

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4851077号

(P4851077)

(45) 発行日 平成24年1月11日(2012. 1. 11)

(24) 登録日 平成23年10月28日(2011. 10. 28)

(51) Int. Cl. F 1
G 0 9 C 1/00 (2006. 01) G 0 9 C 1/00 6 5 0 A
G 0 6 F 7/72 (2006. 01) G 0 6 F 7/72

請求項の数 6 (全 21 頁)

(21) 出願番号	特願2004-274103 (P2004-274103)	(73) 特許権者	000005821
(22) 出願日	平成16年9月21日(2004. 9. 21)		パナソニック株式会社
(65) 公開番号	特開2005-141198 (P2005-141198A)		大阪府門真市大字門真1006番地
(43) 公開日	平成17年6月2日(2005. 6. 2)	(74) 代理人	100109210
審査請求日	平成19年7月18日(2007. 7. 18)		弁理士 新居 広守
(31) 優先権主張番号	特願2003-353439 (P2003-353439)	(72) 発明者	横田 薫
(32) 優先日	平成15年10月14日(2003. 10. 14)		大阪府門真市大字門真1006番地 松下
(33) 優先権主張国	日本国(JP)		電器産業株式会社内
		(72) 発明者	大森 基司
			大阪府門真市大字門真1006番地 松下
			電器産業株式会社内
		(72) 発明者	山道 将人
			大阪府門真市大字門真1006番地 松下
			電器産業株式会社内

最終頁に続く

(54) 【発明の名称】 データ変換装置およびその方法

(57) 【特許請求の範囲】

【請求項 1】

有限体 $GF(2^n)$ 上の乗算を行なう有限体乗算手段と、
 複数のデータを取得する取得手段と、
 前記有限体乗算手段を利用して前記複数のデータの各々に対して、有限体 $GF(2^n)$
 (n は自然数) 上の値を係数とする多項式剰余環上における所定のべき数による、べき乗
 演算に基づく変換を行うべき乗手段と、

前記べき乗手段で変換された後の前記複数のデータに基づいて出力データを生成する出
 力データ生成手段とを備え、

前記所定のべき数は3以上であり、かつ 2^m (m は1以上の整数) 以外の値であり、

前記べき乗手段は、第1入力データと第2入力データとの有限体 $GF(2^n)$ 上の値を
 係数とする多項式剰余環上の乗算を行う有限体多項式乗算部を有し、前記複数のデータの
 各々について、当該データを前記第1入力データとし、初回の多項式剰余環上の乗算では
 当該データを前記第2入力データとし、2回目以降の多項式剰余環上の乗算では直近の多
 項式剰余環上の乗算結果を前記第2入力データとして、前記有限体多項式乗算部により前
 記第1入力データと前記第2入力データとの有限体 $GF(2^n)$ 上の値を係数とする多項
 式剰余環上の乗算を、(前記所定のべき数-1) 回繰り返した乗算結果を変換後の値とす
 ることにより、前記複数のデータの各々に対して、有限体 $GF(2^n)$ (n は自然数) 上
 の値を係数とする多項式剰余環上における所定のべき数による、べき乗演算に基づく変換
 を行い、

10

20

前記有限体多項式乗算部は、前記第1入力データを n ビットずつ分割した複数の第1分割データの各々と、前記第2入力データを n ビットずつ分割した複数の第2分割データの各々とを前記有限体乗算手段の入力とすることにより得られる有限体 $GF(2^n)$ 上の乗算結果を加算器を用いて加算し、加算結果を結合することにより、前記第1入力データと前記第2入力データとの有限体 $GF(2^n)$ 上の値を係数とする多項式剰余環上の結果を得る

ことを特徴とするデータ変換装置。

【請求項2】

前記出力データ生成手段は、

前記べき乗手段で変換された後の前記複数のデータ同士で、前記多項式剰余環上の加算を行なう加算部と、

前記加算部の加算結果に対して所定の定数との前記有限体 $GF(2^n)$ 上の乗算を行なう乗算部とを有する

ことを特徴とする請求項1に記載のデータ変換装置。

【請求項3】

有限体乗算手段、取得手段、べき乗手段および出力データ生成手段を備えるデータ変換装置が実行するデータ変換方法であって、

前記べき乗手段は、有限体多項式乗算部を有し、

前記データ変換方法は、

前記取得手段が、複数のデータを取得する取得ステップと、

前記べき乗手段が、前記複数のデータの各々に対して、有限体 $GF(2^n)$ （ n は自然数）上の値を係数とする多項式剰余環上における所定のべき数による、べき乗演算に基づく変換を行なうべき乗ステップと、

前記出力データ生成手段が、前記べき乗ステップで変換された後の前記複数のデータに基づいて出力データを生成する出力データ生成ステップとを含み、

前記所定のべき数は3以上であり、かつ 2^m （ m は1以上の整数）以外の値であり、

前記べき乗ステップでは、前記べき乗手段が、第1入力データと第2入力データとの有限体 $GF(2^n)$ 上の値を係数とする多項式剰余環上の乗算を行う前記有限体多項式乗算部を用いて、前記複数のデータの各々について、当該データを前記第1入力データとし、初回の多項式剰余環上の乗算では当該データを前記第2入力データとし、2回目以降の多項式剰余環上の乗算では直近の多項式剰余環上の乗算結果を前記第2入力データとして、前記有限体多項式乗算部により前記第1入力データと前記第2入力データとの有限体 $GF(2^n)$ 上の値を係数とする多項式剰余環上の乗算を、（前記所定のべき数-1）回繰り返した乗算結果を変換後の値とすることにより、前記複数のデータの各々に対して、有限体 $GF(2^n)$ （ n は自然数）上の値を係数とする多項式剰余環上における所定のべき数による、べき乗演算に基づく変換を行い、

前記有限体多項式乗算部は、前記第1入力データを n ビットずつ分割した複数の第1分割データの各々と、前記第2入力データを n ビットずつ分割した複数の第2分割データの各々とを、有限体 $GF(2^n)$ 上の乗算を行なう前記有限体乗算手段の入力とすることにより得られる有限体 $GF(2^n)$ 上の乗算結果を加算器を用いて加算し、加算結果を結合することにより、前記第1入力データと前記第2入力データとの有限体 $GF(2^n)$ 上の値を係数とする多項式剰余環上の結果を得る

ことを特徴とするデータ変換方法。

【請求項4】

前記出力データ生成手段は加算部と乗算部とを含み、

前記出力データ生成ステップは、

前記加算部が、前記べき乗ステップで変換された後の前記複数のデータ同士で、前記多項式剰余環上の加算を行なう加算サブステップと、

前記乗算部が、前記加算サブステップの加算結果に対して所定の定数との前記有限体 $GF(2^n)$ 上の乗算を行なう乗算サブステップとを含む

ことを特徴とする請求項 3 に記載のデータ変換方法。

【請求項 5】

有限体 $GF(2^n)$ (n は自然数) 上の乗算を行なう有限体乗算手段と、

複数のデータを取得する取得手段と、

前記有限体乗算手段を利用して前記複数のデータの各々に対して、有限体 $GF(2^n)$ (n は自然数) 上の値を係数とする多項式剰余環上における所定のべき数による、べき乗演算に基づく変換を行なうべき乗手段と、

前記べき乗手段で変換された後の前記複数のデータに基づいて出力データを生成する出力データ生成手段とを備え、

前記所定のべき数は 3 以上であり、かつ 2^m (m は 1 以上の整数) 以外の値であり、

前記べき乗手段は、第 1 入力データと第 2 入力データとの有限体 $GF(2^n)$ 上の値を係数とする多項式剰余環上の乗算を行う有限体多項式乗算部を有し、前記複数のデータの各々について、当該データを前記第 1 入力データとし、初回の多項式剰余環上の乗算では当該データを前記第 2 入力データとし、2 回目以降の多項式剰余環上の乗算では直近の多項式剰余環上の乗算結果を前記第 2 入力データとして、前記有限体多項式乗算部により前記第 1 入力データと前記第 2 入力データとの有限体 $GF(2^n)$ 上の値を係数とする多項式剰余環上の乗算を、(前記所定のべき数 - 1) 回繰り返した乗算結果を変換後の値とすることにより、前記複数のデータの各々に対して、有限体 $GF(2^n)$ (n は自然数) 上の値を係数とする多項式剰余環上における所定のべき数による、べき乗演算に基づく変換を行い、

