

【公報種別】特許法第 17 条の 2 の規定による補正の掲載
 【部門区分】第 6 部門第 2 区分
 【発行日】平成 24 年 9 月 27 日 (2012.9.27)

【公表番号】特表 2010-515083 (P2010-515083A)
 【公表日】平成 22 年 5 月 6 日 (2010.5.6)
 【年通号数】公開・登録公報 2010-018
 【出願番号】特願 2009-541607 (P2009-541607)
 【国際特許分類】
 G 0 9 C 1/02 (2006.01)
 【F I】
 G 0 9 C 1/02

【誤訳訂正書】
 【提出日】平成 24 年 8 月 7 日 (2012.8.7)
 【誤訳訂正 1】
 【訂正対象書類名】特許請求の範囲
 【訂正対象項目名】全文
 【訂正方法】変更
 【訂正の内容】
 【特許請求の範囲】
 【請求項 1】

暗号化デバイス上で動作する方法であって、
複数の入力シンボルを得ることと、
暗号化されるべき前記入力シンボルのおののおについて、異なるシンボルごとの置換を
定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを得ることと

、
各入力シンボルを個別に暗号化するために、前記入力シンボルのおののおについてそれ
らの対応する変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへ変換
することと、を備え、

ここにおいて、暗号化されるべき前記入力シンボルのおののおについて前記擬似ラン
ダムに選択された変換テーブルを得ることは、

第 1 の入力シンボルについての、擬似ランダムに選択された第 1 の変換テーブルを得ることと、

第 2 の入力シンボルについての、擬似ランダムに選択された第 2 の変換テーブルを得ることと、を含み、

ここにおいて、前記第 1 の変換テーブル及び前記第 2 の変換テーブルは、擬似乱数
を使用してシンボルをシャッフルすることにより生成された変換テーブルであり、

そして、ここにおいて、前記入力シンボルのおののおについてそれらの対応する変換
テーブルを使用して前記入力シンボルを対応する出力シンボルへ変換することは、

前記第 1 の変換テーブルを使用して前記第 1 の入力シンボルを第 1 の出力シンボルへ変換することと、

前記第 2 の変換テーブルを使用して前記第 2 の入力シンボルを第 2 の出力シンボルへ変換することと、を含む、方法。

【請求項 2】

前記複数の入力シンボルは、1 組の N 個のシンボルによって定義される、但し、N は、正の整数であり、変換テーブルは、前記 N 個のシンボルの置換である、請求項 1 に記載の方法。

【請求項 3】

暗号化されるべき前記入力シンボルのおののおについて、前記擬似ランダムに選択され

た変換テーブルを得ることは、

暗号化されるべき第 1 の入力シンボルについての第 1 の擬似乱数を得ることと、

前記組の N 個のシンボルの異なる置換を得るために前記組の N 個のシンボルをシャッフルすること、および前記第 1 の入力シンボルについての前記変換テーブルとしてその置換を使用することと、

を含む、

請求項 2 に記載の方法。

【請求項 4】

前記第 1 の擬似乱数は、

前記第 1 の入力シンボルについての擬似乱数を生成することと、但し、前記擬似乱数は k ビットの長さであり、k は正の整数である；

前記擬似乱数が最大数 Pmax の範囲内にあるかどうかを決定することと、但し、Pmax は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 Pmax よりも大きい場合は、捨てることと；

前記最大数 Pmax 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を得ることと；

前記第 1 の擬似乱数を得るために前記許容可能な擬似乱数のモジュロ N の階乗を除算することと；

によって得られる、

請求項 3 に記載の方法。

【請求項 5】

前記組の N 個のシンボルをシャッフルすることは、

前記組の N 個のシンボルのうちのすべてのシンボルを用いて置換ベクトル P を初期化することと、

前記第 1 の擬似乱数に基づいて前記置換ベクトルの中の前記シンボルをシャッフルすることと、

を含む、

請求項 3 に記載の方法。

【請求項 6】

前記出力シンボルを暗号解読デバイスに送信すること、

をさらに備える請求項 1 に記載の方法。

【請求項 7】

暗号化されるべき前記入力シンボルのおののおについて、異なるシンボルごとの置換を定義する前記複数の変換テーブルから、第 2 の擬似ランダムに選択された変換テーブルを得ることと、

各入力シンボルをさらに個別に暗号化するために、前記出力シンボルのおののおについてそれらの対応する第 2 の変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへ変換することと、

をさらに備える請求項 1 に記載の方法。

【請求項 8】

複数の入力シンボルを得るための手段と、

暗号化されるべき前記入力シンボルのおののおについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを得るための手段と、

各入力シンボルを個別に暗号化するために、前記入力シンボルのおののおについてそれらの対応する変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへ変換するための手段と、を備え、

第 1 の入力シンボルについての、擬似ランダムに選択された第 1 の変換テーブルを得るための手段と、

第 2 の入力シンボルについての、擬似ランダムに選択された第 2 の変換テーブルを得る

ための手段と、

ここにおいて、前記第 1 の変換テーブル及び前記第 2 の変換テーブルは、擬似乱数を使用してシンボルをシャッフルすることにより生成された変換テーブルであり、

前記第 1 の変換テーブルを使用して前記第 1 の入力シンボルを第 1 の出力シンボルへ変換するための手段と、

前記第 2 の変換テーブルを使用して前記第 2 の入力シンボルを第 2 の出力シンボルへ変換するための手段と、

をさらに備える暗号化デバイス。

【請求項 9】

暗号化されるべき第 1 の入力シンボルについての第 1 の擬似乱数を得るための手段と、

前記組の N 個のシンボルの異なる置換を得るために前記組の N 個のシンボルをシャッフルし、そして前記第 1 の入力シンボルについての前記変換テーブルとしてその置換を使用するための手段と、

をさらに備える請求項 8 に記載のデバイス。

【請求項 10】

前記第 1 の入力シンボルについての擬似乱数を生成するための手段と、但し、前記擬似乱数は k ビットの長さであり、k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを決定するための手段と、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨てるための手段と；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を得るための手段と；

前記第 1 の擬似乱数を得るために前記許容可能な擬似乱数のモジュロ N の階乗を除算するための手段と；

をさらに備える請求項 9 に記載のデバイス。

【請求項 11】

前記出力シンボルを暗号解読デバイスに送信するための手段、

をさらに備える請求項 8 に記載のデバイス。

【請求項 12】

暗号化されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の変換テーブルから、第 2 の擬似ランダムに選択された変換テーブルを得るための手段と、

各入力シンボルをさらに個別に暗号化するために、前記出力シンボルのおののについてそれらの対応する第 2 の変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへ変換するための手段と、

をさらに備える請求項 8 に記載のデバイス。

【請求項 13】

入力シンボルストリームを受信するための入力インターフェースと、

前記入力インターフェースに結合された処理回路と、

を備え、

前記処理回路は、

複数の入力シンボルを前記入力インターフェースから取得し、

暗号化されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを取得し、そして

各入力シンボルを個別に暗号化するために、前記入力シンボルのおののについてそれらの対応する変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへ変換する、

ように構成されている、

ここにおいて、暗号化されるべき前記入力シンボルのおののについて前記擬似ランダムに選択された変換テーブルを取得することは、

第 1 の入力シンボルについての、擬似ランダムに選択された第 1 の変換テーブルを取得することと、

第 2 の入力シンボルについての、擬似ランダムに選択された第 2 の変換テーブルを取得することと、を含み、

ここにおいて、前記第 1 の変換テーブル及び前記第 2 の変換テーブルは、擬似乱数を使用してシンボルをシャッフルすることにより生成された変換テーブルであり、

ここにおいて、前記入力シンボルのおののについてそれらの対応する変換テーブルを使用して前記入力シンボルを対応する出力シンボルへ変換することは、

前記第 1 の変換テーブルを使用して前記第 1 の入力シンボルを第 1 の出力シンボルへ変換することと、

前記第 2 の変換テーブルを使用して前記第 2 の入力シンボルを第 2 の出力シンボルへ変換することと、を含む、暗号化デバイス。

【請求項 1 4】

前記出力シンボルを送信するための、前記処理回路に結合された出力インターフェース、

をさらに備える請求項 1 3 に記載のデバイス。

【請求項 1 5】

前記複数の入力シンボルは、1 組の N 個のシンボルによって定義されている、但し、N は正の整数であり、変換テーブルは、前記 N 個のシンボルの置換である、請求項 1 3 に記載のデバイス。

【請求項 1 6】

前記処理回路に結合されたキーストリームジェネレータと、なお、前記キーストリームジェネレータは、暗号化されるべき第 1 の入力シンボルについて前記キーストリームジェネレータから第 1 の擬似乱数を取得するように構成されている；

前記処理回路に結合された変換テーブルジェネレータと、なお、前記変換テーブルジェネレータは、前記組の N 個のシンボルの異なる置換を取得するために、前記組の N 個のシンボルをシャッフルするように構成され、そして前記第 1 の入力シンボルについての前記変換テーブルとしてその置換を使用する；

をさらに備える請求項 1 5 に記載のデバイス。

【請求項 1 7】

前記処理回路は、さらに、

前記第 1 の入力シンボルについての擬似乱数を生成し、なお、前記擬似乱数は k ビットの長さであり、k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを決定し、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨て；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を取得し；そして

前記第 1 の擬似乱数を取得するために前記許容可能な擬似乱数のモジュロ N の階乗を除算する；

ように構成されている、

請求項 1 5 に記載のデバイス。

【請求項 1 8】

前記処理回路は、さらに、

暗号化されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の変換テーブルから、第 2 の擬似ランダムに選択された変換テーブルを取得し、そして

各入力シンボルをさらに個別に暗号化するために、前記出力シンボルのおののについ

てそれらの対応する第 2 の変換テーブルを使用して、前記出力シンボルに対応する第 2 の出力シンボルへ変換する、

ように構成されている、

請求項 15 に記載のデバイス。

【請求項 19】

プロセッサによって実行されるときに、前記プロセッサに、

複数の入力シンボルを取得させ、

暗号化されるべき前記入力シンボルのおのこのについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを取得させ、そして

各入力シンボルを個別に暗号化するために、前記入力シンボルのおのこのについてそれらの対応する変換テーブルを使用して、前記入力シンボルに対応する出力シンボルへ変換させる、

