置であって、鍵の初期値と対応するフラグを保持する第１のメモリと、命令と、前記第１のメモリに保持された前記フラグが示す鍵が共に復号化に関するもので、前記命令が復号化命令であり前記フラグが復号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、第２のメモリと、前記比較結果信号が入力されると、前記復号化命令及びトリガ信号に基づいて前記第１のメモリに保持された前記鍵を初期値として前記第２のメモリにロードする第１のセレクタと、前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する復号化用拡張計算部を備え、前記第１のセレクタは、前記第２のメモリへ鍵の初期値をロードする時以外は、前記復号化命令に基づいて前記復号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記暗号鍵を第Ｎの拡張鍵～第１の拡張鍵の順に拡張する復号化装置が提供される。

【００２８】

本発明の一観点によれば、データを記憶装置に記録し、前記記憶装置からデータを再生する制御を行う制御部と、１つの暗号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用し、前記記憶装置に記録するデータを暗号化し、前記記憶装置から再生されたデータを復号化する暗号化及び復号化装置を備え、前記暗号化及び復号化装置は、鍵の初期値と対応するフラグを保持する第１のメモリと、命令と、前記第１のメモリに保持された前記フラグが示す鍵が共に暗号化に関するもので前記命令が暗号化命令であり前記フラグが暗号鍵を示すか、或いは、共に復号化に関するもので前記命令が復号化命令であり前記フラグが復号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、第２のメモリと、前記比較結果信号が入力されると、前記命令及びトリガ信号に基づいて前記第１のメモリに保持された前記鍵を初期値として前記第２のメモリにロードする第１のセレクタと、前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する暗号化用拡張計算部と、前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する復号化用拡張計算部を有し、前記第１のセレクタは、前記第２のメモリへ鍵の初期値をロードする時以外は、前記暗号化命令に基づいて前記暗号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記暗号鍵を第１の拡張鍵～第Ｎの拡張鍵の順に拡張すると共に、前記復号化命令に基づいて前記復号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記復号鍵を第Ｎの拡張鍵～第１の拡張鍵の順に拡張する記憶装置が提供される。

【発明の効果】

【００２９】

開示の暗号化装置、復号化装置及び記憶装置によれば、暗号鍵及び復号鍵を保持するのに要するメモリ容量を低減し、且つ、暗号鍵又は復号鍵の初期値を設定する際のプロセッサのオーバーヘッドを低減することが可能となる。

【発明を実施するための最良の形態】

【００３０】

開示の暗号化装置、復号化装置及び記憶装置は、鍵の初期値と対応するフラグを保持する第１のメモリと、命令と第１のメモリに保持されたフラグが示す鍵が共に暗号化に関するもので命令が暗号化命令でありフラグが暗号鍵を示すか、或いは、共に復号化に関するもので命令が復号化命令でありフラグが復号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、第２のメモリと、比較結果信号が入力されると命令及びトリガ信号に基づいて第１のメモリに保持された鍵を初期値として第２のメモリにロ

ードするセレクタと、第２のメモリに保持された鍵に基づいて順次拡張鍵を計算してセレクタに入力する暗号化用拡張計算部と、第２のメモリに保持された鍵に基づいて順次拡張鍵を計算してセレクタに入力する復号化用拡張計算部を有する。

【００３１】

セレクタは、第２のメモリへ鍵の初期値をロードする時以外は、暗号化命令に基づいて暗号化用拡張計算部で計算された拡張鍵を第２のメモリにロードすることで暗号鍵を第１の拡張鍵～第Ｎの拡張鍵の順に拡張すると共に、復号化命令に基づいて復号化用拡張計算部で計算された拡張鍵を第２のメモリにロードすることで復号鍵を第Ｎの拡張鍵～第１の拡張鍵の順に拡張する。

【００３２】

これにより、暗号鍵及び復号鍵を保持するのに要するメモリ容量を低減し、且つ、暗号鍵又は復号鍵の初期値を設定する際のプロセッサのオーバーヘッドを低減可能となる。

【００３３】

以下に、本発明の暗号化装置、復号化装置及び記憶装置の各実施例を、図４以降と共に説明する。

【実施例】

【００３４】

図４は、本発明の一実施例における暗号化及び復号化装置を説明する図である。ここでは説明の便宜上、暗号用鍵及び復号鍵の鍵長が２５６ビットであるものとする。暗号化及び復号化装置は、図４に示す如き鍵拡張回路と、ＡＥＳ方式の暗号化処理及び復号化処理を行うＡＥＳ方式のエンジン（図示せず）を有する。ＡＥＳ方式のエンジンについては後述する。

【００３５】

先ず、ＣＰＵ２０が拡張鍵１と拡張鍵２、即ち、暗号鍵の初期値をメモリ２１に設定すると共に、メモリ２１に設定された拡張鍵１と拡張鍵２が暗号鍵であることを示す１ビットのフラグをメモリ２２に設定する。或いは、ＣＰＵ２０が拡張鍵Ｎ－１と拡張鍵Ｎ、即ち、復号鍵の初期値をメモリ２１に設定すると共に、メモリ２１に設定された拡張鍵Ｎ－１と拡張鍵Ｎが復号鍵であることを示す１ビットのフラグをメモリ２２に設定する。メモリ２１は鍵スケジュールの初期値を保持できる大きさを有するものであれば良く、又、メモリ２２は１ビットのフラグを保持できる大きさのものであれば良い。メモリ２１は、ＣＰＵ２０が設定した鍵スケジュールの初期値、若しくは、最後にＡＥＳ方式のエンジンが使用した鍵スケジュールの初期値を保持し、メモリ２２はメモリ２１の状態（暗号鍵の初期値と復号鍵の初期値のどちらを保持しているか）を示すフラグを保持する。

【００３６】

暗号化処理は、ＣＰＵ２０が発行する暗号化命令に応答して開始される。又、復号化処理は、ＣＰＵ２０が発行する復号化命令に応答して開始される。ＣＰＵ２０が発行した暗号化命令又は復号化命令は、コントロール回路１００に供給される。

【００３７】

コントロール回路１００は、命令保持メモリ１０１、処理ブロック数カウンタ１０２、内部命令生成回路１０３、鍵拡張ラウンドカウンタ１０４及びトリガ信号生成回路２９を有する。

【００３８】

命令保持メモリ１０１は、ＣＰＵ２０が発行した命令が暗号化命令であるか復号化命令であるかを認識できるように、ＣＰＵ２０が発行した命令を保持する。処理ブロック数カウンタ１０２は、ＣＰＵ２０が発行した命令に基づいて例えば１２８ビットの暗号化処理又は復号化処理が完了する度にインクリメントされることで、処理ブロックを０からＭま（Ｍは２以上の自然数）でカウントするものであり、カウント値がＭに達するとカウント値が０に初期化される。内部命令生成回路１０３は、処理ブロック数カウンタ１０２のカウント値が１からＭ－１までの場合は、命令保持メモリ１０１に保持された命令を実行し、処理ブロックカウンタ１０２のカウント値がＭの場合は命令保持メモリ１０１に保持さ