前記有限体多項式乗算部は、前記第 1 入力データを n ビットずつ分割した複数の第 1 分割データの各々と、前記第 2 入力データを n ビットずつ分割した複数の第 2 分割データの各々とを前記有限体乗算手段の入力とすることにより得られる有限体 $GF(2^n)$ 上の乗算結果を加算器を用いて加算し、加算結果を結合することにより、前記第 1 入力データと前記第 2 入力データとの有限体 $GF(2^n)$ 上の値を係数とする多項式剰余環上の結果を得る

ことを特徴とする集積回路。

【請求項 6】

コンピュータを、請求項 1 または請求項 2 に記載のデータ変換装置として機能させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、認証方式などに使用されるデータ変換方式を実現するデータ変換装置およびその方法に関し、特に小さな実装規模で実現でき、かつ高いデータ攪乱性能有するデータ変換装置およびその方法に関する。

【背景技術】

【0002】

通信相手の正当性を確認する手段の一つであるチャレンジレスポンス認証方式などでは、認証側および被認証側において秘密の変換方式が必要となる。秘密の変換方式に望まれる要件としては、高いデータ攪乱性能 (アバランシェ性能) のみならず、その方式を低コストで機器に実装できることが望まれる。

【0003】

データ変換方式の従来例として、秘密鍵暗号方式を用いる方式がある。例えば、56 ビット鍵長 DES (Data Encryption Standard) 暗号方式 (DES 暗号方式の詳細は、非特許文献 1 参照) を用いたデータ変換方式によってチャレンジレスポンス認証方式を実現する場合には、DES 暗号方式の 56 ビット鍵を認証用鍵として認証側、被認証側の双方で秘密に保持する。また、DES 暗号方式の平文および暗号文をそれぞれデータ変換方式の入力および出力とする。これにより、DES 暗号方式を、認証用の秘密のデータ変換方式として使用することができる (認証方式の詳細は、非特許文

10

20

30

40

50

献 1 参照)。

【非特許文献 1】岡本龍明、山本博資「現代暗号」(産業図書)、1997 年

【発明の開示】

【発明が解決しようとする課題】

【0004】

しかしながら、DES 暗号方式などの秘密鍵暗号方式は、暗号回路とともに機器内で実装される他の回路との回路共有化を考慮したような方式とはなっておらず、このため、他の回路とは独立の回路として実装する必要がある。従って、従来例の秘密鍵暗号方式を用いたデータ変換方式においては、機器内に実装される他の回路とは全く独立して暗号回路を実装することになり、機器全体として回路規模が大きくなる。つまり、一般に機器を低コストで実現するためには、機器に実装される回路の総規模は極力小さくすることが要求されるので、機器に実装する暗号回路としては、他の回路と共有化できる方が望ましいが、前記従来構成では、それが実現されていないという課題を有している。

10

【0005】

本発明は、上述の課題を解決するためになされたものであり、機器内の総実装規模を削減することが可能なデータ変換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】

【0006】

本発明のデータ変換装置は、入力データを複数のデータに分割する分割手段と、前記複数のデータの各々に対して、有限体 $GF(2^n)$ (n は自然数) 上の値を係数とする多項式剰余環上における所定のべき数によるべき乗演算に基づく変換を行うべき乗手段と、前記べき乗手段で変換された後の前記複数のデータに基づいて出力データを生成する出力データ生成手段とを備え、前記所定のべき数は 3 以上であり、かつ 2^m (m は 1 以上の整数) 以外の値であることを特徴とする。

20

【0007】

この構成によると、べき乗ステップにおいて、多項式剰余環上の乗算が行なわれる。多項式剰余環上の演算を行なうと、後述するように入力データの一部でも変更されると、その影響が出力データのあらゆるビットに及ぶことになる。このため、データ攪乱性を向上させることができる。また、2 つ以上の変数の乗算を行なった場合には、いずれか 1 つの変数が 0 の場合には、その他の変数がどのような値であっても乗算結果が 0 となり、良好なデータ攪乱性能を示さないが、入力データのべき乗の場合には、このような問題が生じず、データの攪乱性を向上させることができる。さらに、べき乗ステップでは、有限体 $GF(2^n)$ (n は自然数) 上の値を係数とする多項式剰余環上における演算を行なっている。Reed-Solomon 符号や、BCH (Bose-Chaudhuri-Hocqenghem) 符号などの誤り訂正符号化回路で使用される有限体 $GF(2^n)$ 上の演算回路との回路共有が可能となる。このため、機器全体としての実装規模が削減でき、コンパクトな回路規模での機器実装が実現される。

30

【発明の効果】

【0008】

以下本発明の実施の形態について、図面を参照しながら説明する。

40

(データ変換装置を利用した認証システムの構成)

図 1 は、本発明の実施の形態に係る認証システムの構成を示すブロック図である。この認証システムは、認証装置 3 が被認証装置 4 をチャレンジャーレスポンス認証方式によって認証する。本認証システムとしては、例えば認証装置 3 を自動車のドアの開錠および施錠を制御する車載機とし、被認証装置 4 をユーザが自動車のドアの開錠および施錠のために携帯する携帯端末とする自動車のキーレスエントリシステムなどが具体的に考えられる。

【0009】

認証装置 3 は、乱数生成器 5 において 64 ビットの乱数データを生成し、これをチャレンジデータとして被認証装置 4 に送付する。被認証装置 4 は、受信したチャレンジデータに対してデータ変換装置 2 においてデータ変換処理を行ない、変換結果である 64 ビット

50

変換データをレスポンスデータとして認証装置 3 に送付する。認証装置 3 は、被認証装置 4 が上述の処理を行っている間に前記乱数データに対してデータ変換装置 1 においてデータ変換処理を行ない、変換結果である 64 ビット変換データを作成する。そして、認証装置 3 は、被認証装置 4 から受信した前記レスポンスデータと前記の変換データとをデータ比較器 6 において比較し、両者が一致する場合に限り被認証装置 4 を正当なものであるとして認証する。ここで、認証装置 3 内部のデータ変換装置 1 および被認証装置 4 内部のデータ変換装置 2 は同一の変換処理を行ない、その処理内容は認証装置 3 および被認証装置 4 の間で秘密に共有化されている。

【0010】

(データ変換装置 1、2 の構成)

データ変換装置 1、2 は同一の構成であるから、以下ではデータ変換装置 1 の内部構成のみについて説明する。

【0011】

図 2 は、データ変換装置 1 の内部構成を示した図である。データ変換装置 1 は、64 ビットの入力データに対して所定の秘密変換処理を行ない 64 ビットの出力データを生成する処理装置であり、有限体多項式 3 乗部 10 と、データ融合部 11a ~ 11d、12 および 13 と、第 1 変換部 14 と、第 2 変換部 15 と、データ分割部 16 と、データ結合部 17 とからなる。以下、64 ビットの入力データが入力されたときの内部動作について説明する。