ここにおいて、暗号化されるべき前記入力シンボルのおのこのについて前記擬似ランダムに選択された変換テーブルを取得させることは、

第 1 の入力シンボルについての、擬似ランダムに選択された第 1 の変換テーブルを取得させることと、

第 2 の入力シンボルについての、擬似ランダムに選択された第 2 の変換テーブルを取得させることと、を含み、

ここにおいて、前記第 1 の変換テーブル及び前記第 2 の変換テーブルは、擬似乱数を使用してシンボルをシャッフルすることにより生成された変換テーブルであり、

ここにおいて、前記入力シンボルのおのこのについてそれらの対応する変換テーブルを使用して前記入力シンボルに対応する出力シンボルへ変換させることは、

前記第 1 の変換テーブルを使用して前記第 1 の入力シンボルを第 1 の出力シンボルへ変換させることと、

前記第 2 の変換テーブルを使用して前記第 2 の入力シンボルを第 2 の出力シンボルへ変換させることと、を含む、

シンボルを暗号化するために動作可能な 1 つまたは複数の命令、を有する機械可読媒体

。

【請求項 20】

前記複数の入力シンボルは、1 組の N 個のシンボルによって定義される、但し、N は正の整数であり、変換テーブルは、前記 N 個のシンボルの置換である、請求項 19 に記載の機械可読媒体。

【請求項 21】

プロセッサによって実行されるときに、前記プロセッサに、さらに、

暗号化されるべき第 1 の入力シンボルについての第 1 の擬似乱数を取得させ、そして

前記組の N 個のシンボルの異なる置換を取得するために、前記組の N 個のシンボルをシャッフルさせ、そして前記第 1 の入力シンボルについての前記変換テーブルとしてその置換を使用させる、

1 つまたは複数の命令、を有する請求項 20 に記載の機械可読媒体。

【請求項 22】

プロセッサによって実行されるときに、前記プロセッサに、さらに、

前記第 1 の入力シンボルについての擬似乱数を生成させ、なお、前記擬似乱数は k ビットの長さであり、k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを決定させ、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨てさせ、；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を取得させ；そして

前記第 1 の擬似乱数を取得するために前記許容可能な擬似乱数のモジュロ N の階乗を除

算させる、

1 つまたは複数の命令、を有する請求項 2 0 に記載の機械可読媒体。

【請求項 2 3】

プロセッサによって実行されるときに、前記プロセッサに、さらに、

暗号化されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の変換テーブルから、第 2 の擬似ランダムに選択された変換テーブルを取得させ、そして

各入力シンボルをさらに個別に暗号化するために、前記出力シンボルのおののについてそれらの対応する第 2 の変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへ変換させる、

1 つまたは複数の命令、を有する請求項 1 9 に記載の機械可読媒体。

【請求項 2 4】

暗号解読デバイス上でシンボルを暗号解読するための方法であって、

1 組の n 個のシンボル内で定義される複数の入力シンボルを得ることと、

暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを得ることと、

各入力シンボルを個別に暗号解読するために、前記入力シンボルのおののについてそれらの対応する逆変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへ変換することと、

を備え、

ここにおいて、暗号解読されるべき前記入力シンボルのおののについて前記擬似ランダムに選択された逆変換テーブルを得ることは、

第 1 の入力シンボルについての、擬似ランダムに選択された第 1 の逆変換テーブルを得ることと、

第 2 の入力シンボルについての、擬似ランダムに選択された第 2 の逆変換テーブルを得ることと、を含み、

ここにおいて、前記第 1 の逆変換テーブル及び前記第 2 の逆変換テーブルは、擬似乱数を使用してシンボルをシャッフルすることにより生成された逆変換テーブルであり、

ここにおいて、前記入力シンボルのおののについてそれらの対応する逆変換テーブルを使用して前記入力シンボルを対応する出力シンボルへ変換することは、

前記第 1 の逆変換テーブルを使用して前記第 1 の入力シンボルを第 1 の出力シンボルへ変換することと、

前記第 2 の逆変換テーブルを使用して前記第 2 の入力シンボルを第 2 の出力シンボルへ変換することと、を含む、方法。

【請求項 2 5】

前記複数の入力シンボルは、1 組の N 個のシンボルによって定義され、但し、 N は正の整数であり、逆変換テーブルは前記 N 個のシンボルの置換であり、

さらに前記方法は、

暗号解読されるべき第 1 の入力シンボルについての第 1 の擬似乱数を得ることと、

前記組の N 個のシンボルの異なる置換を得るために前記組の N 個のシンボルをシャッフルすること、および前記第 1 の入力シンボルについての前記逆変換テーブルとしてその置換を使用することと、

をさらに備える、

請求項 2 4 に記載の方法。

【請求項 2 6】

前記第 1 の擬似乱数は、

前記第 1 の入力シンボルについての擬似乱数を生成することと、なお、前記擬似乱数は k ビットの長さであり、 k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを決定することと、但し、 P_m

$a \times$ は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨てることと；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を得ることと；

前記第 1 の擬似乱数を得るために前記許容可能な擬似乱数のモジュロ N の階乗を除算することと；

によって得られる、請求項 2 5 に記載の方法。

【請求項 2 7】

暗号解読されるべき前記入力シンボルのおののおのについて、異なるシンボルごとの置換を定義する前記複数の逆変換テーブルから、第 2 の擬似ランダムに選択された逆変換テーブルを得ることと、

各入力シンボルをさらに個別に暗号解読するために、前記出力シンボルのおののおのについてそれらの対応する第 2 の逆変換テーブルを使用して、前記出力シンボルに対応する第 2 の出力シンボルへ変換することと、

をさらに備える請求項 2 4 に記載の方法。

【請求項 2 8】

1 組の n 個のシンボル内で定義される複数の入力シンボルを得るための手段と、

暗号解読されるべき前記入力シンボルのおののおのについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを得るための手段と、

各入力シンボルを個別に暗号解読するために前記入力シンボルのおののおのについてそれらの対応する逆変換テーブルを使用して前記入力シンボルに対応する出力シンボルへと変換するための手段と、

を備え、

第 1 の入力シンボルについての、擬似ランダムに選択された第 1 の逆変換テーブルを得るための手段と、

第 2 の入力シンボルについての、擬似ランダムに選択された第 2 の逆変換テーブルを得るための手段と、

ここにおいて、前記第 1 の逆変換テーブル及び前記第 2 の逆変換テーブルは、擬似乱数を使用してシンボルをシャッフルすることにより生成された逆変換テーブルであり、

前記第 1 の逆変換テーブルを使用して前記第 1 の入力シンボルを第 1 の出力シンボルへ変換するための手段と、

前記第 2 の逆変換テーブルを使用して前記第 2 の入力シンボルを第 2 の出力シンボルへ変換するための手段と、

をさらに備える暗号解読デバイス。

【請求項 2 9】

前記複数の入力シンボルは、1 組の N 個のシンボルによって定義され、但し、 N は正の整数であり、逆変換テーブルは前記 N 個のシンボルの置換であり、

前記デバイスはさらに、

暗号解読されるべき第 1 の入力シンボルについての第 1 の擬似乱数を得るための手段と

、

前記組の N 個のシンボルの異なる置換を得るために、前記組の N 個のシンボルをシャッフルし、そして前記第 1 の入力シンボルについての前記逆変換テーブルとしてその置換を使用するための手段と、

を備える、

請求項 2 8 に記載のデバイス。

【請求項 3 0】

前記第 1 の擬似乱数は、

前記第 1 の入力シンボルについての擬似乱数を生成するための手段と、なお、前記擬似乱数は k ビットの長さであり、 k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを、決定するための手段と、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨てるための手段と；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を得るための手段と；

前記第 1 の擬似乱数を得るために前記許容可能な擬似乱数のモジュロ N の階乗を除算するための手段と；

によって得られる、

請求項 29 に記載のデバイス。

【請求項 31】

暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の逆変換テーブルから、第 2 の擬似ランダムに選択された逆変換テーブルを得るための手段と、

各入力シンボルをさらに個別に暗号解読するために、前記出力シンボルのおののについてそれらの対応する第 2 の逆変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへと変換するための手段と、

をさらに備える請求項 28 に記載のデバイス。

【請求項 32】

入力シンボルストリームを受信するための入力インターフェースと、

前記入力インターフェースに結合された処理回路と、

を備え、

前記処理回路は、

1 組の n 個のシンボル内で定義される複数の入力シンボルを取得し、

暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを取得し、そして

各入力シンボルを個別に暗号解読するために、前記入力シンボルのおののについてそれらの対応する逆変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへ変換する、

ように構成されている、

ここにおいて、暗号解読されるべき前記入力シンボルのおののについて前記擬似ランダムに選択された逆変換テーブルを取得することは、

第 1 の入力シンボルについての、擬似ランダムに選択された第 1 の逆変換テーブルを取得することと、

第 2 の入力シンボルについての、擬似ランダムに選択された第 2 の逆変換テーブルを取得することと、を含み、

ここにおいて、前記第 1 の逆変換テーブル及び前記第 2 の逆変換テーブルは、擬似乱数を使用してシンボルをシャッフルすることにより生成された逆変換テーブルであり

、

ここにおいて、前記入力シンボルのおののについてそれらの対応する逆変換テーブルを使用して前記入力シンボルを対応する出力シンボルへ変換することは、

前記第 1 の逆変換テーブルを使用して前記第 1 の入力シンボルを第 1 の出力シンボルへ変換することと、

前記第 2 の逆変換テーブルを使用して前記第 2 の入力シンボルを第 2 の出力シンボルへ変換することと、を含む、暗号解読デバイス。

【請求項 33】

前記複数の入力シンボルは、1 組の N 個のシンボルによって定義され、但し、 N は正の整数であり、逆変換テーブルは前記 N 個のシンボルの置換であり、

前記デバイスはさらに、

前記処理回路に結合されたキーストリームジェネレータと、なお、前記キーストリームジェネレータは、暗号解読されるべき第1の入力シンボルについて前記キーストリームジェネレータから第1の擬似乱数を取得するように構成されている；

前記処理回路に結合された逆変換テーブルジェネレータと、なお、前記逆変換テーブルジェネレータは、前記組のN個のシンボルの異なる置換を取得するために、前記組のN個のシンボルをシャッフルするように構成され、そして前記第1の入力シンボルについての前記逆変換テーブルとしてその置換を使用する；

を備える請求項32に記載のデバイス。

【請求項34】

前記複数の入力シンボルは、1組のN個のシンボルによって定義され、但し、Nは正の整数であり、逆変換テーブルは前記N個のシンボルの置換であり、

前記処理回路は、さらに、

前記第1の入力シンボルについての擬似乱数を生成し、なお、前記擬似乱数はkビットの長さであり、kは正の整数である；

前記擬似乱数が最大数Pmaxの範囲内にあるかどうかを、決定し、但し、Pmaxは、最大しきい値 2^k よりも小さいNの階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数Pmaxよりも大きい場合は、捨て；

前記最大数Pmax以下である許容可能な擬似乱数が得られるまで、前記第1の入力シンボルについての異なる擬似乱数を取得し；そして

前記第1の擬似乱数を得るために前記許容可能な擬似乱数のモジュロNの階乗を除算する；

ように構成されている、

請求項32に記載のデバイス。

【請求項35】

前記処理回路は、さらに、

暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の逆変換テーブルから、第2の擬似ランダムに選択された逆変換テーブルを取得し、そして