れた命令と逆の命令を実行する１ビットの内部命令を生成する。

【００３９】

鍵拡張ラウンドカウンタ１０４は、拡張鍵１～Ｎ（即ち、第１の拡張鍵～第Ｎの拡張鍵）のラウンドをカウントする。トリガ信号生成回路２９は、処理ブロック数カウンタ１０２のカウント値が０からＭ－１までの場合は、鍵拡張ラウンドカウンタ１０４でカウントされた鍵拡張のＮラウンド毎、即ち、例えば１２８ビットの復号化処理の度に比較回路２３の出力に応答してトリガ信号を生成して後述する鍵拡張計算回路２８内のセレクタ２４に出力される。一方、処理ブロック数カウンタ１０２のカウント値がＭの場合は、トリガ信号生成回路２９が生成するトリガ信号はマスクされてセレクタ２４には出力されない。

【００４０】

比較回路２３は、暗号化処理を行う時にはＣＰＵ２０から命令保持メモリ１０１を介して得られる１ビットの暗号化命令とメモリ２２に保持されたフラグが示す鍵が共に暗号化に関するものであると、比較したビットが一致することを示す１ビットの比較結果信号をセレクタ２４へ出力する。又、比較回路２３は、復号化処理を行う時にはＣＰＵ２０から命令保持メモリ１０１を介して得られる１ビットの復号化命令とメモリ２２に保持されたフラグが示す鍵が共に復号化に関するものであると、比較したビットが一致することを示す比較結果信号をセレクタ２４へ出力する。

【００４１】

暗号化処理を行う時、セレクタ２４は、一致を示す比較結果信号が入力されると、暗号化命令及びメモリ２５へ初期値のロードを指示する１ビットのトリガ信号に基づいてメモリ２１に保持された暗号鍵を初期値としてメモリ２５にロードし、メモリ２１，２２の保持内容は変更されない。トリガ信号は、上記の如く比較回路２３の出力に応答してトリガ信号生成回路２９から出力される。暗号化用拡張計算部２６は、暗号化処理を行う際にはメモリ２５内の暗号鍵に基づいて順次拡張鍵１～Ｎ（即ち、第１の拡張鍵～第Ｎの拡張鍵）を計算する。セレクタ２４は、メモリ２５へ鍵の初期値をロードする時以外は、暗号化命令に基づいて暗号化用拡張計算部２６で計算された拡張鍵をメモリ２５にロードする。従って、暗号化処理を行う時には暗号鍵が拡張鍵１～Ｎの順に拡張される。

【００４２】

復号化処理を行う時、セレクタ２４は、一致を示す比較結果信号が入力されると、復号化命令及びトリガ信号に基づいて復号鍵を初期値としてメモリ２５にロードし、メモリ２１，２２の保持内容は変更されない。復号化用拡張計算部２７は、復号化処理を行う際にはメモリ２５内の復号鍵に基づいて順次拡張鍵Ｎ～１（即ち、第Ｎの拡張鍵～第１の拡張鍵）を計算する。セレクタ２４は、メモリ２５へ鍵の初期値をロードする時以外は、復号化命令に基づいて復号化用拡張計算部２７で計算された拡張鍵をメモリ２５にロードする。従って、復号化処理を行う時には復号鍵が拡張鍵Ｎ～１の順に拡張される。

【００４３】

セレクタ２４、メモリ２５、暗号化用拡張計算部２６及び復号化用拡張計算部２７は、鍵拡張計算回路２８を形成する。

【００４４】

図３に示す従来の装置では、ＣＰＵ１０が暗号鍵と復号鍵の両方をメモリ１１，１２に設定していたのに対し、本実施例では、ＣＰＵ２０は暗号鍵又は復号鍵の一方と、鍵が暗号鍵であるか復号鍵であるかを示す１ビットのフラグをメモリ２１，２２設定すれば良いため、鍵の初期値を設定するための時間が従来の装置と比べて略半分になると共に、鍵の初期値を設定するメモリ１１，１２のメモリ容量も従来の装置のメモリ１１，１２と比べて約半分になる。

【００４５】

更に、比較回路２３は、ＣＰＵ２０からの命令とメモリ２２に保持されたフラグが示す鍵のうち、一方が暗号化に関するものであり他方が復号化に関するもので互いに異なるものであると、比較したビットが不一致であることを示す比較結果信号をセレクタ２４へ出力する。