【0012】

まず、データ分割部 16 が、前記入力データを上位から 32 ビットずつ 2 つのデータに分割する。ここでは 2 つのデータを、上位からデータ A、データ B とする。データ A はデータ融合部 11a および 11c に、データ B はデータ融合部 11b および 11d にそれぞれ入力される。データ融合部 11a および 11c は、入力された 32 ビットデータ A に対して、固定の 32 ビットデータ K1 および K3 とのビット毎の排他的論理和演算 (XOR) をそれぞれ行ない、32 ビットデータ A0 および A1 をそれぞれ出力する。また、データ融合部 11b および 11d は、入力された 32 ビットデータ B に対して、固定の 32 ビットデータ K2 および K4 とのビット毎の排他的論理和演算 (XOR) をそれぞれ行ない、32 ビットデータ B0 および B1 をそれぞれ出力する。ここで、32 ビットデータ K1 ~ K4 は、予め所定の値に定められた固定値である。

【0013】

次に、有限体多項式 3 乗部 10 は、上記 32 ビットデータ A0、B0、A1 および B1 のそれぞれに対して、後で述べる有限体 GF (2⁸) 上の値を係数とする多項式剰余環上の 3 乗算を行ない、それぞれ 32 ビットデータ (A0)³、(B0)³、(A1)³、(B1)³ を算出する。有限体多項式 3 乗部 10 の処理の詳細については後に説明する。

【0014】

次に、32 ビットデータ (A0)³ および (B0)³ は、データ融合部 12 に入力され、32 ビットデータ (A1)³ および (B1)³ は、データ融合部 13 に入力される。

【0015】

データ融合部 12 およびデータ融合部 13 は、入力されたそれぞれ 2 つの 32 ビットデータに対してビット毎の排他的論理和演算を行ない、32 ビットデータ (A0)³ (+) (B0)³ および (A1)³ (+) (B1)³ をそれぞれ出力する。ここで、「X (+) Y」は X と Y とのビット毎の排他的論理和演算 (XOR) を意味する。

【0016】

次に第 1 変換部 14 は、32 ビットデータ (A0)³ (+) (B0)³ に対して後で説明する有限体 GF (2⁸) 上の演算に基づく所定の変換を行ない、32 ビットデータ G0 を出力する。また、第 2 変換部 15 は、前記 (A1)³ (+) (B1)³ に対して後で説明する有限体 GF (2⁸) 上の演算に基づく所定の変換を行ない、32 ビットデータ G1 を出力する。

【0017】

10

20

30

40

50

以上の処理の後、データ結合部 17 は、前記 32 ビットデータ G0 を上位 32 ビット、前記 32 ビットデータ G1 を下位 32 ビットとしてデータ連結し、その結果を 64 ビットデータとして出力する。この 64 ビットデータが、データ変換装置 1 の出力データとなる。次に、有限体多項式 3 乗部 10 の内部構成と動作について説明する。

【0018】

(有限体多項式 3 乗部 10 の内部構成)

図 3 は、有限体多項式 3 乗部 10 の内部構成の一例を示した図である。有限体多項式 3 乗部 10 は、有限体 GF(2⁸) 上の値を係数とする多項式剰余環上の 3 乗算を行なう処理部であり、入力制御部 101 と、有限体多項式乗算部 100 と、出力制御部 102 とからなる。

10

【0019】

入力制御部 101 は、二つの入力データのうちどちらか一方を出力するような制御を行なう。有限体多項式乗算部 100 は、二つの入力データに対して有限体 GF(2⁸) 上の値を係数とする多項式剰余環上の乗算を行なう。出力制御部 102 は、入力データを二つの出力先のうちどちらか一方に出力するような制御を行なう。

【0020】

以下、有限体多項式 3 乗部 10 に、32 ビットの入力データ X が入力された場合の内部動作について説明する。入力データ X は、入力制御部 101 および有限体多項式乗算部 100 に入力される。入力制御部 101 は、前記入力データ X をそのまま有限体多項式乗算部 100 に入力する。有限体多項式乗算部 100 は、外部より入力された 32 ビットデータ X と入力制御部 101 より入力された 32 ビットデータ X との有限体 GF(2⁸) 上の値を係数とする多項式剰余環上の乗算(詳細は後で説明する)を行ない、乗算結果 X² を出力し、出力制御部 102 に入力する。有限体多項式乗算部 100 の処理の詳細については後で説明する。

20

【0021】

出力制御部 102 は、入力データ X² をそのまま入力制御部 101 に入力する。そして、入力制御部 101 は、入力データ X² を有限体多項式乗算部 100 に入力する。

【0022】

有限体多項式乗算部 100 は、前記入力データ X² と、入力データ X との乗算を行ない、乗算結果 X³ を出力制御部 102 に入力する。このときの乗算は既に述べた通り多項式剰余環上の乗算である。

30

【0023】

出力制御部 102 は、前記入力データ X³ を有限体多項式 3 乗部 10 の出力データとして出力する。次に、有限体多項式乗算部 100 の内部構成と動作について説明する。

【0024】

(有限体多項式乗算部 100 の内部構成)

図 4 は、有限体多項式乗算部 100 の内部構成の一例を示した図である。有限体多項式乗算部 100 は、32 ビットの第 1 入力データ X と 32 ビットの第 2 入力データ Y との有限体 GF(2⁸) 上の値を係数とする多項式剰余環上の乗算を行ない、32 ビットの出力データ D を出力する。このときの多項式剰余環の剰余多項式は $L(X) = X^4 - 1$ とし、有限体 GF(2⁸) の原始多項式 $m(x)$ は $m(x) = x^8 + x^4 + x^3 + x + 1$ とする。有限体多項式乗算部 100 の動作説明の前に、有限体 GF(2⁸) 上の演算および多項式剰余環上の演算について簡単に説明する。

40

【0025】

まず、有限体 GF(2⁸) 上の演算について説明する。有限体 GF(2⁸) 上の演算では、8 ビットデータ A の各ビットの値を上位から a₇、a₆、・・・、a₀ としたとき、7 次の多項式 $a(x) = a_7 \times x^7 + a_6 \times x^6 + \dots + a_1 \times x + a_0$ と対応づけて考える。同様に 8 ビットデータ B の各ビットを上位から b₇、b₆、・・・、b₀ とし、7 次の多項式 $b(x) = b_7 \times x^7 + b_6 \times x^6 + \dots + b_1 \times x + b_0$ と対応づける。このとき、有限体 GF(2⁸) 上における A と B の加算結果 C は、 $c(x) = a(x) + b(x)$

50

(x) を有限体 GF (2) 上で計算した結果の c (x) を上で説明した 7 次多項式と 8 ビットデータとの対応付けによって 8 ビットデータに変換した結果になる。即ち、c (x) = c₇ × x⁷ + c₆ × x⁶ + … + c₁ × x + c₀ としたとき、

$$c_7 = a_7 + b_7$$

$$c_6 = a_6 + b_6$$

$$\dots \dots \dots$$

$$c_1 = a_1 + b_1$$

$$c_0 = a_0 + b_0$$

として求められる。ここで上記の 1 ビットデータどうしの加算「+」は全て有限体 GF (2) 上で行われる。即ち、0 + 0 = 1 + 1 = 0 であり 0 + 1 = 1 + 0 = 1 として計算される。

10

【0026】

上記のことから有限体 GF (2⁸) 上の加算は、ビットごとの排他的論理和演算をしていることに他ならない。即ち、A、B に対する加算結果 C は、C = A (+) B である。

【0027】

次に、有限体 GF (2⁸) 上の乗算について説明する。8 ビットデータ A、B、C を前述のように、7 次多項式 a (x)、b (x)、c (x) とした時、8 ビットデータ A、B の乗算結果 C は、それぞれに対応する 7 次多項式 a (x)、b (x)、c (x) による以下の計算で求められる。

【0028】

20

$$c(x) = a(x) \times b(x) \mod m(x)$$

ここで、「f (x) mod g (x)」は、g (x) を法とする f (x) の剰余計算結果であり、m (x) は、前にも述べた通り有限体 GF (2⁸) 上の原始多項式 m (x) = x⁸ + x⁴ + x³ + x + 1 である。また、このときの多項式乗算、剰余算における係数の加算、乗算は有限体 GF (2) 上で行なう。有限体 GF (2) 上の加算は既に述べたとおりであり、乗算は 0 × 0 = 0 × 1 = 1 × 0 = 0、1 × 1 = 1 として計算される。