各入力シンボルをさらに個別に暗号解読するために、前記出力シンボルのおののについてそれらの対応する第2の逆変換テーブルを使用して、前記出力シンボルを対応する第2の出力シンボルへと変換する、

ように構成されている、

請求項32に記載のデバイス。

【請求項36】

プロセッサによって実行されるときに、前記プロセッサに、

1組のn個のシンボル内で定義される複数の入力シンボルを取得させ、

暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを取得させ、そして

各入力シンボルを個別に暗号解読するために、前記入力シンボルのおののについてそれらの対応する逆変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへと変換させる、

ここにおいて、暗号解読されるべき前記入力シンボルのおののについて前記擬似ランダムに選択された逆変換テーブルを取得させることは、

第1の入力シンボルについての、擬似ランダムに選択された第1の逆変換テーブルを取得させることと、

第2の入力シンボルについての、擬似ランダムに選択された第2の逆変換テーブルを取得させることと、を含み、

ここにおいて、前記第1の逆変換テーブル及び前記第2の逆変換テーブルは、擬似乱数を使用してシンボルをシャッフルすることにより生成された逆変換テーブルであり、

ここにおいて、前記入力シンボルのおのおのについてそれらの対応する逆変換テーブルを使用して前記入力シンボルを対応する出力シンボルへ変換させることは、

前記第 1 の逆変換テーブルを使用して前記第 1 の入力シンボルを第 1 の出力シンボルへ変換させることと、

前記第 2 の逆変換テーブルを使用して前記第 2 の入力シンボルを第 2 の出力シンボルへ変換させることと、を含む、

シンボルを暗号解読するための 1 つまたは複数の命令、を有する機械可読媒体。

【請求項 3 7】

前記複数の入力シンボルは、1 組の N 個のシンボルによって定義され、但し、N は正の整数であり、逆変換テーブルは前記 N 個のシンボルの置換であり、

そして前記機械可読媒体は、

プロセッサによって実行されるときに、前記プロセッサに、さらに、

暗号解読されるべき第 1 の入力シンボルについての第 1 の擬似乱数を取得させ、そして前記組の N 個のシンボルの異なる置換を取得するために、前記組の N 個のシンボルをシャッフルさせ、そして前記第 1 の入力シンボルについての前記逆変換テーブルとしてその置換を使用させる、

1 つまたは複数の命令、をさらに備える、

請求項 3 6 に記載の機械可読媒体。

【請求項 3 8】

プロセッサによって実行されるときに、前記プロセッサに、さらに、

暗号解読されるべき前記入力シンボルのおのおのについて、異なるシンボルごとの置換を定義する前記複数の逆変換テーブルから、第 2 の擬似ランダムに選択された逆変換テーブルを取得させ、そして

各入力シンボルをさらに個別に暗号解読するために、前記出力シンボルのおのおのについてそれらの対応する第 2 の逆変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへと変換させる、

1 つまたは複数の命令、を有する請求項 3 6 に記載の機械可読媒体。

【誤訳訂正 2】

【訂正対象書類名】明細書

【訂正対象項目名】全文

【訂正方法】変更

【訂正の内容】

【発明の詳細な説明】

【発明の名称】組み合わせコンバイナ暗号化方法

【背景】

【0 0 0 1】

[分野]

様々な例は、一般にセキュリティ保護された通信(secure communications)に関し、より詳細には、変換テーブル(translation tables)を効率的に生成することにより、シンボル(symbols)をセキュリティ保護する(secures)ストリーム暗号化(stream encryption)のための方法に関する。

【0 0 0 2】

[背景]

ストリーム暗号化は、通常、暗号化された出力または暗号文を生成するために、擬似乱数(pseudo-random numbers)のキーストリーム(keystream)を生成することと、それらを平文シンボル(plaintext symbols)と組み合わせることと、によって進む。普通は、2 進キーストリームシンボルと 2 進平文シンボルは、排他的論理和(Exclusive-OR) (XOR) 演算を使用してビット毎に組み合わせられる、というのは、それは自己反転型(self-inversive)であるからである。しかしながら、平文シンボル上でビット毎の暗号化(bit-wise encryption)を実行するのではなくて、全体の平文シンボル(whole plaintext symbol)を暗

号化することが、時に望ましいこともある。それ故に、X O R 演算は、使用されることができない。一般的には、平文シンボルは、暗号文シンボル(ciphertext symbol)を得るために、キーストリームシンボルのモジュロ n (keystream symbol modulo n) に加算される。しかし、アクティブな攻撃者(attacker)は、特定のシンボルの位置(position)を知り、送信された暗号文シンボルから減算することにより、そのシンボルを変更することができる可能性がある。例えば、1組の平文ディジット(plaintext digits) (0 から 9) は、キーストリームのモジュロ 10 からの擬似ランダムに(pseudorandomly)生成されたディジット(digit) (0 から 9) に各ディジットを加算することにより、暗号化されることができる。しかしながら、特定の平文ディジットが、「1」であったが、出力暗号文ディジットは、「7」であったことを知っている攻撃者は、キーストリームディジットが、「6」であったことを決定することができ、そしてそのときには、その特定のシンボル位置のそれらの選択についての他の任意のディジットを正しく暗号化することができる。攻撃者は、平文シンボルと、キーストリームシンボルとがどのようにして組み合わせられているかを決定することができるので、そのような部分的な暗号解読は、暗号化された情報の残りについてのセキュリティを弱める。すなわち、ひとたび暗号文シンボルと、平文シンボルとの間の関係

が見出された後に、その情報は、他の暗号文シンボルが、攻撃者によって暗号解読されることを可能にすることができるので、残りの暗号文シンボルのセキュリティは、損なわれる。さらに、簡単な数学的演算(例えば、加算、減算、乗算など)は、速く、そして効率的に平文シンボルと、キーストリームシンボルとを組み合わせるが、より複雑な数学的関数を使用することは、暗号化において注目し値する遅延を引き起こし、あるいはより多くの処理リソースを要求する可能性がある。

【0003】

したがって、もし、1つの暗号化されたシンボルが見出されるあるいは解読される(discovered or broken)場合は、他のシンボルのセキュリティ(security)を弱めることのない効果的な暗号化方法が必要とされる。

【発明の概要】

【0004】

データをセキュリティ保護するための、暗号化デバイス(encrypting device)上で動作する方法が、提供される。複数の入力シンボルが、暗号化デバイスによって得られる。擬似ランダムに選択された変換テーブルが、暗号化されるべき入力シンボルのおののについて、異なるシンボルごとの置換(different symbol-to-symbol permutations)を定義する複数の変換テーブルから、得られる。入力シンボルは、各入力シンボルを個別に暗号化するために、入力シンボルのおののについてそれらの対応する変換テーブルを使用して、対応する出力シンボルへ変換される。出力シンボルは、暗号解読デバイス(decrypting device)に送信されることができる。

【0005】

さらに、第2の擬似ランダムに選択された変換テーブルが、暗号化されるべき入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の変換テーブルから得られることができる。出力シンボルは、各入力シンボルをさらに個別に暗号化するために、出力シンボルのおののについてそれらの対応する第2の変換テーブルを使用して、対応する第2の出力シンボルへ変換されることができる。

【0006】

一例においては、暗号化されるべき入力シンボルのおののについて擬似ランダムに選択された変換テーブルを得ることは、(a)第1の入力シンボルについての、擬似ランダムに選択された第1の変換テーブルを得ることと、(b)第2の入力シンボルについての、擬似ランダムに選択された第2の変換テーブルを得ることと、を含むことができる。入力シンボルのおののについてそれらの対応する変換テーブルを使用して入力シンボルを対応する出力シンボルへ変換することは、(a)第1の変換テーブルを使用して第1の入力シンボルを第1の出力シンボルへ変換することと、(b)第2の変換テーブルを使用し

て第 2 の入力シンボルを第 2 の出力シンボルへ変換することと、を含むことができる。複数の入力シンボルは、1 組の N 個のシンボル(a set of N symbols)によって定義されることができ、但し、 N は、正の整数であり、変換テーブルは、 N 個のシンボルの置換である。

【0007】

別の例においては、暗号化されるべき入力シンボルのおののについて、擬似ランダムに選択された変換テーブルを得ることは、(a) 暗号化されるべき第 1 の入力シンボルについての第 1 の擬似乱数を得ることと、(b) 該組の N 個のシンボルの異なる置換を得るために該組の N 個のシンボルをシャッフルすること(shuffling)、および第 1 の入力シンボルについての変換テーブルとしてその置換を使用することと、を含むことができる。

【0008】

1 つのコンフィギュレーション(configuration)においては、第 1 の擬似乱数は、(a) 第 1 の入力シンボルについての擬似乱数を生成すること(generating)、但し、擬似乱数は、 k ビットの長さであり、そして k は、正の整数である；(b) 擬似乱数が最大数 P_{max} の範囲内にあるかどうか、を決定すること(determining)、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数(the largest multiple of N factorial)である；(c) 擬似乱数が最大数 P_{max} よりも大きい場合は、擬似乱数を捨てること(discarding)；(d) 最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、第 1 の入力シンボルについての異なる擬似乱数を得ること(obtaining)；および / または (e) 第 1 の擬似乱数を得るために許容可能な擬似乱数のモジュロ N の階乗を除算すること(dividing the acceptable pseudorandom number modulo N factorial)；によって得られることができる。該組の N 個のシンボルをシャッフルすることは、(a) 該組の N 個のシンボルのうちのすべてのシンボルを用いて置換ベクトル(permutation vector) P を初期化することと、(b) 第 1 の擬似乱数に基づいて置換ベクトルの中のシンボルをシャッフルすることと、を含むことができる。

【0009】

暗号化デバイス(encryption device)もまた、提供され、入力インターフェースと、処理回路と、を備える。入力インターフェースは、入力シンボルストリームを受信する役割を果たすことができる。処理回路は、(a) 入力インターフェースから複数の入力シンボルを取得し、(b) 暗号化されるべき入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを取得し、かつ / または (c) 各入力シンボルを個別に暗号化するために、入力シンボルのおののについてそれらの対応する変換テーブルを使用して、入力シンボルを対応する出力シンボルへ変換する、ように構成されていることができる。暗号化デバイスは、出力シンボルを送信するための、処理回路に結合された出力インターフェースを含むこともできる。

【0010】

複数の入力シンボルは、1 組の N 個のシンボルによって定義されることができ、但し、 N は、正の整数であり、そして変換テーブルは、 N 個のシンボルの置換である。暗号化デバイスはまた、(a) 処理回路に結合されたキーストリームジェネレータと、なお、キーストリームジェネレータは、暗号化されるべき第 1 の入力シンボルについて第 1 の擬似乱数をキーストリームジェネレータから得るように構成されている；および / または (b) 処理回路に結合された変換テーブルジェネレータと、なお、変換テーブルジェネレータは、該組の N 個のシンボルの異なる置換を得るために該組の N 個のシンボルをシャッフルする(shuffle)ように構成され、第 1 の入力シンボルについての変換テーブルとしてその置換を使用する；も含むことができる。