【００４６】
セレクタ２４は、不一致を示す比較結果信号が入力されると、ＣＰＵ２０からの命令が暗号化命令であればトリガ信号に基づいてメモリ２１に保持された復号鍵を初期値としてメモリ２５にロードする。復号化用拡張計算部２７は、メモリ２５内の復号鍵に基づいて順次拡張鍵を計算し、復号鍵が拡張鍵Ｎ～１の順に拡張されて暗号鍵が得られる。得られた暗号鍵は、メモリ２５からメモリ２１へ設定され、メモリ２２へはメモリ２１に設定された鍵が暗号鍵であることを示すフラグがＣＰＵ２０から設定されることで、メモリ２１，２２の内容が暗号化処理用の内容に更新される。
【００４７】
セレクタ２４は、不一致を示す比較結果信号が入力されると、ＣＰＵ２０からの命令が復号化命令であればトリガ信号に基づいてメモリ２１に保持された暗号鍵を初期値としてメモリ２５にロードする。暗号化用拡張計算部２６は、メモリ２５内の暗号鍵に基づいて順次拡張鍵を計算し、暗号鍵が拡張鍵１～Ｎの順に拡張されて復号鍵が得られる。得られた復号鍵は、メモリ２５からメモリ２１へ設定され、メモリ２２へはメモリ２１に設定された鍵が復号鍵であることを示すフラグがＣＰＵ２０から設定されることで、メモリ２１，２２の内容が復号化処理用の内容に更新される。
【００４８】
つまり、比較回路２３での比較結果が不一致であると、メモリ２２に保存されている鍵の初期値をメモリ２５に設定し、暗号化用拡張計算部２６又は復号化用拡張計算部２７において鍵拡張を行って得られたメモリ２５内の鍵の初期値をメモリ２２に設定し、メモリ２２に設定された鍵の初期値に対応するフラグをメモリ２２に設定することでメモリ２２の内容を更新する。このように、メモリ２２のフラグの更新は、ＣＰＵ２０が設定する必要はなく、メモリ２１の鍵の初期値を更新するタイミングで命令保持メモリ１０１を介して得られる１ビットの命令により自動的に行うことができる。
【００４９】
このように、本実施例では、ＣＰＵ２０からの命令とメモリ２２に保持されたフラグが示す鍵のうち、一方が暗号化に関するものであり他方が復号化に関するもので互いに異なるものであると、鍵の初期値を設定するために暗号化用拡張計算部２６又は復号化用拡張計算部２７において一度鍵拡張を行う必要が生じるが、鍵長が２５６ビットの場合でも必要となる鍵拡張は１４サイクルで完了可能である。周知の構成を有する暗号化用拡張計算部２６及び周知の構成を有する復号化用拡張計算部２７の演算速度は、いずれもＣＰＵ２０と比べて高速であるため、このような鍵の初期値を設定のための鍵拡張はＣＰＵ２０のオーバーヘッドとしては見えてこない。従って、鍵の初期値を設定するための鍵拡張によりＣＰＵ２０のオーバーヘッドが従来の装置と比べて大きくなることはない。
【００５０】
ＡＥＳ方式のエンジン（図示せず）は、暗号化命令に基づいて、平文に対してメモリ２５に格納された拡張鍵を用いて暗号化処理を行い暗号文を生成する。又、ＡＥＳ方式のエンジンは、復号化命令に基づいて、暗号文に対してメモリ２５に格納された拡張鍵を用いて復号化処理を行い平文（復号文）を生成する。
【００５１】
尚、暗号化処理の直後に復号化処理を行う場合は、メモリ２５に保持されている暗号化でＮ番目に使用された拡張鍵を次に行われる復号鍵の初期値として使用することにより、復号化処理のオーバーヘッドを低減可能である。同様に、復号化処理の直後に暗号化処理を行う場合は、メモリ２５に保持されている復号化処理でＮ番目に使用された拡張鍵を次に行われる暗号鍵の初期値として使用することにより、暗号化処理のオーバーヘッドを低減可能である。
【００５２】
又、メモリ２１，２２は、別体のメモリである必要はなく、単一のメモリ内の異なるメモリ領域により形成されていても良い。又、暗号鍵又は復号鍵の初期値とフラグを、１つのデータとして扱うようにしても良い。メモリ２１をＣＰＵ２０からアクセス可能とする

ことで、ＣＰＵ２０は暗号鍵又は復号鍵を実暗号鍵ビット長＋１ビット（フラグ）として扱うこともできる。この場合、ＣＰＵ２０及び鍵拡張計算回路２８は共に、暗号鍵又は復号鍵（暗号化用の拡張鍵＋１ビット、若しくは、復号化用の拡張鍵＋１ビット）を暗号化用と復号化用の兼用の鍵として使用することができる。特に複数の暗号鍵を使用する暗号化及び復号化装置では、単一のＡＥＳ方式のエンジンを用いて暗号化処理及び復号化処理を行う場合に、記憶しておく鍵が暗号鍵若しくは復号鍵のどちらか一方で良いため、鍵の初期値を設定するのに要するメモリ容量を従来に比べて低減可能であると共に、設定する鍵の初期値の鍵長も暗号鍵＋１ビット又は復号鍵＋１ビットで良いため、鍵の初期値を設定するためのオーバヘッドを従来に比べて小さくすることが可能である。

【００５３】

図５は、記憶装置を示すブロック図である。図５は、上記実施例が単一のＡＥＳ方式のエンジンを用いる記憶装置に適用された場合を示す。図５に示す記憶装置３０は、ＣＰＵ２０、メモリ３１、セレクタ３２、メモリ３３、鍵拡張ブロック３４、ＡＥＳ方式のエンジン（以下、単にＡＥＳエンジンと言う）３５、ヘッド３６及びディスク３７を有する。

【００５４】

メモリ３３は、図４に示すメモリ２１，２２に相当する。鍵拡張ブロック３４は、図４に示す比較回路２３、トリガ信号生成回路２９及び鍵拡張計算回路２８に相当するが、コントロール回路１００の他の部分を含むものであっても良い。ヘッド３６は、制御部として機能するＣＰＵ２０の制御下で、ディスク３７に情報を記録すると共に、ディスク３７に記録された情報を再生する。ディスク３７は、磁気ディスク、光ディスクや光磁気ディスク等の記録媒体であっても良い。ディスク３７が例えば磁気ディスクの場合、ヘッド３６は磁気ディスク上を所定の浮上量を維持しながら走査するよう移動及び制御されるが、ヘッド３６の移動及び制御機構自体はＨＤＤ（Hard Disk Drive）の分野では周知であるため、その図示及び説明は省略する。又、ヘッド３６及びディスク３７の数は夫々複数であっても良い。

【００５５】

本実施例では、データの記録再生に使用される記憶装置は、ヘッド３６とディスク３７を有するディスク装置で形成されているが、記憶装置はヘッドを用いる装置に限定されず、フラッシュメモリ等の半導体記憶装置をデータの記録再生に使用しても良いことは言うまでもない。半導体記憶装置等をデータの記録再生に使用する場合も、データの記憶装置への記録及びデータの記憶装置からの再生は、制御部として機能するＣＰＵ２０の制御下で行われる。

【００５６】

図５の例では、記憶装置３０は３種類の鍵ｋ１，ｋ２，ｋ３から鍵を選択可能であるが、選択可能な鍵の数は３種類に限定されるものではない。ＣＰＵ２０は、セレクタ３２に鍵選択信号を出力することで、メモリ３１に保持されている鍵ｋ１，ｋ２，ｋ３と対応するフラグｆ１，ｆ２，ｆ３のうち選択された鍵と対応するフラグをメモリ３３に設定する。鍵拡張ブロック３４は、ＣＰＵ２０からの命令とメモリ３３に保持されたフラグのビットが一致すると、トリガ信号生成回路２９からのトリガ信号に基づいてメモリ３３に保持された鍵を初期値としてメモリ２５にロードする。メモリ３３に保持された鍵が暗号鍵の場合は暗号化用拡張計算部２６が順次拡張鍵１～Ｎを計算し、復号鍵の場合は復号化用拡張計算部２７が順次拡張鍵Ｎ～１を計算する。

【００５７】

一方、鍵拡張ブロック３４は、ＣＰＵ２０からの命令とメモリ３３に保持されたフラグのビットが不一致であると、ＣＰＵ２０からの命令及びトリガ信号生成回路２９からのトリガ信号に基づいてメモリ３３に保持された鍵を初期値としてメモリ２５にロードする。メモリ３３に保持された鍵が復号鍵の場合は復号化用拡張計算部２７が復号鍵に基づいて順次拡張鍵Ｎ～１を計算し、暗号鍵の場合は暗号化用拡張計算部２６が暗号鍵に基づいて順次拡張鍵１～Ｎを計算する。これにより、得られた暗号鍵又は復号鍵は、メモリ２５からメモリ３３へ設定されると共に、メモリ３３に設定された鍵に対応するフラグが設定さ