【0029】

上記の乗算の例を挙げる。A = 57 (16 進数)、B = 83 (16 進数) としたとき、a (x) = x⁶ + x⁴ + x² + x + 1、b (x) = x⁷ + x + 1 である。これより、a (x) × b (x) = x¹³ + x¹¹ + x⁹ + x⁸ + x⁶ + x⁵ + x⁴ + x³ + 1 となり、a (x) × b (x) mod m (x) = x⁷ + x⁶ + 1 となるので、A、B の乗算結果 C は 16 進数で C1 となる。

30

【0030】

次に有限体 GF (2⁸) 上の値を係数とする多項式剰余環上の演算について説明する。多項式剰余環上の演算では、32 ビットデータ A の各バイトを上位バイトから A₀、A₁、A₂、A₃ としたとき、1 バイトデータ A₀ ~ A₃ を 3 次の多項式 A (X) = A₀ + A₁ × X + A₂ × X² + A₃ × X³ と対応づけて考える。同様に 32 ビットデータ B、C の各バイトを上位バイトから B₀、B₁、B₂、B₃ および C₀、C₁、C₂、C₃ としたとき、1 バイトデータ B₀ ~ B₃ および C₀ ~ C₃ を 3 次の多項式 B (X) = B₀ + B₁ × X + B₂ × X² + B₃ × X³ および C (X) = C₀ + C₁ × X + C₂ × X² + C₃ × X³ とそれぞれ対応づけて考える。このとき、多項式剰余環上の加算は C (X) = A (X) + B (X) として求められる。このときの多項式係数の加算は先ほど説明した有限体 GF (2⁸) 上の加算である。上記のことから言い換えると、32 ビットデータ A と B との加算は、単にビット毎の排他的論理和演算を行えば良いことになる。

40

【0031】

次に、多項式剰余環上の乗算について説明する。32 ビットデータ A および B の乗算結果を 32 ビットデータ C とし、上記のように各データを 3 次の多項式 A (X)、B (X) および C (X) と対応づけたとき、多項式剰余環上の乗算は、以下のような多項式演算で示される。

【0032】

50

$$C(X) = A(X) \times B(X) \pmod{L(X)}$$

ここで、 $L(X)$ は既に説明した通り、剰余多項式 $L(X) = X^4 - 1$ であり、上記の多項式係数の加算、乗算は有限体 $GF(2^8)$ 上で行われる。このため、上式は、以下のよう計算される。

【0033】

$$\begin{aligned} C(X) = & A_0 \times B_0 \\ & + (A_0 \times B_1 + A_1 \times B_0) \times X \\ & + (A_0 \times B_2 + A_1 \times B_1 + A_2 \times B_0) \times X^2 \\ & + (A_0 \times B_3 + A_1 \times B_2 + A_2 \times B_1 + A_3 \times B_0) \times X^3 \\ & + (A_1 \times B_3 + A_3 \times B_1 + A_2 \times B_2) \times X^4 \\ & + (A_2 \times B_3 + A_3 \times B_2) \times X^5 \\ & + (A_3 \times B_3) \times X^6 \pmod{X^4 - 1} \end{aligned}$$

10

このとき、 $X^4 = 1 \pmod{X^4 - 1}$ であることから、上記の式はさらに、以下のように変形できる。

【0034】

$$\begin{aligned} C(X) = & (A_0 \times B_0 + A_3 \times B_1 + A_2 \times B_2 + A_1 \times B_3) \\ & + (A_1 \times B_0 + A_0 \times B_1 + A_3 \times B_2 + A_2 \times B_3) \times X \\ & + (A_2 \times B_0 + A_1 \times B_1 + A_0 \times B_2 + A_3 \times B_3) \times X^2 \\ & + (A_3 \times B_0 + A_2 \times B_1 + A_1 \times B_2 + A_0 \times B_3) \times X^3 \end{aligned}$$

このことより、

20

$$C_0 = A_0 \times B_0 + A_3 \times B_1 + A_2 \times B_2 + A_1 \times B_3$$

$$C_1 = A_1 \times B_0 + A_0 \times B_1 + A_3 \times B_2 + A_2 \times B_3$$

$$C_2 = A_2 \times B_0 + A_1 \times B_1 + A_0 \times B_2 + A_3 \times B_3$$

$$C_3 = A_3 \times B_0 + A_2 \times B_1 + A_1 \times B_2 + A_0 \times B_3$$

としてCは計算できることが分かる。このとき上記の加算「+」、乗算「×」は有限体 $GF(2^8)$ 上で行われる。

【0035】

以上で有限体 $GF(2^8)$ 上の演算および多項式剰余環上の演算について説明が終わったので、有限体多項式乗算部100の動作説明を行なう。

【0036】

30

有限体多項式乗算部100は、二つの入力データに対して有限体 $GF(2^8)$ 上の値を係数とする多項式剰余環上の乗算を行なう処理部であり、有限体乗算部110と、データ分割部111～112と、データ加算部113～115と、データ結合部116と、演算制御部117とからなる。

【0037】

有限体乗算部110は、有限体 $GF(2^8)$ 上の乗算を行なう。データ分割部111～112の各々は、32ビットの入力データを4個の8ビットのデータに分割する。データ加算部113～115の各々は、2つの入力データに対して有限体 $GF(2^8)$ 上の加算を行なう。データ結合部116は、4個の8ビットデータを結合して32ビットデータとして出力する。演算制御部117は、データ分割部111およびデータ分割部112から有限体乗算部110へ入力される被乗数および乗数の入力制御および有限体乗算部110から出力されるデータの出力先制御を行なう。以下、有限体多項式乗算部100の動作について説明する。

40

【0038】

データ分割部111は、32ビットの第1入力データを上位から8ビットずつ4つのデータに分割する。ここでは、4つのデータを上位からデータX0、X1、X2およびX3とする。データ分割部112も同様に、32ビットの第2入力データを上位から8ビットずつ4つのデータに分割する。ここでは、4つのデータを上位からデータY0、Y1、Y2およびY3とする。以下、有限体乗算部110およびデータ加算部113～115は、演算制御部117によりその入出力データが適宜制御されて、次式(1)～(4)に従い

50

、8ビットデータD0、D1、D2およびD3の値を計算する。

【0039】

$$D0 = X0 \times Y0 + X3 \times Y1 + X2 \times Y2 + X1 \times Y3 \quad \dots (1)$$

$$D1 = X1 \times Y0 + X0 \times Y1 + X3 \times Y2 + X2 \times Y3 \quad \dots (2)$$

$$D2 = X2 \times Y0 + X1 \times Y1 + X0 \times Y2 + X3 \times Y3 \quad \dots (3)$$

$$D3 = X3 \times Y0 + X2 \times Y1 + X1 \times Y2 + X0 \times Y3 \quad \dots (4)$$

但し、上記の乗算「×」、加算「+」は全て有限体GF(2⁸)上で行われる。上記の式がデータXおよびデータYの積を示している理由については既に説明した通りである。

【0040】

上記4つの式のうち、データD0の計算を行なう際の有限多項式乗算部100の動作についてのみ説明する。データD1～D3については、これと同様の動作によって計算される。