【0011】

一例においては、処理回路は、さらに、(a) 第 1 の入力シンボルについての擬似乱数を生成し、なお擬似乱数は、 k ビットの長さであり、そして k は、正の整数である；(b) 擬似乱数が最大数 P_{max} の範囲内にあるかどうか、を決定し、但し、 P_{max} は、最

大きい値 2^k よりも小さい N の階乗の最大の倍数である；(c) 擬似乱数が、もし最大数 P_{max} よりも大きい場合は、擬似乱数を捨て；(d) 最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、第 1 の入力シンボルについての異なる擬似乱数を取得し；かつ/または(e) 第 1 の擬似乱数を得るために許容可能な擬似乱数 modulo N の階乗を除算する (divide the acceptable pseudorandom modulo N factorial)；ように構成されることもできる。

【0012】

1 つのコンフィギュレーションにおいては、暗号化デバイスの処理回路はまた、(a) 暗号化されるべき入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、第 2 の擬似ランダムに選択された変換テーブルを取得し、かつ/または(b) 各入力シンボルをさらに個別に暗号化するために、出力シンボルのおののについてそれらの対応する第 2 の変換テーブルを使用して、出力シンボルを対応する第 2 の出力シンボルへと変換する、ように構成されることもできる。

【0013】

その結果として、(a) 複数の入力シンボルを得るための手段、(b) 暗号化されるべき入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを得るための手段、(c) 各入力シンボルを個別に暗号化するために入力シンボルのおののについてそれらの対応する変換テーブルを使用して、入力シンボルを対応する出力シンボルへ変換するための手段、および/または(d) 出力シンボルを暗号解読デバイスに送信するための手段、を備える暗号化デバイスが、提供される。暗号化デバイスはまた、(a) 暗号化されるべき入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、第 2 の擬似ランダムに選択された変換テーブルを得るための手段、および/または(b) 各入力シンボルをさらに個別に暗号化するために、出力シンボルのおののについてそれらの対応する第 2 の変換テーブルを使用して、出力シンボルを対応する第 2 の出力シンボルへ変換するための手段、を含むこともできる。

【0014】

1 つのコンフィギュレーションにおいては、暗号化デバイスは、(a) 第 1 の入力シンボルについての擬似ランダムに選択された第 1 の変換テーブルを得るための手段、(b) 第 2 の入力シンボルについての擬似ランダムに選択された第 2 の変換テーブルを得るための手段、(c) 第 1 の変換テーブルを使用して第 1 の入力シンボルを第 1 の出力シンボルへと変換するための手段、および/または(d) 第 2 の変換テーブルを使用して第 2 の入力シンボルを第 2 の出力シンボルへ変換するための手段、を含むことができる。

【0015】

一例においては、暗号化デバイスはまた、(a) 暗号化されるべき第 1 の入力シンボルについて第 1 の擬似乱数を得るための手段、および/または(b) 該組の N 個のシンボルの異なる置換を得るために該組の N 個のシンボルをシャッフルし、そして第 1 の入力シンボルについての変換テーブルとしてその置換を使用するための手段、を含むこともできる。

【0016】

プロセッサによって実行されるときに、プロセッサに、(a) 複数の入力シンボルを取得させ、(b) 暗号化されるべき入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを取得させ、かつ/または(c) 各入力シンボルを個別に暗号化するために入力シンボルのおののについてそれらの対応する変換テーブルを使用して入力シンボルを対応する出力シンボルへ変換させる、シンボルを暗号化するために動作可能な 1 つまたは複数の命令、を有する機械可読媒体もまた、提供される。複数の入力シンボルは、1 組の N 個のシンボルによって定義されることができる、但し、 N は、正の整数であり、そして変換テーブルは、 N 個のシンボルの置換である。

【0017】

機械可読媒体は、プロセッサによって実行されるときに、プロセッサに、さらに、(a) 暗号化されるべき第1の入力シンボルについての第1の乱数を取得させ、かつ/または (b) 該組のN個のシンボルの異なる置換を取得するために該組のN個のシンボルをシャッフルし、そして第1の入力シンボルについての変換テーブルとしてその置換を使用させる1つまたは複数の命令、を含むこともできる。

【0018】

1つのコンフィギュレーションにおいては、機械可読媒体は、プロセッサによって実行されるときに、プロセッサに、さらに、(a) 第1の入力シンボルについての擬似乱数を生成させ、なお、擬似乱数は、kビットの長さであり、そしてkは、正の整数である；(b) 擬似乱数が最大数 P_{max} の範囲内にあるかどうか、を決定させ、但し、 P_{max} は、最大しきい値 2^k よりも小さいNの階乗の最大の倍数である；(c) 擬似乱数が、もし最大数 P_{max} よりも大きい場合は、擬似乱数を捨てさせ；(d) 最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、第1の入力シンボルについての異なる擬似乱数を取得させ；かつ/または (e) 第1の擬似乱数を得るために許容可能な擬似乱数のモジュロNの階乗を除算させる、1つまたは複数の命令、を含むこともできる。

【0019】

シンボルを暗号解読するための方法もまた提供される。1組のn個のシンボル内で(within a set of n symbols)定義される複数の入力シンボルが、得られる。擬似ランダムに選択された逆変換テーブル(reverse translation table)が、暗号解読されるべき入力シンボルのおのおのについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、選択される。次いで、入力シンボルは、各入力シンボルを個別に暗号解読するために、入力シンボルのおのおのについてそれらの対応する逆変換テーブルを使用して、対応する出力シンボルへ変換される。複数の入力シンボルは、1組のN個のシンボルによって定義されることができ、但し、Nは、正の整数であり、逆変換テーブルは、N個のシンボルの置換である。本方法は、(a) 暗号解読されるべき第1の入力シンボルについての第1の擬似乱数を得ること、および/または (b) 該組のN個のシンボルの異なる置換を得るために該組のN個のシンボルをシャッフルし、そして第1の入力シンボルについての逆変換テーブルとしてその置換を使用すること、をさらに含むことができる。

【0020】

一例においては、第1の擬似乱数は、(a) 第1の入力シンボルについての擬似乱数を生成すること、なお擬似乱数は、kビットの長さであり、そしてkは、正の整数である；(b) 擬似乱数が最大数 P_{max} の範囲内にあるかどうか、を決定すること、但し、 P_{max} は、最大しきい値 2^k よりも小さいNの階乗の最大の倍数である；(c) 擬似乱数が、もし最大数 P_{max} よりも大きい場合は、擬似乱数を捨てること；(d) 最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、第1の入力シンボルについての異なる擬似乱数を得ること；および/または (e) 第1の擬似乱数を得るために許容可能な擬似乱数のモジュロNの階乗を除算すること；によって得られることができる。

【0021】

別のコンフィギュレーションにおいては、本方法は、(a) 暗号解読されるべき入力シンボルのおのおのについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、第2の擬似ランダムに選択された逆変換テーブルを得ること、および/または (b) 各入力シンボルをさらに個別に暗号解読するために、出力シンボルのおのおのについてそれらの対応する第2の逆変換テーブルを使用して、出力シンボルを対応する第2の出力シンボルへ変換すること、をさらに含むことができる。

【0022】

入力インターフェースと処理回路とを備える暗号解読デバイスが提供される。入力インターフェースは、入力シンボルストリームを受信することができる。処理回路は、(a) 1組のn個のシンボル内で定義される複数の入力シンボルを取得し、(b) 暗号解読されるべき入力シンボルのおのおのについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを取得し、かつ/または (

c) 各入力シンボルを個別に暗号解読するために入力シンボルのおののおについてそれらの対応する逆変換テーブルを使用して入力シンボルに対応する出力シンボルへ変換する、ように構成されることができる。

【0023】

複数の入力シンボルは、1組のN個のシンボルによって定義されることができる、但し、Nは、正の整数であり、そして逆変換テーブルは、N個のシンボルの置換である。暗号解読デバイスはまた、(a) 処理回路に結合されたキーストリームジェネレータと、なお、キーストリームジェネレータは、暗号解読されるべき第1の入力シンボルについてキーストリームジェネレータから第1の擬似乱数を得るように構成されている；および/または(b) 処理回路に結合された逆変換テーブルジェネレータと、なお、逆変換テーブルジェネレータは、該組のN個のシンボルの異なる置換を得るために該組のN個のシンボルをシャッフルするように構成され、そして第1の入力シンボルについての逆変換テーブルとしてその置換を使用する；を含むこともできる。

【0024】

複数の入力シンボルは、1組のN個のシンボルによって定義されることができる、但し、Nは、正の整数であり、そして逆変換テーブルは、N個のシンボルの置換である。処理回路は、さらに、(a) 第1の入力シンボルについての擬似乱数を生成し、なお擬似乱数は、kビットの長さであり、そしてkは、正の整数である；(b) 擬似乱数が最大数 P_{max} の範囲内にあるかどうか、を決定し、但し、 P_{max} は、最大しきい値 2^k よりも小さいNの階乗の最大の倍数である；(c) 擬似乱数が、もし最大数 P_{max} よりも大きい場合は、擬似乱数を捨て；(d) 最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、第1の入力シンボルについての異なる擬似乱数を取得し；かつ/または(e) 第1の擬似乱数を得るために許容可能な擬似乱数のモジュロNの階乗を除算する、ように構成されることができる。

【0025】

別の例においては、処理回路は、さらに、(a) 暗号解読されるべき入力シンボルのおののおについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、第2の擬似ランダムに選択された逆変換テーブルを取得し、かつ/または(b) 各入力シンボルをさらに個別に暗号解読するために出力シンボルのおののおについてそれらの対応する第2の逆変換テーブルを使用して、出力シンボルに対応する第2の出力シンボルへ変換する、ように構成されることができる。

【0026】

その結果として、(a) 1組のn個のシンボル内で定義される複数の入力シンボルを得るための手段、(b) 暗号解読されるべき入力シンボルのおののおについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを得るための手段、および/または(c) 各入力シンボルを個別に暗号解読するために入力シンボルのおののおについてそれらの対応する逆変換テーブルを使用して入力シンボルに対応する出力シンボルへと変換するための手段、を備える暗号解読デバイスもまた、提供される。複数の入力シンボルは、1組のN個のシンボルによって定義されることができる、但し、Nは、正の整数であり、そして逆変換テーブルは、N個のシンボルの置換である。暗号解読デバイスは、(a) 暗号解読されるべき第1の入力シンボルについて第1の擬似乱数を得るための手段、および/または(b) 該組のN個のシンボルの異なる置換を得るために該組のN個のシンボルをシャッフルし、そして第1の入力シンボルについての逆変換テーブルとしてその置換を使用するための手段、をさらに含むこともできる。第1の擬似乱数は、(a) 第1の入力シンボルについての擬似乱数を生成するための手段、なお擬似乱数は、kビットの長さであり、そしてkは、正の整数である；(b) 擬似乱数が最大数 P_{max} の範囲内にあるかどうか、を決定するための手段、但し、 P_{max} は、最大しきい値 2^k よりも小さいNの階乗の最大の倍数である；(c) 擬似乱数が、もし最大数 P_{max} よりも大きい場合は、擬似乱数を捨てるための手段；(d) 最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、第1の入力シンボルについての異