れることで、メモリ３３の内容が更新される。

【００５８】

ＡＥＳエンジン３５は、暗号化命令に基づいて、ホスト装置等の外部装置（図示せず）から記憶装置３０に入力された平文に対して鍵拡張ブロック３４内のメモリ２５に格納された拡張鍵を用いて図１に示す如き暗号化処理を行い暗号文を生成する。生成された暗号文は、ヘッド３６によりディスク３７に記録される。又、ＡＥＳエンジン３５は、復号化命令に基づいて、ヘッド３６によりディスク３７から再生された暗号文に対して鍵拡張ブロック３４内のメモリ２５に格納された拡張鍵を用いて図２に示す如き復号化処理を行い平文（復号文）を生成する。生成された平文は、記憶装置３０からホスト装置等の外部装置へ出力される。ＡＥＳエンジン３５自体は、周知の構成を有するものである。

【００５９】

記憶装置３０がＨＤＤの場合、暗号化処理及び復号化処理では同じ鍵を用いて例えば数メガビット（Mbit）分の連続処理が実行される。このような連続処理は、上記の如き暗号化処理及び復号化処理を繰り返し実行することにより実現される。

【００６０】

次に、本実施例における拡張鍵の生成シーケンスを、図６と共に説明する。図６は、拡張鍵の生成シーケンスを説明するフローチャートである。図６において、ステップＳ１，Ｓ２はＣＰＵ２０により実行され、ステップＳ１１～Ｓ１７は鍵拡張ブロック３４により実行される。

【００６１】

ＣＰＵ２０では、ステップＳ１が暗号鍵又は復号鍵と対応するフラグをメモリ３３に設定する。これにより、鍵拡張ブロック３４では、ステップＳ１１が暗号鍵又は復号鍵と対応するフラグがメモリ３３に保持する。ＣＰＵ２０では、ステップＳ２で暗号化命令を発行して暗号化処理を開始するか、或いは、復号化命令を発行して復号化処理を開始する。鍵拡張ブロック３４では、ステップＳ１２がＣＰＵ２０が発行した命令とメモリ３３内のフラグを比較し、ステップＳ１３が比較結果が一致するか否かを判定する。ステップＳ１３の判定結果がＹＥＳであると処理はステップＳ１４へ進み、ＮＯであると処理はステップＳ１６へ進む。

【００６２】

ステップＳ１３の判定結果がＹＥＳであると、鍵拡張ブロック３４では、ステップＳ１４がトリガ信号生成回路２９によりトリガ信号を生成し、このトリガ信号に応答してメモリ３３に保持された鍵をメモリ２５に保持し、ＣＰＵ２０からの命令が暗号化命令であれば暗号化用拡張計算部２６で１２８ビットのデータの暗号化用拡張計算を行い、命令が復号化命令であれば復号化用拡張計算部２７で１２８ビットのデータの復号化拡張計算を行う。この際、ＡＥＳエンジン３５では、各拡張鍵を用いてデータの暗号化又は復号化を行い、対応するラウンド完了時のデータを求める。鍵拡張ブロック３４では、ステップＳ１５が暗号化処理又は復号化処理をＮ回連続して行ったか否かを判定し、判定結果がＮＯであると処理はステップＳ１４へ戻り、ＹＥＳであると処理は終了する。

【００６３】

一方、ステップＳ１３の判定結果がＮＯであると、鍵拡張ブロック３４では、ステップＳ１６がトリガ信号生成回路２９によりトリガ信号を生成し、このトリガ信号に応答してメモリ３３に保持された鍵をメモリ２５に保持し、ＣＰＵ２０からの命令が暗号化命令であれば復号化用拡張計算部２７で１２８ビットのデータの復号化用拡張計算を行い、命令が復号化命令であれば暗号化用拡張計算部２６で１２８ビットのデータの暗号化用拡張計算を行う。又、ステップＳ１６は、計算された暗号鍵又は復号鍵をメモリ３３に保持すると共に、メモリ３３に保持された暗号鍵又は復号鍵に対応するフラグをメモリ３３に保持する。この際、ＡＥＳエンジン３５では、各拡張鍵を用いたデータの暗号化又は復号化を行わない。

【００６４】

ところで、復号化処理の直後に一度暗号化処理を行うシーケンスの一例としては、ブロ

ック・サイファー・モード・オブ・オペレーション（Block Cipher Modes of Operation）で用いられる場合がある。例えば、ＣＢＣ（Cipher Block Chain）モードでは、ノンスワード（Nonce Word）を同じ鍵で暗号化した結果を初期化ベクトル（Initialization Vector）として使用する方法が推奨されている。初期化ベクトルとは、ＣＢＣモードを開始する際に最初のデータ処理に使用される初期値であり、１２８ビットのデータ処理の場合は初期値も１２８ビットである。この推奨されている方法を、ＨＤＤのようなセクタ単位に暗号化処理又は復号化処理を行い、連続セクタライトや連続セクタリードを行う記憶装置に適用し復号化を行う場合、セクタの最後の１２８ビットのデータの復号化後に、１２８ビットのノンスワードの暗号化を行って初期化ベクトルを生成し、次のセクタの先頭の１２８ビットのデータの復号化を行うという一連の処理を、ＣＰＵ２０の介在なしで連続して行うことが必要となる。一方、暗号化を行う場合は、初期化ベクトルの暗号化、セクタの暗号化を行うことになるため、ＡＥＳは常に暗号化を行うことになる。暗号化の説明はここでは省略する。

【００６５】

次に、１セクタの処理を、１回のノンスワードの暗号化処理とM回の連続した復号化処理で行うものとしたときに、ＣＰＵ２０の介在なしにＬセクタの処理を行う場合のシーケンスを図７と共に説明する。図７は、このようにＣＰＵ２０の介在なしにＬセクタの処理を行う場合のシーケンスを説明するフローチャートである。

【００６６】

図７において、ステップＳ２１では、初期条件が設定される。初期条件が設定された状態では、メモリ３３（又は、メモリ２１，２２）の内容は復号化用に初期化されており、図６のステップＳ１３の判定結果がＹＥＳ、或いは、ステップＳ１６が完了している、この状態では、既に復号化が１回以上行われ、メモリ２５には暗号化用の拡張鍵が保持されているものとする。ステップＳ２２では、トリガ信号生成回路２９からトリガ信号を出力せず、メモリ２５の保持値をそのまま拡張鍵の初期値とし、初期化ベクトルを暗号化するために生成される内部命令に従い暗号化用拡張計算部２６で暗号化用拡張計算による鍵拡張を行うと共に、ＡＥＳエンジン３５によるノンスワードの暗号化も同時に行う。ステップＳ２３では、トリガ信号生成回路２９からトリガ信号を出力し、拡張鍵の初期値をメモリ２５に保持し、データを復号化するために生成される内部命令に従い復号化用拡張計算部２７で復号化用拡張計算による鍵拡張を行うと共に、ＡＥＳエンジン３５によるデータの復号化も同時に行う。