【0041】

演算制御部117は、有限体乗算部110への入力として、8ビットデータX0～X3のうちからデータX0を、8ビットデータY0～Y3のうちからデータY0をそれぞれ選択する。有限体乗算部110は、データX0およびデータY0に対して有限体GF(2⁸)上の乗算を行ない、乗算結果をデータZ0として出力する。即ち、

$$Z0 = X0 \times Y0$$

を計算する。

【0042】

次に同様な動作により、有限体乗算部110は、データX3とY1に対して有限体GF(2⁸)上の乗算を行ない、データZ1として出力する。即ち、

$$Z1 = X3 \times Y1$$

を計算する。同様にして

$$Z2 = X2 \times Y2$$

$$Z3 = X1 \times Y3$$

を計算する。

【0043】

上記Z0～Z3の出力が完了後、データ加算部113～115は、上記データZ0～Z3に対して、有限体GF(2⁸)上の加算を行なう。即ち、

$$D0 = Z0 + Z1 + Z2 + Z3$$

を計算する。但し有限体GF(2⁸)上の加算はビット毎の排他的論理和演算に他ならない。このため、上記の計算はデータZ0～Z3に対してビット毎の排他的論理和演算を行なうことになる。同様にして、データ加算部113～115は、

$$D1 = X1 \times Y0 + X0 \times Y1 + X3 \times Y2 + X2 \times Y3$$

$$D2 = X2 \times Y0 + X1 \times Y1 + X0 \times Y2 + X3 \times Y3$$

$$D3 = X3 \times Y0 + X2 \times Y1 + X1 \times Y2 + X0 \times Y3$$

を計算する。データ結合部116は、上位からデータD0、D1、D2およびD3を連結して32ビットデータDを有限体多項式乗算部100の出力データとして出力する。

【0044】

次に、第1変換部14の内部構成と動作について説明する。

(第1変換部14の内部構成)

図5は、第1変換部14の内部構成の一例を示した図である。

【0045】

第1変換部14は、32ビットの入力データXに対して、有限体GF(2⁸)上の演算を用いた所定の変換を行ない、32ビットの出力データYを出力する処理部であり、データ分割部20と、データ結合部21と、定数記憶部22と、有限体乗算部210とからなる。データ分割部20は、32ビット入力データを4つの8ビットデータに分割する。データ結合部21は、4つの8ビットデータを結合して32ビットデータとして出力する。定数記憶部22は、4つの8ビット定数C1～C4を記憶する。有限体乗算部210は、

2つの8ビット入力データに対して、有限体GF(2⁸)上の乗算を行って8ビットの出力データを出力する。以下、第1変換部14の動作について説明する。

【0046】

データ分割部20は、32ビットの入力データXを上位から8ビットずつ分割する。ここでは、分割後の8ビットデータを上位からデータX0、X1、X2およびX3とする。有限体乗算部210は、データX0と、定数記憶部22に記憶されている8ビット定数C1との有限体GF(2⁸)上の乗算を行ない、出力データY0として出力する。同様に、有限体乗算部210は、データX1と定数C2との乗算を行ないデータY1として出力し、データX2と定数C3との乗算を行ないデータY2として出力し、データX3と定数C4との乗算を行ないデータY3として出力する。上記一連の動作により

$$Y0 = C1 \times X0$$

$$Y1 = C2 \times X1$$

$$Y2 = C3 \times X2$$

$$Y3 = C4 \times X3$$

が計算される。但し、上記の乗算「×」は全て有限体GF(2⁸)上で行われる。

【0047】

上記処理の後、データ結合部21は、上位からデータY0、Y1、Y2およびY3を結合して32ビットデータYを第1変換部14の出力データとして出力する。

【0048】

(第2変換部15の内部構成)

第2変換部15の内部構成は図6に示す通り、第1変換部14の構成において、定数記憶部32に記憶される定数がC1、C2、C3およびC4からC5、C6、C7およびC8にそれぞれに変わるだけであり、その他の内部構成および動作は第1変換部14と同一である。このため、その詳細な説明はここでは繰り返さない。

【0049】

次に、図7を参照して、有限体乗算部110、210および310の内部構成と動作とについて説明する。

【0050】

(有限体乗算部110、210および310の内部構成)

有限体乗算部110、210および310は、いずれも同一の内部構成を有し、同一の動作を行なう。このため、ここでは、有限体乗算部110の動作についてのみ説明する。有限体GF(2⁸)上の乗算方法については、既に述べた通りであるが、ここでは、その計算をコンパクトな回路で実現した場合の構成を説明する。

【0051】

有限体乗算部110は、8ビットの第1入力データXと第2入力データYに対して、有限体GF(2⁸)上の乗算を行ない、8ビットの出力データZを出力する処理部であり、第1入力制御部411と、第2入力制御部414と、出力制御部412と、有限体2倍算部410と、データ融合部413と、データ分割部415とからなる。

【0052】

第1入力制御部411および第2入力制御部414の各々は、2つの入力データのうちのどちらか一方を選択して、出力するよう制御を行なう。有限体2倍算部410は、入力データに対して有限体GF(2⁸)上の2倍算を行なう。データ融合部413は、2つの入力データを融合させる。データ分割部415は、入力データを複数のデータに分割する。以下、有限体乗算部110の動作について説明する。

【0053】

まず、データ分割部415は、8ビットの第2入力データYを上位から1ビットずつに分割して、それぞれ値Y7、Y6、・・・、Y0とする。次に、i=7、6、5、4、3、2、1、0の順に以下の(1)から(5)の処理を繰り返し実行する。

【0054】

(1) 第1入力制御部411は、i=7のときは8ビットの初期値=0を、i≠7のと

10

20

30

40

50

きは出力制御部 4 1 2 から出力される 8 ビットデータを有限体 2 倍算部 4 1 0 に入力する。

【0055】

(2) 有限体 2 倍算部 4 1 0 は、第 1 入力制御部 4 1 1 から入力される 8 ビットデータに対して有限体 GF (2⁸) 上の 2 倍算を行ない、結果の 8 ビットデータをデータ融合部 4 1 3 に入力する。

【0056】

(3) 第 2 入力制御部 4 1 4 は、Y_i (i = 7、6、・・・、0) が 0 の場合は 8 ビット定数 0 を、それ以外の場合は第 1 入力データ X をデータ融合部 4 1 3 に入力する。

【0057】

(4) データ融合部 4 1 3 は、有限体 2 倍算部 4 1 0 から入力される 8 ビットデータと、第 2 入力制御部 4 1 4 から入力される 8 ビットデータとに対してビット毎の排他的論理和演算を行ない、演算結果の 8 ビットデータを出力制御部 4 1 2 に入力する。

【0058】

(5) i ≠ 0 の場合は、出力制御部 4 1 2 は、データ融合部 4 1 3 から入力される 8 ビットデータを第 1 入力制御部 4 1 1 に入力する。その後、i の値が 1 だけ減らされ、(1) から処理が再度開始される。i = 0 の場合は、出力制御部 4 1 2 は、データ融合部 4 1 3 から入力される 8 ビットデータを有限体乗算部 1 1 0 の出力データとして出力する。その後、一連の処理が終了する。

【0059】

上記の処理によって第 1 入力データ X と第 2 入力データ Y との乗算結果が計算できる理由について簡単に説明する。

【0060】

第 2 入力データは、各ビットの値 Y₇、Y₆、・・・、Y₀ を用いると

$$Y = Y_7 \times 2^7 + Y_6 \times 2^6 + \dots + Y_i \times 2^i + \dots + Y_0$$

と表せるので、

$$\begin{aligned} X \times Y &= X \times (Y_7 \times 2^7 + Y_6 \times 2^6 + \dots + Y_i \times 2^i + \dots + Y_0) \\ &= (\dots ((((((0 + X \times Y_7) \times 2 + X \times Y_6) \times 2 + X \times Y_5) \times 2 + \\ &X \times Y_4) \times 2 + X \times Y_3) \dots) \times 2 + X \times Y_0 \end{aligned}$$

と表すことができる。この計算式を元にした処理を上記で行っている。

【0061】

次に、有限体 2 倍算部 4 1 0 の内部構成および動作について説明する。

(有限体 2 倍算部 4 1 0 の内部構成)

図 8 は、有限体 2 倍算部 4 1 0 の内部構成を示した図である。

【0062】

有限体 2 倍算部 4 1 0 は、入力される 8 ビットデータ X に対して有限体 GF (2⁸) 上の 2 倍算を行ない、その結果の 8 ビットデータ Y を出力する処理部であり、データ分割部 5 1 1 と、データ結合部 5 1 2 と、データ融合部 5 1 3 ~ 5 1 5 とからなる。