なる擬似乱数を得るための手段；および／または（e）第１の擬似乱数を得るために許容可能な擬似乱数のモジュロNの階乗を除算するための手段；によって得られる。

【００２７】

プロセッサによって実行されるときに、プロセッサに、（a）１組のn個のシンボル内で定義される複数の入力シンボルを取得させ、（b）暗号解読されるべき入力シンボルのおのおのについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを取得させ、かつ／または（c）各入力シンボルを個別に暗号解読するために、入力シンボルのおのおのについてそれらの対応する逆変換テーブルを使用して、入力シンボルに対応する出力シンボルへ変換させる、シンボルを暗号解読するための１つまたは複数の命令、を有する機械可読媒体もまた、提供される。複数の入力シンボルは、１組のN個のシンボルによって定義される、但し、Nは、正の整数であり、そして逆変換テーブルは、N個のシンボルの置換である。

【００２８】

機械可読媒体はまた、プロセッサによって実行されるときに、プロセッサに、さらに、（a）暗号解読されるべき第１の入力シンボルについての第１の擬似乱数を取得させ、かつ／または（b）該組のN個のシンボルの異なる置換を得るために該組のN個のシンボルをシャッフルし、そして第１の入力シンボルについての逆変換テーブルとしてその置換を使用させる、１つまたは複数の命令、を備えることもできる。

【００２９】

別の例においては、機械可読媒体はまた、プロセッサによって実行されるときに、プロセッサに、さらに、（a）暗号解読されるべき入力シンボルのおのおのについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、第２の擬似ランダムに選択された逆変換テーブルを取得させ、かつ／または（b）各入力シンボルをさらに個別に暗号解読するために、出力シンボルのおのおのについてそれらの対応する第２の逆変換テーブルを使用して、出力シンボルに対応する第２の出力シンボルへ変換させる、１つまたは複数の命令、を備えることもできる。

【図面の簡単な説明】

【００３０】

【図１】図１は、セキュリティデバイスが、電話とセキュリティ保護されたサーバとの間のある種の通信を、セキュリティ保護するために、通信回線に沿って電話に結合されることが出来るシステムを示す。

【図２】図２は、図１の電話と発行機関(issuing institution)に属するセキュリティサーバとの間のある種の通信を、セキュリティ保護するための一方法を示す流れ図である。

【図３】図３は、送信中にDTMFトーンをセキュリティ保護することを可能にすることができるテレサービスセキュリティサーバ(tele-services security server)の一例のブロック図を示す。

【図４】図４は、電話デバイスからのDTMFトーンをセキュリティ保護するための、セキュリティサーバ上で動作する一方法を示す。

【図５】図５は、送信中にDTMFトーンを保護するように構成されることが出来るセキュリティデバイスの一例のブロック図を示す。

【図６】図６は、電話デバイスからのDTMFトーンをセキュリティ保護するための、セキュリティデバイス上で動作する一方法を示す。

【図７】図７は、セキュリティサーバを用いてそれ自体を認証するように構成されたモバイル通信デバイスのブロック図である。

【図８】図８は、通信ネットワーク上でテレサービス局に対してモバイル通信デバイスを認証するための一方法を示す流れ図である。

【図９】図９は、暗号化されるべき各シンボルについて変換テーブルを擬似ランダムに選択することにより、平文シンボルをセキュリティ保護するための組み合わせコンバイナのブロック図を示す。

【図１０】図１０は、平文シンボルを暗号化されたシンボルに変換するための、シンボル

ごとの変換テーブル(a symbol-to-symbol translation table)の一例を示す。

【図 1 1】図 1 1 は、どのようにして平文シンボルが、暗号化されたシンボルを得るために異なる変換テーブルを使用して暗号化されることができるのかの一例を示す。

【図 1 2】図 1 2 は、1 組の n 個のシンボルについての複数の可能な置換から変換テーブルを選択するためのアルゴリズムを示しており、ここで n は、正の整数である。

【図 1 3】図 1 3 は、単一の平文シンボルを暗号化するために複数の変換テーブルを使用することにより、シンボル認証を達成することができる別の暗号化スキームを示すブロック図である。

【図 1 4】図 1 4 は、どのようにして複数の変換テーブルが、対応する暗号化されたシンボルを得るために各平文シンボルを暗号化するために使用されることができるのかを示す。

【図 1 5】図 1 5 は、どのようにして 2 つの変換テーブルが、平文シンボルを暗号化されたシンボルへ変換または暗号化するために使用されることができるのかの一例を示す。

【図 1 6】図 1 6 は、一例による、平文暗号化を実行するための一方法を示す。

【図 1 7】図 1 7 は、どのようにして暗号化されたシンボルが、単一の平文シンボルを得るために 1 つまたは複数の逆変換テーブルを使用することにより暗号解読されることができるのかを示すブロック図である。

【図 1 8】図 1 8 は、一例による、平文暗号化を実行するための一方法を示す。

【図 1 9】図 1 9 は、一例による暗号化モジュールを示すブロック図である。

【図 2 0】図 2 0 は、一例による暗号解読モジュールを示すブロック図である。

【詳細な説明】

【0031】

以下の説明においては、具体的な詳細が、例の十分な理解を提供するために与えられる。しかしながら、それらの例は、これらの特定の詳細なしに実行されることができることが、当業者によって理解されるであろう。例えば、回路は、それらの例を不必要に詳細にあいまいにしないようにするために、ブロック図の形で示されなくてもよい。

【0032】

また、例は、フローチャート、流れ図、構造図、またはブロック図として示されるプロセスとして説明されることができることにも注意すべきである。フローチャートは、逐次的なプロセスとしてオペレーションを説明することができるが、オペレーションの多くは、並列に、あるいは同時に実行されることができる。さらに、オペレーションの順序は、並べ換えられる(re-arranged)ことができる。プロセスは、そのオペレーションが完了されるときに終了される。プロセスは、方法、ファンクション(function)、プロシージャ、サブルーチン、サブプログラムなどに対応することができる。プロセスが、ファンクションに対応するとき、その終了は、呼出しファンクション(calling function)または主要ファンクションに対するファンクションのリターン(return)に対応する。

【0033】

さらに、ストレージ媒体は、読取り専用メモリ(read-only memory)(ROM)、ランダムアクセスメモリ(random access memory)(RAM)、磁気ディスクストレージ媒体、光ストレージ媒体、フラッシュメモリデバイス、および/または情報を記憶するための他の機械可読媒体を含めて、データを記憶するための 1 つまたは複数のデバイスを表すことができる。用語「機械可読媒体」は、それだけには限定されないが、ポータブルストレージデバイスまたは固定ストレージデバイスと、光ストレージデバイスと、ワイヤレスチャネルと、命令(単数または複数)および/またはデータを記憶し、含み、あるいは搬送することができる様々な他の媒体と、を含む。

【0034】

さらに、様々なコンフィギュレーションは、ハードウェア、ソフトウェア、ファームウェア、ミドルウェア、マイクロコード、あるいはそれらの組合せによってインプリメントされることができる。ソフトウェア、ファームウェア、ミドルウェア、またはマイクロコードの形でインプリメントされるときには、説明されたタスクを実行するプログラムコー

ドまたはコードセグメントは、ストレージ媒体や他のストレージ手段などの機械可読媒体に記憶されることができる。プロセッサは、定義されたタスクを実行することができる。コードセグメントは、プロシージャ、ファンクション、サブプログラム、プログラム、ルーチン、サブルーチン、モジュール、ソフトウェアパッケージ、クラス、あるいは命令、データ構造、またはプログラムステートメントの組合せを表すことができる。コードセグメントは、情報、データ、引数(arguments)、パラメータ、またはメモリ内容を渡すこと、および/または受け取ることにより、別のコードセグメントまたはハードウェア回路に結合されることができる。情報、引数、パラメータ、データなどは、とりわけメモリ共有、メッセージパッシング(message passing)、トークンパッシング(token passing)、およびネットワーク送信を含めて適切な手段を経由して、渡され、転送され、または送信されることができる。ここにおいて開示される方法は、ハードウェア、ソフトウェア、またはそれらの両方の形でインプリメントされることができる。

【0035】

ここにおいて開示される例に関連して説明される様々な例示的な論理的なブロック、モジュール、回路、要素、および/またはコンポーネントは、ここにおいて説明される機能を実行するように設計された汎用プロセッサ、デジタル信号プロセッサ(digital signal processor) (DSP)、特定用途向け集積回路(application specific integrated circuit) (ASIC)、フィールドプログラマブルゲートアレイ(field programmable gate array) (FPGA)または他のプログラマブルロジックコンポーネント、個別ゲートまたはトランジスタロジック、個別ハードウェアコンポーネント、あるいはそれらの任意の組合せを用いてインプリメントされ、または実行されることができる。汎用プロセッサは、マイクロプロセッサとすることができるが、代替案においては、プロセッサは、従来の任意のプロセッサ、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もできる。プロセッサはまた、コンピューティングコンポーネント(computing component)の組合せとして、例えばDSPとマイクロプロセッサとの組合せ、いくつかのマイクロプロセッサ、DSPコアと組み合わせられた1つまたは複数のマイクロプロセッサ、あるいは他のそのような任意のコンフィギュレーションとしてインプリメントされることができる。

【0036】

ここにおいて開示される例に関連して説明される方法またはアルゴリズムは、処理ユニット、プログラミング命令、または他の指示の形態において、ハードウェアの形で、プロセッサにより実行可能なソフトウェアモジュールの形で、あるいは両方の組合せの形で直接に実施されることができ、そして単一のデバイスの中に含まれ、あるいは複数のデバイスを通して分散されることができる。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、着脱可能ディスク、CD-ROM、あるいは当技術分野において知られている他の任意の形態のストレージ媒体の中に存在することができる。ストレージ媒体は、プロセッサが、そのストレージ媒体から情報を読み取り、そのストレージ媒体へと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ことができる。代替案においては、ストレージ媒体は、プロセッサと一体化していてもよい。