【００６７】

上記の如く、内部命令は、処理ブロック数をカウントする処理ブロックカウンタ１０２のカウント値が０からＭ－１までの場合はＣＰＵ２０が設定した復号化命令を実行し、カウント値がMの場合は暗号化命令を実行する。又、処理ブロックカウンタ１０２は、１２８ビットの暗号化処理又は復号化処理が完了する度にインクリメントされ、カウント値がMに達すると０に初期化される。トリガ信号生成回路２９が生成するトリガ信号は、処理ブロック数カウンタ１０２の値が０からＭ－１までの場合は鍵拡張ラウンドカウンタ１０４でカウントされた鍵拡張のＮラウンド毎、即ち、１２８ビットの復号化処理の度に比較回路２３の出力に応答して出力される。処理ブロック数カウンタ１０２のカウント値がMの間は、トリガ信号生成回路２９が生成するトリガ信号はマスクされて出力されない。

【００６８】

ステップＳ２４では、ＡＥＳエンジン３５において復号化をM回行ったか否かを判定し、判定結果がＮＯであると処理はステップＳ２３へ戻る。ステップＳ２４の判定結果がＹＥＳであると、ステップＳ２５では、ＡＥＳエンジン３５においてＬセクタまでの処理が終了したか否かを判定し、判定結果がＮＯであると処理はステップＳ２２へ戻る。ステップＳ２５の判定結果がＹＥＳであると、処理は終了する。

【００６９】

次に、ブロック・サイファー・モード・オブ・オペレーションについて、図８～図１１と共に説明する。

【００７０】
一般的に、１２８ビット以上のデータを同じ暗号鍵で暗号化していくときには、ブロック・サイファー・モード・オブ・オペレーションという手法を用いる。良く使用されるモード（Mode）の一例としてＣＢＣモードと呼ばれるノードがあり、ＣＢＣモードでは１セクタの処理は暗号化処理の場合は図８に示すシーケンスで行われ、復号化処理の場合は図９に示すシーケンスで行われる。
【００７１】
図８は、ＣＢＣモードの暗号化処理を説明する図である。図８において、データＤ１は、初期化ベクトルとの排他的論理和（ＸＯＲ：eXclusive-OR）を取られてＡＥＳ方式の暗号化を施され、暗号文Ｅ１が得られる。データＤ２は、暗号文Ｅ１とのＸＯＲを取られてＡＥＳ方式の暗号化を施され、暗号文Ｅ２が得られる。以下同様の処理が行われ、最終的には、データＤＭは、暗号文ＥＭ－１とのＸＯＲを取られてＡＥＳ方式の暗号化を施され、暗号文ＥＭが得られる。
【００７２】
図９は、ＣＢＣモードの復号化処理を説明する図である。図９において、暗号文Ｅ１は、ＡＥＳ方式の復号化を施され、初期化ベクトルとのＸＯＲを取られてデータＤ１が得られる。暗号文Ｅ２は、ＡＥＳ方式の復号化を施され、暗号文Ｅ１とのＸＯＲを取られてデータＤ２が得られる。以下同様の処理が行われ、最終的には、暗号文ＥＭは、ＡＥＳ方式の復号化を施され、暗号文ＥＭ－１とのＸＯＲを取られてデータＤＭが得られる。
【００７３】
又、推奨されている初期化ベクトルの生成方法の一例として、ノンスワードを暗号鍵で暗号化して使用する方法が挙げられる。この場合、ＣＢＣモードの１セクタの処理は暗号化処理の場合は図１０に示すシーケンスで行われ、復号化処理の場合は図１１に示すシーケンスで行われる。
【００７４】
図１０は、ノンスワードを暗号鍵で暗号化して初期化ベクトルとして使用する場合のＣＢＣモードの暗号化処理を説明する図である。図１０において、ノンスワードは、ＡＥＳ方式の暗号化を施されて初期化ベクトルとされる。初期ベクトルは、図８の場合と同様に暗号化に使用される。
【００７５】
図１１は、ノンスワードを暗号鍵で暗号化して初期化ベクトルとして使用する場合のＣＢＣモードの復号化処理を説明する図である。図１１において、ノンスワードは、ＡＥＳ方式の暗号化を施されて初期化ベクトルとされる。初期ベクトルは、図９の場合と同様に復号化に使用される。
【００７６】
以上の実施例を含む実施形態に関し、更に以下の付記を開示する。
（付記１）
１つの暗号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用する暗号化装置であって、
鍵の初期値と対応するフラグを保持する第１のメモリと、
命令と、前記第１のメモリに保持された前記フラグが示す鍵が共に暗号化に関するもので、前記命令が暗号化命令であり前記フラグが暗号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、
第２のメモリと、
前記比較結果信号が入力されると、前記暗号化命令及びトリガ信号に基づいて前記第１のメモリに保持された前記鍵を初期値として前記第２のメモリにロードする第１のセレクタと、
前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する暗号化用拡張計算部を備え、
前記第１のセレクタは、前記第２のメモリへ鍵の初期値をロードする時以外は、前記暗