【0063】

データ分割部 5 1 1 は、入力データを 1 ビットずつのデータに分割する。データ結合部 5 1 2 は、複数個の入力データを結合して 1 つのデータとして出力する。データ融合部 5 1 3 ~ 5 1 5 の各々は、2 つの入力データを融合する。以下、有限体 2 倍算部 4 1 0 の動作について説明する。

【0064】

まず、データ分割部 5 1 1 は、8 ビット入力データ X を上位から 1 ビットずつ分割して、上位からデータ X₇、X₆、・・・、X₀ として出力する。次に、データ融合部 5 1 3 は、データ X₇ とデータ X₃ との排他的論理和演算を行ないデータ Y₄ として出力する。データ融合部 5 1 4 は、データ X₇ とデータ X₂ との排他的論理和演算を行ないデータ Y₃ として出力する。データ融合部 5 1 5 は、データ X₇ とデータ X₀ との排他的論理和演算を行ないデータ Y₁ として出力する。また、データ X₆、X₅、X₄、X₁ および X₇

10

20

30

40

50

はそれぞれデータ Y 7、Y 6、Y 5、Y 2 および Y 0 とされる。データ結合部 5 1 2 は、データ Y 7、Y 6、Y 5、・・・、Y 0 を上位からこの順序で結合した 8 ビットデータを、有限体 2 倍算部 4 1 0 の出力データとして出力する。

【0065】

有限体 2 倍算部 4 1 0 は、8 ビット入力データ X の各ビット X 7、X 6、・・・、X 0 に対して、

$$Y 7 = X 6$$

$$Y 6 = X 5$$

$$Y 5 = X 4$$

$$Y 4 = X 3 (+) X 7$$

$$Y 3 = X 2 (+) X 7$$

$$Y 2 = X 1$$

$$Y 1 = X 0 (+) X 7$$

$$Y 0 = X 7$$

として、8 ビット出力データ Y の各ビット Y 7、Y 6、・・・、Y 0 の値を算出している。このとき出力データ Y は、入力データ X に対して有限体 GF (2⁸) 上の 2 倍算を行った結果になっている。このことを簡単に説明する。

【0066】

入力データ X は、係数が有限体 GF (2) の値である以下のような α の多項式によって表せる。

【0067】

$$X 7 \times \alpha^7 + X 6 \times \alpha^6 + \dots + X 1 \times \alpha + X 0$$

このとき、有限体 GF (2⁸) 上で 2 倍算を行なうということは、上の多項式に α を掛け算することであるから、

$$X 7 \times \alpha^8 + X 6 \times \alpha^7 + \dots + X 1 \times \alpha^2 + X 0 \times \alpha$$

となる。ここで、原始多項式が $x^8 + x^4 + x^3 + x + 1$ であることから、 $\alpha^8 = \alpha^4 + \alpha^3 + \alpha + 1$ が成り立つ。このため、上記の多項式は、

$$X 6 \times \alpha^7 + X 5 \times \alpha^6 + X 4 \times \alpha^5 + (X 3 + X 7) \times \alpha^4 + (X 2 + X 7) \times \alpha^3 + X 1 \times \alpha + (X 0 + X 7)$$

と書き直される。これが

$$Y 7 \times \alpha^7 + Y 5 \times \alpha^6 + \dots + Y 1 \times \alpha + Y 0$$

に対応することから、上述の有限体 2 倍算部 4 1 0 のような処理を行なう理由が説明できる。

【0068】

データ変換装置 1 および 2 は、64 ビットの入力データ X に対して以下の処理を行っていることになる。

【0069】

(1) 入力データ X を上位 32 ビット、下位 32 ビットに分割してデータ X 0 および X 1 とする。

【0070】

(2) $T 0 = (X 0 + K 1)^3 + (X 1 + K 2)^3$ および $T 1 = (X 0 + K 3)^3 + (X 1 + K 4)^3$ を計算する。但し、ここでの加算、乗算は全て有限体 GF (2⁸) 上の値を係数とする多項式剰余環上で行なう。

【0071】

(3) 32 ビットのデータ T 0 を上位から 8 ビットずつに分割して、データ a 0、a 1、a 2 および a 3 とし、32 ビットのデータ T 1 を上位から 8 ビットずつに分割して、データ b 0、b 1、b 2 および b 3 とする。

【0072】

(4) $G 0 = C 1 \times a 0 \parallel C 2 \times a 1 \parallel C 3 \times a 2 \parallel C 4 \times a 3$ および $G 1 = C 5 \times b 0 \parallel C 6 \times b 1 \parallel C 7 \times b 2 \parallel C 8 \times b 3$ を計算し、G 0 ∥ G 1 を出力データとして

10

20

30

40

50

出力する。但し、「 $\parallel$ 」はデータ連結を表し、上記の乗算は全て有限体 $GF(2^8)$ 上で行なわれる。

【0073】

有限体 $GF(2^8)$ 上の値を係数とする多項式剰余環上の乗算は式(1)～式(4)から分かるように、入力データの一部でも変更されると、その変更の影響が出力データの全てに及ぶことになる。例えば、式(1)～式(4)においてデータ X_0 の値が変更されるものとする。データ X_0 は式(1)～式(4)の全ての計算に使用されている。このため、出力データ $D_0 \sim D_3$ の全てに影響が及ぶことになる。他の値($X_1 \sim X_3$ 、 $Y_0 \sim Y_3$)についても同様のことが言える。このことにより、本実施の形態では、変換処理に上記の乗算を用いることで高いデータ攪乱性能を実現することができる。次に、本実施の形態では、二つ以上の変数の乗算(例えば、 $X \times Y$ 、 $X \times Y \times Z$)ではなく、べき乗算を用いているが、これは、二つ以上の変数の乗算の場合、それらの変数のうちいずれか一つでも0の場合、その他の変数がどのような値であっても乗算結果は常に0となり、演算結果の値が0になる入力変数値の組み合わせが多数存在することになり、良好なデータ攪乱性能を示さないからである。一方、べき乗算を用いた場合には、入力変数値が0の時にしか演算結果が0とならないので、上記のようなデータ攪乱性能を低下させる問題がなくなり、高いデータ攪乱性能が保証される。

【0074】

また、本実施の形態では、3乗のべき乗算を用いているが、これは以下の理由による。まず2乗算を用いて、 $Y = X^2$ のような変換を考えたとき、入力値 α に対する出力値は、 α^2 である。次に、入力値 α に差分 Δ が加わったときの出力値は、 $(\alpha + \Delta)^2 = \alpha^2 + \alpha \times \Delta + \Delta \times \alpha + \Delta^2$ であり、 $\alpha \times \Delta = \Delta \times \alpha$ かつ $\alpha \times \Delta + \alpha \times \Delta = 0$ (いずれも、有限体 $GF(2^8)$ 上の計算方法から明らか)であることから、 $(\alpha + \Delta)^2 = \alpha^2 + \Delta^2$ となる。従って、差分 Δ を入力値に加えることによる出力値の変化分は、 Δ^2 である。即ち、出力値の変化分は、入力値 α によらず一定の出力差分 Δ^2 となってしまう、データ攪乱という観点からは好ましい性質とはいえない。これにより、少なくとも3乗以上のべき乗算を用いる必要があるが、べき数が大きいほどべき乗処理が多くなり変換処理負荷が高くなるため、本実施の形態では3乗のべき乗算を用いている。このとき、

$$\begin{aligned} (\alpha + \Delta)^3 &= (\alpha + \Delta) \times (\alpha + \Delta) \times (\alpha + \Delta) \\ &= (\alpha^2 + \alpha \times \Delta + \Delta \times \alpha + \Delta^2) \times (\alpha + \Delta) \\ &= (\alpha^2 + \Delta^2) \times (\alpha + \Delta) \\ &= \alpha^3 + \Delta \times \alpha^2 + \Delta^2 \times \alpha + \Delta^3 \end{aligned}$$