【0037】

1つの特徴は、顧客の電話回線と直列に挿入されることができる小型フォームファクタ(small form-factor)のセキュリティデバイスを提供し、これは、2ファクタの認証スキームにおける第2のファクタとしての役割を果たし、DTMFトーンを暗号化し、それによって慎重な扱いを要する情報(sensitive information)の開示を防止する。本デバイスは、電話の通常オペレーションには干渉しない。本デバイスは、それらが関連づけられるバンクおよび支払いのサービスについてブランドの機会(branding opportunity)を提供することもできる小型フォームファクタの容器(enclosure)を含むことができる。本デバイスは、それが結合される電話回線から電力を供給されることができる。1つのコンフィギュレーションにおいては、複数のそのようなデバイスは、複数の異なるパーティ(例えば、銀行)とのセキュリティ保護された通信を提供するために電話回線に沿ってチェーン接

続され(chained)、あるいはカスケード接続されることができる。

【 0 0 3 8 】

別の特徴は、暗号化されたシンボルのセキュリティを保護する効率的な暗号化方法を提供する。各平文シンボルは、別個の擬似ランダムに選択された変換テーブルを使用することにより暗号化される。シンボルのあらゆる可能な置換を変換テーブルとしてあらかじめ記憶するのではなくて、変換テーブルは、擬似乱数と、シンボルシャッフリングアルゴリズム(symbol shuffling algorithm)とに基づいてオンザフライで(on-the-fly)効率的に生成されることができる。受信デバイスは、同様に、受信された暗号化シンボルを暗号解読するために、オンザフライで逆変換テーブルを生成することもできる。

【 0 0 3 9 】

D T M F トーンをセキュリティ保護すること(Securing DTMF Tones)

図 1 は、セキュリティデバイス 1 0 2 (security device)が、電話 1 0 4 と、セキュアサーバ(secure server) 1 0 8 との間のある種の通信をセキュリティ保護するために通信回線に沿って電話 1 0 4 に結合されることができるシステムを示している。セキュリティデバイス 1 0 2 は、電話 1 0 4 と、通信ネットワーク 1 0 6 との間の電話回線上でインラインに、あるいは直列に接続されることができる小型フォームファクタのデバイスとすることができる。セキュリティデバイス 1 0 2 は、電話 1 0 4 の近くの、またはそれに近接した電話回線に結合されることができる。

【 0 0 4 0 】

一例においては、セキュリティデバイス 1 0 2 は、アカウントおよび/または発行機関(issuing institution) 1 0 8 (例えば、銀行、クレジットカード会社など)に関連づけられることができる。例えば、銀行は、そのようなセキュリティデバイス 1 0 2 をその顧客に対して支給することができ、各セキュリティデバイスは、顧客、または顧客のアカウントに一意に関連づけられる。発行機関 1 0 8 は、顧客との電話トランザクションを容易にするセキュリティサーバ 1 1 0 を有することができる。

【 0 0 4 1 】

図 2 は、図 1 の電話 1 0 4 と、発行機関 1 0 8 に属するセキュリティサーバ 1 1 0 との間のある種の通信をセキュリティ保護するための一方法を示す流れ図である。セキュリティデバイス 1 0 2 は、アクティブモードのオペレーションと、非アクティブ(パッシブ)モードのオペレーションとを有することができる。セキュリティデバイス 1 0 2 が、発行機関 1 0 8 (例えば、セキュリティサーバ 1 1 0)以外の誰かを呼び出すために使用されるときには、それは非アクティブであり、そしてその呼出しは、D T M F トーンを含めて、不変に、ただセキュリティデバイス 1 0 2 を通過する。しかしながら、電話 1 0 4 が、発行機関に対して呼出しを開始する 2 0 8 ときに、セキュリティサーバ 1 1 0 (例えば、セキュリティサーバ 1 1 0)は、セキュリティデバイスをウェイクアップする(wakes up)アクティブ化信号(activation signal)を送信する 2 1 0。アクティブ化信号は、セキュリティデバイス 1 0 2 が、一致によってトリガされる可能性が高くないことをかなり確実にするのに、十分に長く、かつ/または一意である(例えば、十分なディジットまたはシンボルを有する)ようにすることができる。一例においては、このアクティブ化信号は、どのような情報も実際には搬送することができず、非アクティブ(パッシブ)モードからアクティブモードへと切り換えるために、ただセキュリティデバイス 1 0 2 をトリガし、あるいはアクティブにするだけである。例えば、アクティブ化信号は、セキュリティデバイス 1 0 2 によって認識される短い一片の音楽またはトーンとすることができる。セキュリティデバイス 1 0 2 は、セキュリティサーバ 1 1 0 からのアクティブ化信号(例えば、D T M F トーンの一意的組)をリッスンし(listens)、認識し、そしてアクティブモードへと変化する 2 1 2。一例においては、アクティブ化信号を受信するとすぐに、セキュリティデバイス 1 0 2 は、電話からセキュリティサーバ 1 1 0 へのすべての D T M F トーンを暗号化することを開始することができる。

【 0 0 4 2 】

1 つのコンフィギュレーションにおいては、チャレンジ - レスポンススキーム(challen

ge-response scheme) が、セキュリティデバイス 102 と、セキュリティサーバ 110 との間でインプリメントされることができる。アクティブ化信号に加えて、セキュリティサーバ 110 は、ランダムなチャレンジをセキュリティデバイスに対して送信する 214 ことができる。セキュリティデバイス 102 は、チャレンジを受信し、応答（例えば、識別子、およびチャレンジに対するレスポンス）を生成し 216、そしてセキュリティサーバ 110 に対してその応答を送信する。応答は、セキュリティデバイス 102 に関連する識別子と、チャレンジに対するレスポンスとを含むことができる。セキュリティデバイス 102 は、電話 104 からセキュリティサーバ 110 への後続の DTMF トーンを暗号化するために使用されることができるセッションキー(session key)を生成することもできる。

【0043】

応答は、セキュリティサーバ 110 に、それが、関連するセキュリティデバイスと通信していること、を通知する。セキュリティサーバ 110 は、特定の顧客のアカウントをルックアップする(look-up) 220 ためにその識別子を使用することができ、それによって、顧客が彼ら自体を手動で識別するトラブルを不要にする(saving)（例えば、顧客に、彼らのアカウント番号を入力させることを避ける）。セキュリティサーバ 110 は、ユーザを認証するために、セキュリティデバイス 102 とセキュリティサーバ 110 との両方において用意される、認証キー(an authentication key)とランダムなチャレンジ(random challenge)とに基づいて、レスポンスが正しいことを検証する(verify) 222 こともできる。セキュリティデバイス 102 は、ユーザの電話に近接して（例えば、ユーザのホームの内側に）位置するので、攻撃者は、攻撃を開始するためにはそれを盗まなければならないであろうことに、留意が必要である。

【0044】

同じチャレンジとレスポンスとを使用することにより、セキュリティサーバ 110 は、セキュリティデバイス 102 が計算する 224 ときに、同じセッションキー(session key)を計算する 226。セキュリティサーバ 110 が、それがセキュリティデバイス 102 から受信する応答と意見を異にする（あるいはレスポンスを全く受信しない）場合、その呼出しは、より説得力のある識別および/または認証のために代替経路に迂回させられることができる。すなわち、セキュリティデバイス 102 は、受信されたランダムなチャレンジと認証キーとに基づいてそのレスポンスを計算することができる。次いで、セキュリティサーバ 110 は、ローカルなレスポンスを（ランダムなチャレンジと認証キーとに基づいて）計算することにより、受信されたレスポンスを検証し、そしてそれをセキュリティデバイス 102 からの受信されたレスポンスと比較することができる。

【0045】

チャレンジ - レスポンスが、適切に認証される場合、セキュリティサーバ 110 は、新しく導き出されたセッションキーを使用して、認証される確認(confirmation that is authenticated)を送信する 228。この確認は、電話 104 からやって来る DTMF トーンを暗号化することを開始するようにセキュリティデバイス 102 に通知する。セキュリティサーバからの確認に伴う問題が存在する（例えば、それが、ある種の最大時間内にセキュリティデバイス 102 によって受信されない、あるいは確認が、失敗する、などの）場合、セキュリティデバイス 102 は、ユーザに対する警告信号を生成することができる。例えば、チャレンジ - レスポンス認証が失敗する場合、ライトが、フラッシュする（または ON になる）ことができ、あるいはアラームが、鳴ることができる。さらに、ライト（例えば、発光ダイオード - LED）は、セキュリティデバイス 102 が、アクティブであり、かつ/またはチャレンジ - レスポンスが、正常に認証されることを示すように、輝くことができる。

【0046】

ひとたび、チャレンジ - レスポンスが、正常に認証された後に、一例においては、セッションキーは、電話 104 からセキュリティサーバへの送信を暗号化するためにセキュリティデバイス 102 によって使用されることができる。ひとたび暗号化が開始された後に

、セキュリティデバイスは、電話からやって来るDTMFトーンを遮断し232、そしてその代わりに暗号化されたDTMFトーンを送信する234。DTMFトーンのそのような暗号化の一例においては、電話104からのDTMFトーンは、異なるDTMFトーンへと変換されることができ、この異なるDTMFトーンは、次いでセキュリティサーバ110に対して送信される。別のコンフィギュレーションにおいては、DTMFトーンは、セキュリティデバイス102によってデジタルシンボルに変換されることができ、これらのデジタルシンボルは、次いで暗号化され、そしてセキュリティサーバ110に対して送信される。セキュリティデバイス102はまた、他のどんなもの（非DTMFトーンまたは信号）でもそれを修正すること、または暗号化することなしに、両方向に渡す。ユーザが要求されることができる第1の物事のうちの1つは、それらのアカウントに関連するPIN番号を入力することであるので、このPIN番号に関連するDTMFトーンは、暗号化され、そして認証のための第2のファクタを形成することができる。同様に、セキュリティサーバ110は、電話からセキュリティデバイスを経由して受信されるDTMFトーンを暗号解読する236のためにセッションキーを使用することができる。

【0047】

代替コンフィギュレーションにおいては、セキュリティデバイス102は、特定の発行機関108に関連する特定の電話番号（単数または複数）を認識するように構成されていることができる。セキュリティデバイス102が、電話が特定の電話番号をダイヤルしていることを認識するときに、そのセキュリティデバイスは、アクティブモードに自動的に切り換わり、かつ/または電話からセキュリティサーバ110へのすべてのDTMFトーンを暗号化することができる。

【0048】

セキュリティデバイス102は、呼出しが終了されるまで電話104からのDTMFトーンを暗号化し続けることができ、このポイントにおいて、セキュリティデバイス102は、非アクティブモードに切り換わって戻り、ここでセキュリティデバイスは、すべてのDTMFトーンが、不変に通過することを可能にする。