号化命令に基づいて前記暗号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記暗号鍵を第１の拡張鍵～第Ｎの拡張鍵の順に拡張する、暗号化装置。
（付記２）
前記比較結果信号に基づいて前記トリガ信号を生成するトリガ信号生成回路を更に備えた、付記１記載の暗号化装置。
（付記３）
前記命令を発行すると共に、前記鍵の初期値と対応する前記フラグを第１のメモリに設定するプロセッサを更に備えた、付記１又は２記載の暗号化装置。
（付記４）
前記第１のメモリは、前記鍵の初期値を保持すると共に、保持する前記鍵の初期値を前記第１のセレクタへ出力するメモリと、前記フラグを保持すると共に、保持する前記フラグを前記比較回路へ出力するメモリを有する、付記１乃至３のいずれか１項記載の暗号化装置。
（付記５）
前記命令を発行すると共に、鍵の初期値と対応するフラグの対を複数第３のメモリに設定するプロセッサと、
前記プロセッサが発行する鍵選択信号に基づいて鍵の初期値と対応するフラグの対を１つ前記第１のメモリに設定する第２のセレクタを更に備えた、付記１又は２記載の暗号化装置。
（付記６）
前記第２のメモリに保持された前記第１の拡張鍵～前記第Ｎの拡張鍵によりデータを暗号化するＡＥＳ（Advanced Encryption Standard）方式のエンジンを更に備えた、付記１乃至５のいずれか１項記載の暗号化装置。
（付記７）
１つの復号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用する復号化装置であって、
鍵の初期値と対応するフラグを保持する第１のメモリと、
命令と、前記第１のメモリに保持された前記フラグが示す鍵が共に復号化に関するもので、前記命令が復号化命令であり前記フラグが復号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、
第２のメモリと、
前記比較結果信号が入力されると、前記復号化命令及びトリガ信号に基づいて前記第１のメモリに保持された前記鍵を初期値として前記第２のメモリにロードする第１のセレクタと、
前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する復号化用拡張計算部を備え、
前記第１のセレクタは、前記第２のメモリへ鍵の初期値をロードする時以外は、前記復号化命令に基づいて前記復号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記暗号鍵を第Ｎの拡張鍵～第１の拡張鍵の順に拡張する、復号化装置。
（付記８）
前記比較結果信号に基づいて前記トリガ信号を生成するトリガ信号生成回路を更に備えた、付記７記載の復号化装置。
（付記９）
前記命令を発行すると共に、前記鍵の初期値と対応する前記フラグを第１のメモリに設定するプロセッサを更に備えた、付記７又は８記載の復号化装置。
（付記１０）
前記第１のメモリは、前記鍵の初期値を保持すると共に、保持する前記鍵の初期値を前記第１のセレクタへ出力するメモリと、前記フラグを保持すると共に、保持する前記フラグを前記比較回路へ出力するメモリを有する、付記７乃至９のいずれか１項記載の復号化装置。

（付記１１）
前記命令を発行すると共に、鍵の初期値と対応するフラグの対を複数第３のメモリに設定するプロセッサと、
前記プロセッサが発行する鍵選択信号に基づいて鍵の初期値と対応するフラグの対を１つ前記第１のメモリに設定する第２のセレクタを更に備えた、付記７又は８記載の復号化装置。
（付記１２）
前記第２のメモリに保持された前記第Ｎの拡張鍵～前記第１の拡張鍵によりデータを暗号化するＡＥＳ（Advanced Encryption Standard）方式のエンジンを更に備えた、付記７乃至１１のいずれか１項記載の暗号化装置。
（付記１３）
データを記憶装置に記録し、前記記憶装置からデータを再生する制御を行う制御部と、
１つの暗号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用し、前記記憶装置に記録するデータを暗号化し、前記記憶装置から再生されたデータを復号化する暗号化及び復号化装置を備え、
前記暗号化及び復号化装置は、
鍵の初期値と対応するフラグを保持する第１のメモリと、
命令と、前記第１のメモリに保持された前記フラグが示す鍵が共に暗号化に関するもので前記命令が暗号化命令であり前記フラグが暗号鍵を示すか、或いは、共に復号化に関するもので前記命令が復号化命令であり前記フラグが復号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、
第２のメモリと、
前記比較結果信号が入力されると、前記命令及びトリガ信号に基づいて前記第１のメモリに保持された前記鍵を初期値として前記第２のメモリにロードする第１のセレクタと、
前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する暗号化用拡張計算部と、
前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する復号化用拡張計算部を有し、
前記第１のセレクタは、前記第２のメモリへ鍵の初期値をロードする時以外は、前記暗号化命令に基づいて前記暗号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記暗号鍵を第１の拡張鍵～第Ｎの拡張鍵の順に拡張すると共に、前記復号化命令に基づいて前記復号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記復号鍵を第Ｎの拡張鍵～第１の拡張鍵の順に拡張する、記憶装置。
（付記１４）
前記暗号化及び復号化装置は、前記比較結果信号に基づいて前記トリガ信号を生成するトリガ信号生成回路を更に有する、付記１３記載の記憶装置。
（付記１５）
前記命令を発行すると共に、前記鍵の初期値と対応する前記フラグを第１のメモリに設定するプロセッサを更に備えた、付記１３又は１４記載の記憶装置。
（付記１６）
前記第１のメモリは、前記鍵の初期値を保持すると共に、保持する前記鍵の初期値を前記第１のセレクタへ出力するメモリと、前記フラグを保持すると共に、保持する前記フラグを前記比較回路へ出力するメモリを有する、付記１３乃至１５のいずれか１項記載の記憶装置。
（付記１７）
前記命令を発行すると共に、鍵の初期値と対応するフラグの対を複数第３のメモリに設定するプロセッサと、
前記プロセッサが発行する鍵選択信号に基づいて鍵の初期値と対応するフラグの対を１つ前記第１のメモリに設定する第２のセレクタを更に備えた、付記１３又は１４記載の記憶装置。

（付記１８）
前記第２のメモリに保持された前記第１の拡張鍵～前記第Ｎの拡張鍵によりデータを暗号化すると共に、前記第２のメモリに保持された前記第Ｎの拡張鍵～前記第１の拡張鍵によりデータを復号化するＡＥＳ（Advanced Encryption Standard）方式のエンジンを更に備えた、付記１３乃至１７のいずれか１項記載の記憶装置。
（付記１９）
前記比較回路での比較結果が不一致であると、前記第１のメモリに保存されている鍵の初期値を前記第２のメモリに設定し、前記暗号化用拡張計算部又は前記復号化用拡張計算部において鍵拡張を行って得られた前記第２のメモリ内の鍵の初期値を前記第１のメモリに設定し、前記第１のメモリに設定された鍵の初期値に対応するフラグを前記第１のメモリに設定することで前記第１のメモリの内容を更新する、付記１５又は１７記載の記憶装置。
（付記２０）
１つの暗号鍵を拡張して得られるＮ（Ｎは２以上の自然数）個の拡張鍵を順次データ処理に使用する暗号化及び復号化装置であって、
鍵の初期値と対応するフラグを保持する第１のメモリと、
命令と、前記第１のメモリに保持された前記フラグが示す鍵が共に暗号化に関するもので前記命令が暗号化命令であり前記フラグが暗号鍵を示すか、或いは、共に復号化に関するもので前記命令が復号化命令であり前記フラグが復号鍵を示すと、比較結果が一致することを示す比較結果信号を出力する比較回路と、
第２のメモリと、
前記比較結果信号が入力されると、前記命令及びトリガ信号に基づいて前記第１のメモリに保持された前記鍵を初期値として前記第２のメモリにロードする第１のセレクタと、
前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する暗号化用拡張計算部と、
前記第２のメモリに保持された鍵に基づいて順次拡張鍵を計算して前記第１のセレクタに入力する復号化用拡張計算部を備え、
前記第１のセレクタは、前記第２のメモリへ鍵の初期値をロードする時以外は、前記暗号化命令に基づいて前記暗号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記暗号鍵を第１の拡張鍵～第Ｎの拡張鍵の順に拡張すると共に、前記復号化命令に基づいて前記復号化用拡張計算部で計算された拡張鍵を前記第２のメモリにロードすることで前記復号鍵を第Ｎの拡張鍵～第１の拡張鍵の順に拡張する、暗号化及び復号化装置。
【００７７】
以上、本発明を実施例により説明したが、本発明は上記実施例に限定されるものではなく、本発明の範囲内で種々の変形及び改良が可能であることは言うまでもない。
【図面の簡単な説明】
【００７８】
【図１】ＡＥＳ方式を用いる暗号化処理を説明する図である。
【図２】ＡＥＳ方式を用いる復号化処理を説明する図である。
【図３】従来の暗号化又は復号化装置の一例を説明する図である。
【図４】本発明の一実施例における暗号化及び復号化装置を説明する図である。
【図５】記憶装置を示すブロック図である。
【図６】拡張鍵の生成シーケンスを説明するフローチャートである。
【図７】ＣＰＵの介在なしにＬセクタの処理を行う場合のシーケンスを説明するフローチャートである。
【図８】ＣＢＣモードの暗号化処理を説明する図である。
【図９】ＣＢＣモードの復号化処理を説明する図である。
【図１０】ノンスワードを暗号鍵で暗号化して初期化ベクトルとして使用する場合のＣＢＣモードの暗号化処理を説明する図である。