となるので、3乗のべき乗算の場合には、2乗のべき乗算のように出力差分が入力値 α によらず一定になることはない。なお、べき乗算のべき数 N を $N = 2^k$ (k は1以上の整数)とした場合には、べき乗算 X^N をデータ変換処理として入力値 α に入力差分 Δ を与えたときの出力値 $(\alpha + \Delta)^{2^k} = \alpha^{2^k} + \Delta^{2^k}$ となり、入力値によらず一定になることが以下のようにして分かる。ここで、「 $X^{\wedge} \alpha$ 」は X の α 乗を意味する。

【0075】

$k = 1$ の場合、べき数 = 2 であるから、上記説明から出力値は $(\alpha + \Delta)^2 = \alpha^2 + \Delta^2$ である。次に $k = m$ 、即ちべき数 = 2^m のときに $(\alpha + \Delta)^{2^m} = \alpha^{2^m} + \Delta^{2^m}$ が成り立つとしたとき、 $(\alpha + \Delta)^{2^{m+1}} = \{(\alpha + \Delta)^{2^m}\}^2 = \{\alpha^{2^m} + \Delta^{2^m}\}^2 = \alpha^{2^{m+1}} + \Delta^{2^{m+1}}$ が成り立つので $k = m + 1$ の時にも上記のことは成り立つ。従って、数学的帰納法より1以上の任意の整数 k に対して $(\alpha + \Delta)^{2^k} = \alpha^{2^k} + \Delta^{2^k}$ 、となることが示される。従って、べき数としては 2^k (k は1以上の整数)以外の値であれば良いことが分かる。即ち、本実施の形態ではべき数として3を用いているが、これは、データ変換装置での処理時間に多少の時間がかかっても良いのであれば、 2^k (k は1以上の整数)以外である3以上の値であれば何でも良い。

【0076】

また、本実施の形態では、 $(X+K)^3$ のようにべき乗算の前に、定数Kによる加算（排他的論理和演算）を行っているが、この定数Kを変更することによって、データ変換装置の変換処理に多くのバリエーションを与えることができる。例えば、この定数Kを認証する相手ごとに使い分けることで、認証相手ごとに認証に用いる変換処理の使い分けができることになる。

【0077】

このとき、 $(X+K)^3$ の値が0になるのは、 $X=K$ の時だけであるから、先に述べたようなべき乗演算を用いることにより、高いデータ攪乱性能が保障できるというメリットは損なわれない。

【0078】

また、本実施の形態では、4つのデータA0～A3に対して同一のべき数3によるべき乗算を行っているが、これは同一のべき数である必要はなく、それぞれ異なるべき数であってもよい。

【0079】

さらに、本実施の形態のデータ変換装置では、データ攪乱のコアとなる処理に有限体GF(2⁸)上の演算処理を用いている。このため、Reed-Solomon符号や、BCH(Bose-Chaudhuri-Hocqenghem)符号などの誤り訂正符号化回路で使用される有限体GF(2⁸)上の演算回路との回路共有が可能となる。このため、機器全体としての実装規模が削減でき、コンパクトな回路規模での機器実装が実現される。

【0080】

なお、本実施の形態の各データサイズはあくまでも一例であって、ここで述べたデータサイズ以外であってもかまわない。また、本実施の形態の原始多項式、剰余多項式もあくまでも一例であって、これに限られるものではない。

【0081】

また、本実施の形態ではデータ変換装置を認証システムに用いた場合について述べているが、秘密のデータ変換を使用するようなものであれば何でも、本実施の形態のデータ変換装置が使用できる。データ変換装置の他の応用例としては、例えば図9に示すようなコンテンツ配信システムへの適用が考えられる。コンテンツ配信システムは、コンテンツを暗号化された状態で放送またはネットワークを介して配信するコンテンツ配信機器7と、前記配信される暗号化コンテンツを受信して復号化して再生する再生機器8とからなる。コンテンツ配信機器7は、データ変換装置70において鍵シードデータ（コンテンツ鍵を生成するためのシードとなるデータ）にデータ変換処理を行ない、コンテンツ鍵を生成する。そして、コンテンツ暗号化装置71において、平文のコンテンツデータに対して暗号化を行ない暗号化コンテンツデータを生成する。上記の処理の後、コンテンツ配信機器7は、鍵シードデータと暗号化コンテンツデータとを再生機器8に送付する。前記データを受信した再生機器8は、まずデータ変換装置80において、鍵シードデータに対して変換処理を行ない、コンテンツ鍵を生成する。そして、コンテンツ復号化装置81において、暗号化コンテンツデータを復号化し平文のコンテンツを得る。コンテンツの再生を認められた再生機器のみがコンテンツ配信機器が有するものと同じデータ変換装置を搭載することによって、不正規再生機器による配信コンテンツの再生を阻止することができる。

【0082】

また、本実施の形態ではデータ融合部11a～11dにおいてデータ融合を行なう固定値K1～K4を予め定められた固定値としているが、これらの固定値をデータ変換装置外部から入力するようにして、使用者が自由に設定できるようにしてもよい。さらに、定数記憶部22に記憶される定数C1～C4も予め設定された固定値としているが、これらも、データ変換装置外部から入力するようにして、使用者が自由に設定できるようにしてもよい。

【0083】

また、本実施の形態では、有限体として有限体GF(2⁸)を用いたが、それ以外の有限体であっても良く、例えば、有限体GF(2ⁿ)（nは自然数）であってもよい。

【0084】

なお、ブロック図（図2など）の各機能ブロックは典型的には集積回路であるLSIとして実現される。これらは個別に1チップ化されても良いし、一部又は全てを含むように1チップ化されても良い。

【0085】

図10は、図2に示したデータ変換装置と同様の機能を有するデータ変換装置を含む誤り訂正・データ変換装置のLSIの外観図を示している。図11は、誤り訂正・データ変換装置のLSIの構成を示す機能ブロック図である。

【0086】

図11に示すように、誤り訂正・データ変換装置600は、データに対して誤り訂正符号化を行なった後、データ変換を行なう装置であり、リードソロモン誤り訂正符号化部601と、データ変換装置604とを備えている。

10

【0087】

リードソロモン誤り訂正符号化部601は、入力されたデータに対してReed-Solomon誤り訂正符号化を行ない、符号化されたデータを出力する処理部であり、データ受信部602と、符号化部603とを備えている。データ受信部602は、外部より入力されるデータを受信する処理部である。符号化部603は、データ受信部602で受信されたデータに対し、有限体GF(2ⁿ)上の乗算を行なうことにより、Reed-Solomon誤り訂正符号化を行なう処理部である。符号化部603は、有限体GF(2ⁿ)上の乗算を行なう有限体乗算部110を備えている。有限体乗算部110の構成は上述したとおりである。

20

【0088】

データ変換装置604は、上述のデータ変換装置1(2)と同様の構成を有するが、有限体多項式乗算部100の代わりに有限体多項式乗算部605を用い、第1変換部14の代わりに第1変換部606を用い、第2変換部15の代わりに第2変換部607を用いている。有限体多項式乗算部605は、符号化部603に設けられた有限体乗算部110を用いて有限体GF(2ⁿ)上の乗算を行なう点が有限体多項式乗算部100と異なり、その他の構成は有限体多項式乗算部100と同様である。第1変換部606は、符号化部603に設けられた有限体乗算部110を用いて有限体GF(2ⁿ)上の乗算を行なう点が第1変換部14と異なり、その他の構成は第1変換部14と同様である。第2変換部607は、符号化部603に設けられた有限体乗算部110を用いて有限体GF(2ⁿ)上の乗算を行なう点が第2変換部15と異なり、その他の構成は第2変換部15と同様である。