【0049】

セキュリティデバイス102は、小型フォームファクタを有することができるので、それは、電話回線に容易にプラグインされることができる。ユーザは、ユーザが、異なるロケーション（例えば、ホーム、オフィスなど）からアカウントにセキュリティ上安全にアクセスすることができるようにするために、単一の機関またはアカウントに関連する複数のセキュリティデバイスを有することができる。ユーザはまた、様々な異なる機関および/またはアカウントに関連する複数のセキュリティデバイスを有することもできる。これらの複数のセキュリティデバイスは、電話回線に沿って直列に結合されることができる。チェーンの中の非アクティブなセキュリティデバイスは、ただそのチェーンの中の次のセキュリティデバイスに信号を渡すだけである。チェーンの中のセキュリティデバイスが、セキュリティサーバによってアクティブにされる場合、そのときには、それは、電話からのDTMFトーンを暗号化する。

【0050】

別の例においては、セキュリティデバイス102は、1つの電話またはロケーションから、複数のユーザにサブする(serve)ことができる。そのような場合には、セキュリティサーバは、セキュリティデバイスが、複数のユーザまたはアカウントに関連することを識別することができる。各ユーザの間で区別するために、セキュリティサーバは、特定のユーザまたはアカウントを識別するPINまたは他の識別子を入力するようにユーザに要求する音声プロンプト(voice prompt)を送信することができる。

【0051】

図3は、送信中にDTMFトーンをセキュリティ保護することを可能にすることができるテレサービスセキュリティサーバの一例のブロック図を示している。セキュリティサーバ302は、小型および/または低パワーのマイクロプロセッサなどの処理回路(processing circuit)304を含むことができる。セキュリティサーバ302は、セキュリティサ

サーバ 302 を通信ネットワークに結合するために使用される第 1 の通信モジュール 306 を含むことができる。認証モジュール(authentication module) 308 は、セキュリティサーバ 302 が、それが通信するセキュリティデバイスを認証することを可能にする。DTMF 暗号解読モジュール(DTMF decryption module) 308 は、セキュリティサーバ 302 が、セキュリティデバイスから受信される暗号化された DTMF トーンを暗号解読する(decrypt encrypted DTMF tones)ことを可能にする。

【0052】

図 4 は、電話デバイスからの DTMF トーンをセキュリティ保護するための、セキュリティサーバ上で動作する一方法を示している。呼出しが、DTMF がイネーブルにされた電話(DTMF-enabled telephone)から受信される 402。アクティブ化信号が、DTMF がイネーブルにされた電話に関連するセキュリティデバイスに対して送信される 404。セキュリティデバイスは、DTMF がイネーブルにされた電話に近接して近くに位置することができる。次いで、セキュリティデバイスは、セキュリティサーバによって認証される。例えば、チャレンジ信号が、セキュリティデバイスに対して送信される 406。セキュリティサーバは、レスポンスが、セキュリティデバイスから受信されるかどうか 408 を決定する。受信されない場合、そのときには、セキュリティデバイスが、電話回線(telephone line)上に存在していない 410 ことが仮定されることができる。そうでない場合には、セキュリティサーバは、受信されたレスポンスが、正常に認証される(successfully authenticated)ことができるのかどうか 412 を決定する。受信されたレスポンスが、正常に認証されることができない場合、そのときには認証は、失敗する 414。そうでない場合には、セッションキーが、生成される 416。セッションキーによって認証される確認メッセージが、セキュリティデバイスに送信される 418。セキュリティサーバは、セキュリティデバイスからの暗号化された DTMF トーンを受信する 420 ことができる。次いで、セキュリティサーバは、電話によって送信される情報を得るために受信された DTMF トーンを暗号解読する 422 ことができる。そのような DTMF トーンは、元の DTMF トーンを暗号化することにより送信中に盗聴者から保護される機密情報(例えば、アカウント番号、パスワード、PIN など)を表すことができる。

【0053】

図 5 は、送信中に DTMF トーンを保護するように構成されていることができるセキュリティデバイスの一例のブロック図を示している。セキュリティデバイス 502 は、小型および/または低パワーのマイクロプロセッサなどの処理回路 504 を含むことができる。セキュリティデバイス 502 は、それが結合される電話回線によって電力を供給されることができる。第 1 の通信インターフェース A 506 は、セキュリティデバイス 502 を電話に結合するために使用されることができる。第 2 の通信インターフェース B 508 は、セキュリティデバイス 502 を通信ネットワークに結合するために使用されることができる。パッシブモードのオペレーションにおいては、セキュリティデバイス 502 は、すべての DTMF トーンを不変に通過させる。処理回路 504 は、(例えば、セキュリティサーバからの)アクティブ化信号をリッスンするように構成されていることができる。DTMF 検出器 510 は、セキュリティデバイスをアクティブモードのオペレーションに切り換える DTMF アクティブ化信号を検出するように構成されていることができる。アクティブモードにおいては、セキュリティデバイス 502 は、セキュリティサーバからの認証チャレンジにレスポンスする(respond)ように構成されていることができる。

【0054】

アクティブ化モードにおいては、DTMF 検出器 510 はまた、通信インターフェース A 506 を経由して受信される(例えば、電話からやって来る)DTMF トーンを検出するように構成されていることもできる。1 つまたは複数の DTMF トーンが、検出される場合、DTMF トーンは、暗号化され、あるいはそうでなければ DTMF 暗号化モジュール(DTMF encryption module) 512 によって修正される。次いで、暗号化された DTMF トーンは、通信インターフェース B 508 を通してセキュリティサーバに対して送信される。

【 0 0 5 5 】

図 6 は、電話デバイスからの D T M F トーンをセキュリティ保護するための、セキュリティデバイス上で動作する一方法を示している。セキュリティデバイスは、呼出しが、電話とセキュリティサーバとの間で開始されるとすぐに電力を供給される 6 0 2。すなわち、呼出しが行われるときに通信回線はエネルギーが供給されるので、セキュリティデバイスは、通信回線からその電力を引き出すことができる。パッシブモード(passive mode)のオペレーションにおいて、セキュリティデバイスは、D T M F トーンが、第 1 の通信インターフェースと、第 2 の通信インターフェースとの間を不変に通過する(pass through unchanged)ことを可能にする 6 0 4。例えば、第 1 の通信インターフェースは、電話に結合されることができ、そして第 2 の通信インターフェースは、第 2 の通信インターフェースに結合されることができ。セキュリティデバイスは、(D T M F) アクティブ化信号が、セキュリティサーバから受信されるかどうか 6 0 6 を決定するように送信を監視する。セキュリティデバイスは、アクティブ化信号が受信されない限り、パッシブモードで動作することを継続する。D T M F アクティブ化信号が、受信される場合、セキュリティデバイスは、アクティブモードのオペレーションへと変化する 6 0 8。セキュリティデバイスはまた、セキュリティサーバからの他の信号をリッスンする 6 1 0 こともできる。

【 0 0 5 6 】

セキュリティデバイスは、セキュリティサーバからのチャレンジを受信する 6 1 2 ことができる。セキュリティデバイスは、そのチャレンジに対してレスポンスで応答する 6 1 4。そのレスポンスが、有効である場合、セキュリティデバイスは、セキュリティサーバが、セキュリティデバイスを正常に認証していることを示す確認を受信する 6 1 6 ことができる。

【 0 0 5 7 】

ひとたびアクティブ化され、そして適切に認証された後に、セキュリティデバイスは、電話からの D T M F トーンをリッスンする。D T M F トーンが、(セキュリティデバイスが結合される) 電話から第 1 の通信インターフェース上で受信される 6 1 8 場合、受信された D T M F トーンは、異なる D T M F トーンへと暗号化される 6 2 0。一例においては、電話からの D T M F トーンは、異なる D T M F トーンへと変換されることができ、この異なる D T M F トーンは、次いでセキュリティサーバへと送信される。別のコンフィギュレーションにおいては、D T M F トーンは、セキュリティデバイス 1 0 2 によってデジタルシンボルへと変換されることができ、このデジタルシンボルは、次いで暗号化され、そしてセキュリティサーバに対して送信される。次いで、暗号化された D T M F トーンは、第 2 の通信インターフェース上でセキュリティサーバへと送信される 6 2 2。セキュリティデバイスは、呼出しが終了するまで、電話からの D T M F トーンを暗号化し続け、その時に、セキュリティデバイスは、パッシブモードに戻る 6 2 4。セキュリティデバイス 1 0 2 は、電話からの暗号化されていない D T M F トーンが、セキュリティサーバに対して渡らないようにする。一例においては、セキュリティデバイス 1 0 2 は、ネットワークがアクティブである間にすべての入力(例えば、送信)を電話から切り離すことができる。この場合においては、例えば、顧客が、代理人(representative)と話す必要がある場合、顧客またはセキュリティサーバのいずれかが、入力を再接続する(例えば、セキュリティデバイス 1 0 2 からの送信を可能にする)ための何らかの用意が存在していてもよい。

【 0 0 5 8 】

セルラ電話のセキュリティスキーム(Cellular Phone Security Scheme)

図 7 は、セキュリティサーバを用いてそれ自体を認証するように構成されたモバイル通信デバイスのブロック図である。モバイル通信デバイス 7 0 2 は、通信モジュール 7 0 6 と、ユーザ入力インターフェース 7 0 8 とに結合される処理回路 7 0 4 を含んでいる。通信モジュール 7 0 6 は、モバイル通信デバイス 7 0 2 が、ワイヤレス通信ネットワーク 7 1 0 上で通信することを可能にする。処理回路 7 0 4 は、呼出し中に 1 つまたは複数のセキュリティサーバを用いてそれ自体を認証するように構成されることができ。例えば、モバイル通信デバイスは、銀行または金融機関が、モバイル通信デバイス 7 0 2 のユ

ーザを認証することを可能にする認証キーおよび/またはユーザ識別子を用いて構成されていることができる。認証キーおよび/またはユーザ識別子は、銀行または金融機関によって事前に（例えば、セットアップまたはコンフィギュレーション中に）提供されることができる。さらに、処理回路 704 はまた、認証プロセスを完了するためにユーザからの P I N、パスワード、および/または他の入力を要求することもできる。

【0059】

図 8 は、通信ネットワーク上でテレサービス局 804 に対してモバイル通信デバイス 802 を認証するための一方法を示す流れ図である。モバイル通信デバイス 802 は、モバイル電話とすることができ、そしてテレサービス局 804 は、銀行または金融機関に関連づけられたセキュリティサーバを含むことができる。モバイル通信デバイス 802 とテレサービス局 804 (tele-services station) とは、おのこの同じ認証キーを有することができる。