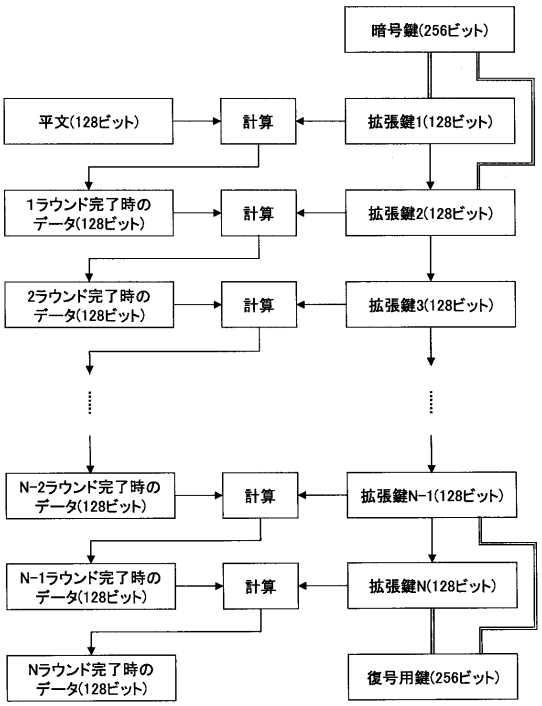
【図１１】ノンスワードを暗号鍵で暗号化して初期化ベクトルとして使用する場合のＣＢＣモードの復号化処理を説明する図である。

【符号の説明】

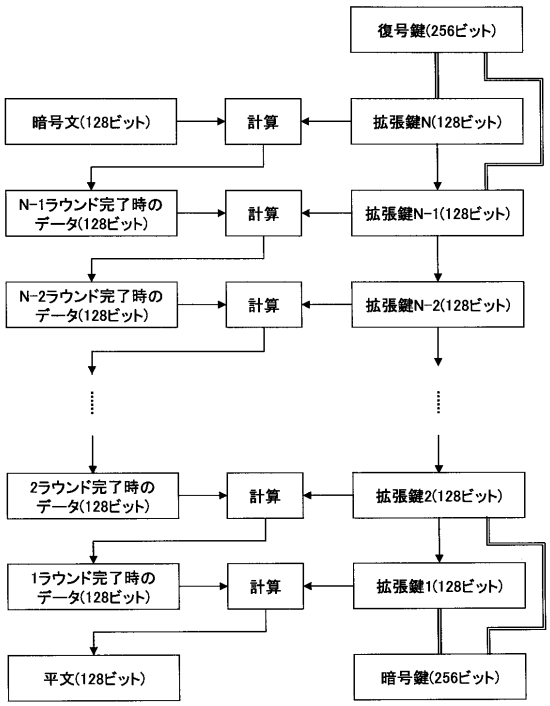
【００７９】

２０　　ＣＰＵ
２１，２２，２５，３１，３３　　メモリ
２３　　比較回路
２４，３２　　セレクタ
２６　　暗号化用拡張計算部
２７　　復号化用拡張計算部
２８　　鍵拡張計算回路
２９　　トリガ信号生成回路
３４　　鍵拡張ブロック
３５　　ＡＥＳエンジン
３６　　ヘッド
３７　　ディスク
１００　　コントロール回路
１０１　　命令保持メモリ
１０２　　処理ブロック数カウンタ
１０３　　内部命令生成回路
１０４　　鍵拡張ラウンドカウンタ