30

【0089】

このように、リードソロモン誤り訂正符号化部601とデータ変換装置604とで有限体乗算部110を共有化することができる。このため、LSIの回路規模を削減することができる。

【0090】

ここでは、LSIにより誤り訂正・データ変換装置600を実現したが、集積度の違いにより、IC、システムLSI、スーパーLSI、ウルトラLSIと呼称されることもある。

40

【0091】

また、集積回路化の手法はLSIに限るものではなく、専用回路又は汎用プロセサで実現してもよい。LSI製造後に、プログラムすることが可能なFPGA(Field Programmable Gate Array)や、LSI内部の回路セルの接続や設定を再構成可能なリコンフィギュラブル・プロセッサを利用しても良い。

【0092】

さらには、半導体技術の進歩又は派生する別技術によりLSIに置き換わる集積回路化の技術が登場すれば、当然、その技術を用いて機能ブロックの集積化を行ってもよい。別技術としてバイオ技術等が可能性としてありえる。

50

【産業上の利用可能性】

【0093】

本発明にかかるデータ変換装置は、誤り訂正符号化回路とデータ変換回路の共有化を図ることにより、データ変換装置を含む回路全体の規模削減が可能であるという効果を有するので、例えば通信路を介して相手認証をする機能を有する機器などに有用である。また、この例に限られず、何らかのデータ変換回路の実装が必要な機器であればどのような機器であっても適用が可能である。

【図面の簡単な説明】

【0094】

【図1】本発明の実施の形態に係る認証システムの構成を示すブロック図である。

10

【図2】本発明の実施の形態に係るデータ変換装置の構成の一例を示すブロック図である。

【図3】本発明の実施の形態に係る有限体多項式3乗部の構成の一例を示すブロック図である。

【図4】本発明の実施の形態に係る有限体多項式乗算部の構成の一例を示すブロック図である。

【図5】本発明の実施の形態に係る第1変換部の構成の一例を示すブロック図である。

【図6】本発明の実施の形態に係る第2変換部の構成の一例を示すブロック図である。

【図7】本発明の実施の形態に係る有限体乗算部の構成の一例を示すブロック図である。

【図8】本発明の実施の形態に係る有限体2倍算部の構成の一例を示すブロック図である。

20

【図9】本発明の変換装置をコンテンツ配信システムに適用した場合のシステム構成の一例を示すブロック図である。

【図10】誤り訂正・データ変換装置のLSIの外観図である。

【図11】誤り訂正・データ変換装置の構成の一例を示すブロック図である。

【符号の説明】

【0095】

1, 2, 70, 80 データ変換装置

3 認証装置

4 被認証装置

30

5 乱数生成器

6 データ比較器

7 コンテンツ配信機器

8 再生機器

10 有限体多項式3乗部

11a, 11b, 11c, 11d, 12, 13, 413, 513, 514, 515

データ融合部

14 第1変換部

15 第2変換部

16, 20, 30, 111, 112, 415, 511 データ分割部

40

17, 21, 31, 116, 512 データ結合部

22, 32 定数記憶部

71 コンテンツ暗号化装置

81 コンテンツ復号化装置

100 有限体多項式乗算部

101 入力制御部

102, 412 出力制御部

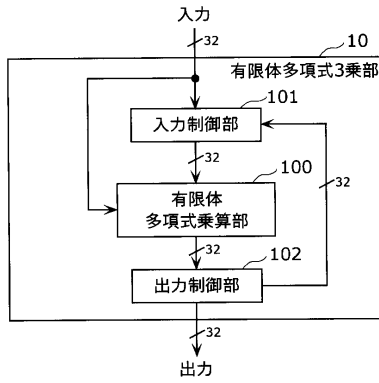
110, 210, 310 有限体乗算部

113, 114, 115 データ加算部

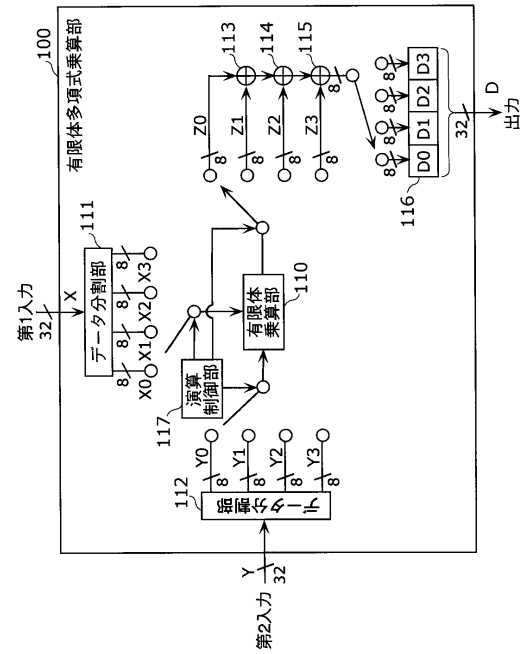
410 有限体2倍算部

50

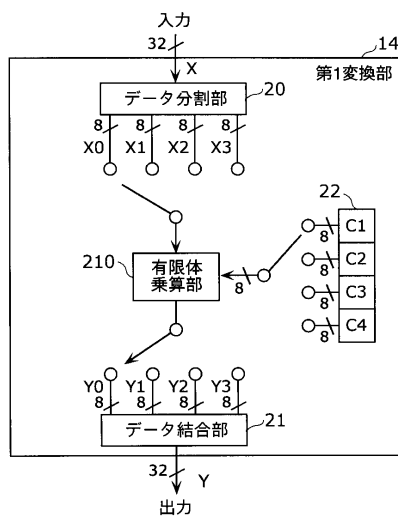
【図 3】



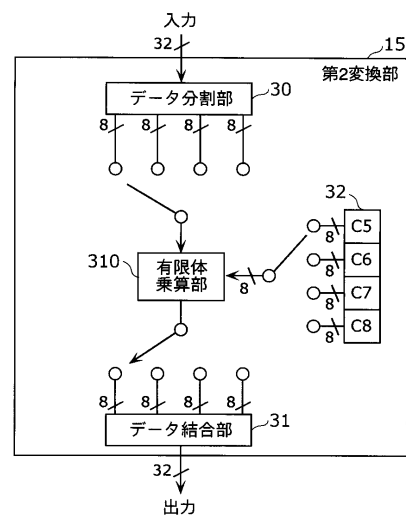
【図 4】



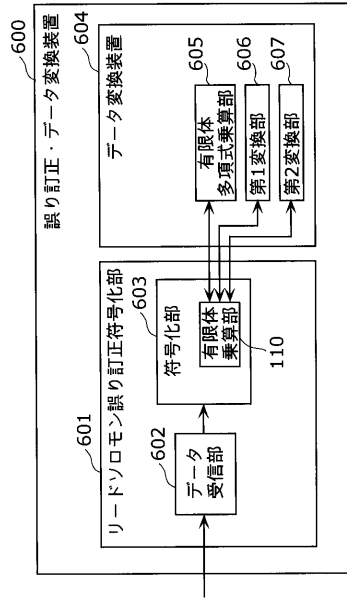
【図 5】



【図 6】



【図 11】



フロントページの続き

- (72)発明者 館林 誠
大阪府門真市大字門真1006番地 松下電器産業株式会社内
- (72)発明者 臼井 誠
大阪府門真市大字門真1006番地 松下電器産業株式会社内

審査官 金沢 史明

- (56)参考文献 特開平09-185518 (JP, A)
特開2000-321979 (JP, A)
特開2003-271056 (JP, A)
特開2002-236445 (JP, A)
特開平11-52854 (JP, A)
特開2004-164383 (JP, A)
欧州特許出願公開第1217750 (EP, A2)

- (58)調査した分野(Int.Cl., DB名)
- | | |
|------|------|
| G09C | 1/00 |
| G06F | 7/72 |