【図１】

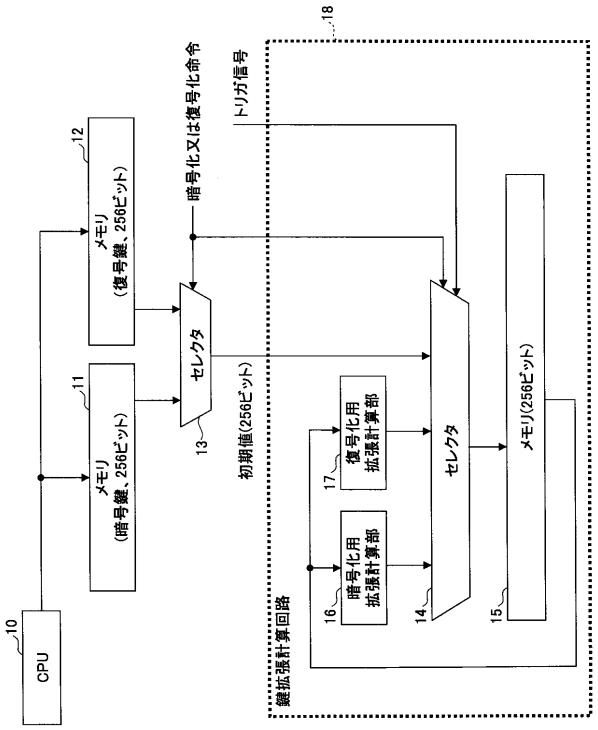


【図２】



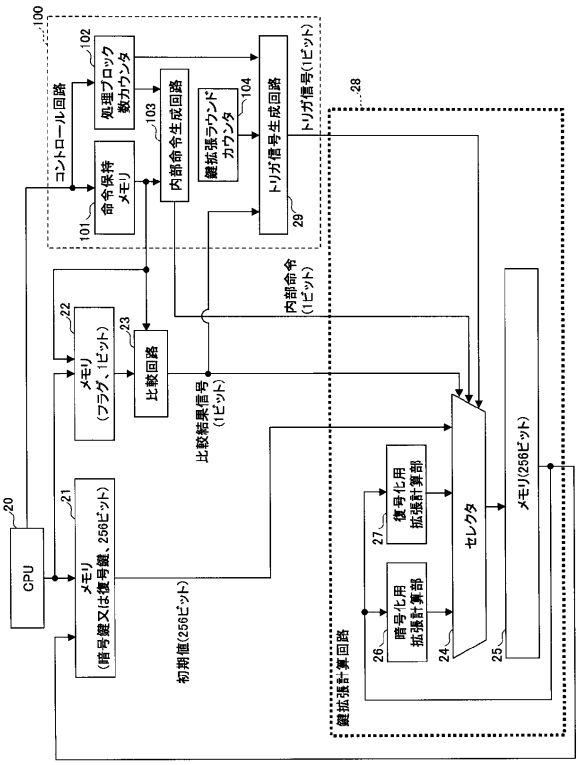
【図３】

従来の暗号化又は復号化装置の一例を説明する図



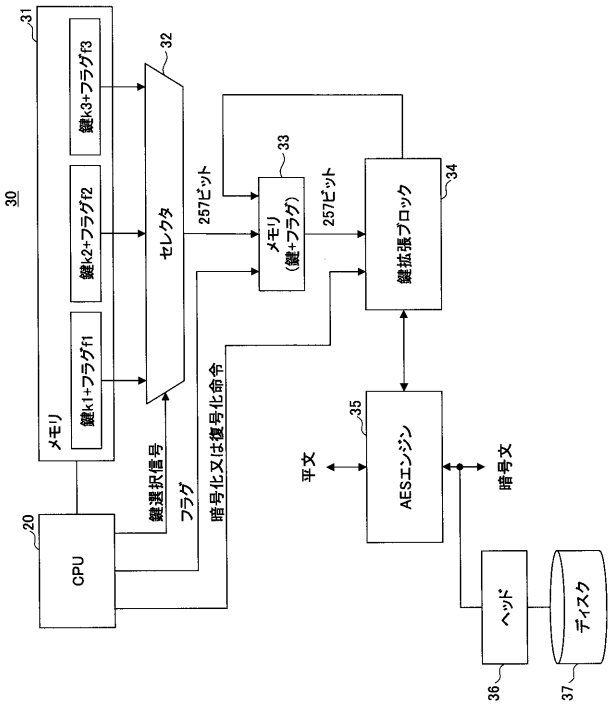
【図４】

本発明の一実施例における暗号化及び復号化装置を説明する図



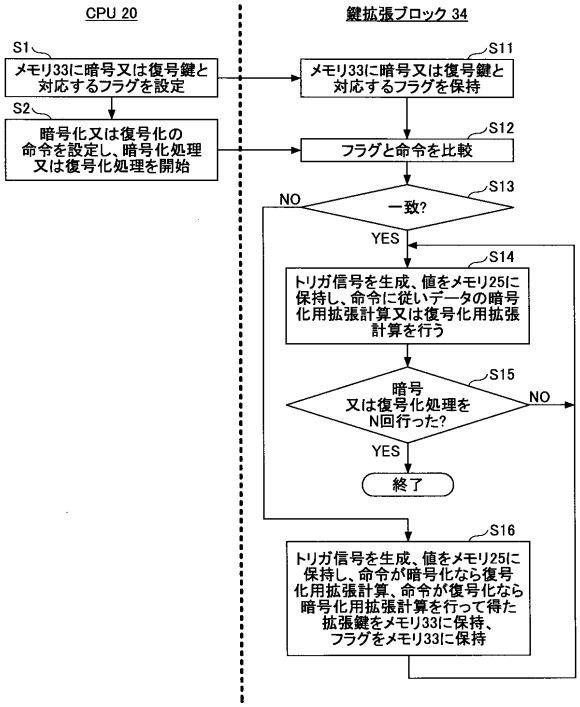
【図５】

記憶装置を示すブロック図



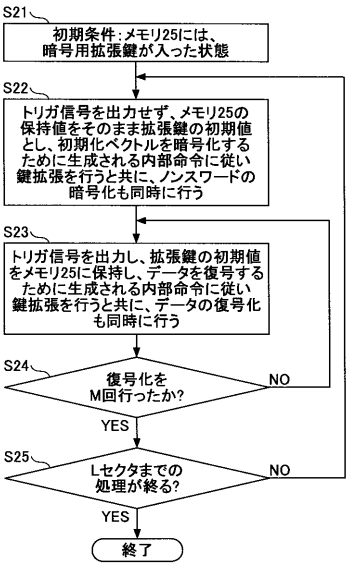
【図６】

拡張鍵の生成シーケンスを説明するフローチャート



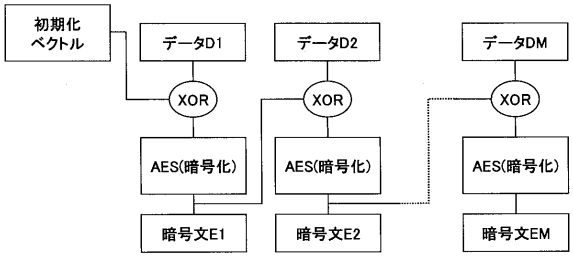
【図７】

CPUの介在なしにLセクタの処理を行う場合のシーケンスを説明するフローチャート



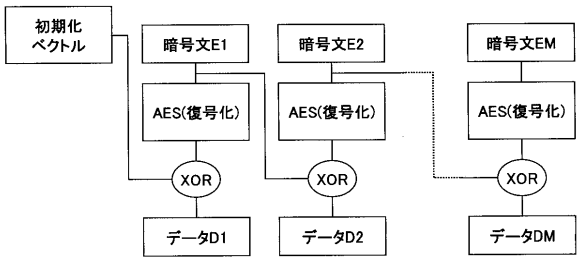
【図８】

CBCモードの暗号化処理を説明する図



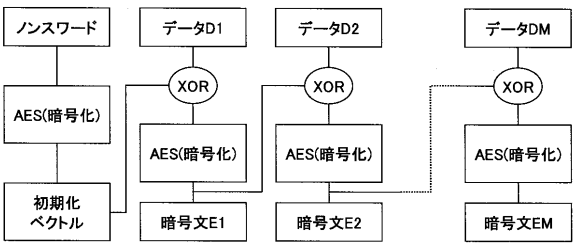
【図９】

CBCモードの復号化処理を説明する図



【図１０】

ノンスワードを暗号鍵で暗号化して初期化ベクトルとして使用する場合のCBCモードの暗号化処理を説明する図



【図１１】

ノンスワードを暗号鍵で暗号化して初期化ベクトルとして使用する場合のCBCモードの復号化処理を説明する図