【0060】

モバイル通信デバイスは、テレサービス局に関連する発行機関に対して呼出しを開始する 806 ことができる。発行機関は、例えば、銀行または金融機関とすることができ、テレサービス局は、モバイル通信デバイスに対してランダムな認証チャレンジ(random authentication challenge) 808 を送信する。次いで、モバイル通信デバイスは、ランダムなチャレンジと、認証キーとに基づいてレスポンスを生成し 809、そしてレスポンスと、（もしかすると）ユーザ識別子とをテレサービス局に対して送信する 810。次いで、テレサービス局は、モバイル通信デバイスからのレスポンスが、正しいかどうかを検証する 812。その認証キーと、ランダムな認証チャレンジとに基づいて検証値を計算し、そしてそれをモバイル通信デバイスから受信されるレスポンスと比較するテレサービス局によって行われることができる。レスポンスが、正常に認証される場合、認証確認(authentication confirmation) 814 は、モバイル通信デバイスに対して送信されることができる。モバイル通信デバイスは、テレサービス局からの慎重な扱いを要する情報(sensitive information)（例えば、銀行アカウント記録など）を要求する 816 ことができる。モバイル通信デバイスが、正常に認証される場合、そのときには、テレサービス局は、要求された、慎重な扱いを要する情報をモバイル通信デバイスに対して供給する 818。このようにして、モバイル通信デバイス（例えば、モバイル電話）は、呼出し中に慎重な扱いを要する情報の送信をセキュリティ保護するために、テレサービス局によって認証されることができる。

【0061】

脅威モデル(Threat Models)

ここにおいて説明されるセキュリティデバイスおよび/または方法によって対処される 1 タイプの脅威(threat)は、盗聴攻撃である。そのような攻撃においては、攻撃者は、電話上でユーザによって入力される番号に関連する D T M F トーンをリッスンするために、電話線に対してレコーダを攻撃することができる。これらの D T M F トーンは、個人および/または機密の情報のうちでもとりわけ、呼び出されている銀行、ユーザの顧客番号および/またはアカウント番号、個人識別番号(P I N)、社会保障番号を、識別することができる。次いで、攻撃者は、ユーザのアカウントからの不正なトランザクションを実行するためにこの情報を使用することができる。ここにおいて説明されるセキュリティデバイスは、D T M F トーンを暗号化すること、およびさらなる認証を提供することにより、そのような攻撃を打ち負かす(defeat)。ほとんどの機関（例えば、銀行など）は、認証についての 2 つのファクタ（例えば、セキュリティデバイスの所有と、P I N の知識と）を使用することができるので、他の慎重な扱いを要する情報を要求する必要があることは、まれであることになる。暗号化された D T M F トーンを単に遮断することは、対応するアカウント番号、P I N などについて何も明らかにしない。

【0062】

攻撃者は、成功するためには、例えば、呼出しが、対象とする(received)レシーバ（例えば、対象とする銀行）へと進まないようにすることにより、呼出しの進行に干渉し、呼

出し者にすべての慎重な扱いを要する情報を入力するように要求して、対象とする(intending)レシーバのふりをする必要があることになる。そのような攻撃を打ち負かすためには、セキュリティデバイスは、「暗号化開始(start encrypting)」信号(すなわち、認証された確認)が、受信機関から受信された後に、セキュリティインジケータ(例えば、ライト)をオン(On)にすることができる。呼出し者(例えば、顧客)は、セキュリティデバイスが、任意の慎重な扱いを要する情報または機密情報を入力する前にそのトーンを暗号化していることを確認するために、ただセキュリティインジケータを検査するだけである。

【0063】

別のタイプの攻撃は、セッションハイジャック攻撃(session hijacking attack)とすることができ、ここでは、攻撃者は、ユーザが、対象とするレシーバ(例えば、銀行)との通信を確立するまで待ち、そのようにして、セキュリティインジケータをアクティブにし、そしてその後に呼出しを乗っ取る。次いで、攻撃者は、呼出しと共に何かが間違ったようなふりをし、そしてユーザに慎重な扱いを要する情報を口頭で提供するように要求することができる。代わりに、攻撃者は、ユーザに、トーン毎の暗号パターンを確立することを試みるために(攻撃者に既に知られている)特定のレスポンスを入力するように要求し、そしてその後に、銀行に対する彼ら自体のレスポンスを暗号化するためにトーン毎の会話を使用することができる。このタイプの攻撃に対処するために、トーン毎の暗号化は、擬似ランダムベース、回転ベース、および/または番号からトーンの関係の発見を禁止する他のベース、に基づいて変更され、あるいは修正されることができる。

【0064】

メッセージおよびセッションの認証(Message and Session Authentication)

セキュリティデバイスは、例えば、メッセージ認証コード(Message Authentication Code)(MAC)ファンクションを使用することにより、メッセージ認証とセッションキー導出とを実行するように構成されていることができる。例えば、セキュリティサーバは、 MAC_K の単一の呼出し(チャレンジ)から出力を分離することにより、呼出し者のセキュリティデバイスを認証することができる。例えば、典型的なMACファンクションは、128ビットの出力を戻すことができ、これは、32個のDTMFトーンとして表されることができる。セキュリティサーバと、セキュリティデバイスとが、MACを計算した後に、セキュリティサーバは、(MACの一部分を表す)第1の16個のDTMFトーンをセキュリティデバイスに対して送信することができ、そして、レスポンスの形で、セキュリティデバイスは、(MACの他の一部分を表す)他の16個のDTMFトーンを返信する。このようにして、セキュリティサーバと、セキュリティデバイスとの両方は、それらが、認可され、あるいは合法的であることを互いに証明することができる。

【0065】

同様に、セッションキー(session key)は、 $Session\ Key = MAC_K(Authentication\ Key || Challenge)$ であるように、おの側の側について(by each side)計算されることができ、ここで認証キー(authentication key)は、セキュリティデバイスへとあらかじめロードされる。セキュリティデバイスが、そのレスポンスをセキュリティサーバに対して送信するときに、セッションキーが、開示されないようにするために、レスポンスは、追加の情報を含んでもよい。例えば、レスポンス(response)は、 $Response = MAC_K(「余分な情報ストリング(extra information string)」 || Authentication\ Key || Challenge)$ とすることができる。

【0066】

ストリーム暗号化(Stream Encryption)

別の特徴は、暗号化されたシンボルのセキュリティを保護する効率的な暗号化方法を提供する。各平文シンボルは、別個の擬似ランダムに選択された変換テーブルを使用することにより暗号化される。シンボルのあらゆる可能な置換をあらかじめ記憶しておくのではなくて、変換テーブルは、擬似乱数と、シンボルシャッフリングアルゴリズムとに基づい

てオンザフライで効率的に生成されることができる。受信デバイスは、同様に、受信された暗号化シンボルを暗号解読するために逆変換テーブルをオンザフライで生成することができる。

【0067】

この暗号化方法は、様々なコンフィギュレーションでインプリメントされることができる。例えば、電話セキュリティデバイスは、DTMFトーンをデジタル値へと変換し、各デジタル値について擬似ランダムに選択された変換テーブルを使用することにより、そのデジタル値を暗号化することができる。次いで、暗号化されたデジタル値は、デジタル形式で、または暗号化されたデジタル値に関連するDTMFトーンとしてのいずれかで、セキュリティサーバ（例えば、テレサービス局）へと送信されることができる。

【0068】

DTMFトーンは、デジタルシンボルによって表され（あるいは、それに関連づけられる）るので、それらは、例えばストリーム暗号化によってセキュリティ保護されることができる。様々な例においては、ストリーム暗号化は、カウンターモード、出力フィードバック(Output Feedback) (OFB) モード、または暗号文フィードバック(Ciphertext Feedback) (CFB) モードの高度暗号化規格(Advanced Encryption Standard) (AES) などのブロック暗号(block cipher)によって生成されるキーストリームを使用することができる。例えば、MACファンクションは、CBC-MACモードのブロック暗号を用いてインプリメントされることができる。これは、例えば、セキュリティデバイスが、AESをハードウェアでインプリメントしている場合に、有利とすることができる。

【0069】

これらのファンクションが、ソフトウェアでインプリメントされる場合、非線形SOBER(Non-linear SOBER) (NLS) などの専用化ストリーム暗号を使用することが好ましい可能性がある。ストリーム暗号はまた、キーまたは臨時の入力として暗号化されるべきデータを使用すること、次いで出力キーストリームを生成することにより、低効率ではあるが、MACファンクションとして使用されることもできる。生成されるキーストリームの長さは、できるだけ望ましいものとするので、レスポンスと、セッションキーとの両方は、単一の呼出しの中で生成されることができる。

【0070】

（真のストリーム暗号を使用しようと、ストリーミングモードのブロック暗号を使用しようと）従来のストリーム暗号化は、通常、擬似乱数のキーストリームを生成することと、暗号化された出力または暗号文を形成するためにそれらを平文（すなわち、DTMFトーンのデジタル表現）と組み合わせることと、により進行する。普通は、キーストリームと平文とは、それが自己反転であるので、排他的論理和(XOR)演算を使用して組み合わせられる。しかしながら、従来のDTMFがイネーブルにされた電話は、各キーが固有のトーンを有する10個以上のキーを有する。それ故に、XOR演算は、キーストリームを用いて前記DTMFトーンを暗号化するために使用されることができない。その代わりに、電話キーに関連するDTMFトーンは、暗号化されたシンボルまたは暗号文を生成するためにキーストリームから得られる擬似ランダムな数/シンボルに追加されることができる異なるデジタルシンボルへと変換される（またはそれに関連づけられる）ことができる。しかし、特定のディジットの位置を知っているアクティブな攻撃者は、送信された暗号文から数を差し引くことによりその数を変更することができる。例えば、特定のDTMFトーンの場合に、入力が、「1」であったが、出力は「7」であったことを知っている攻撃者は、このトーンについて生成された擬似乱数は、「6」であったことを決定することができ、そしてそのときには、特定のディジット位置についての彼らの選択した任意のキャラクタを正しく暗号化することができる。

【0071】

組み合わせコンバイナ(Combinational Combiner)

1つの特徴は、暗号化されるべき各平文シンボルについて擬似ランダムに選択され、あるいは生成された変換テーブルを得るために、または生成するためにキーストリームを使

用することを提供する。キーストリームから擬似乱数を取ることと、平文を同じように（例えば、モジュロ n を加算することにより）変更することとの代わりに、1つの特徴は、複数の変換テーブルのうちの1つを擬似ランダムに選択することにより、各平文シンボルを入力ストリームに変換することを提供する。変換テーブルは、数またはシンボルの組の異なる可能な置換を提供することができる。これは、ここにおいて組み合わせコンバイナと称される。

【0072】

図9は、暗号化されるべき各シンボルについて変換テーブルを擬似ランダムに選択することにより、平文シンボルをセキュリティ保護するための組み合わせコンバイナのブロック図を示している。暗号ジェネレータ902は、擬似ランダムな数/シンボル(pseudorandom numbers/symbols)のキーストリーム(keystream) S_i 904を生成するために使用される。キーストリーム904の擬似乱数は、入力ストリームの中の各平文入力シンボル(plaintext input symbol) P_i 908について、複数の可能な変換テーブルから異なる変換テーブル(different translation table) 906を生成し、または得るために使用される。平文入力シンボル908を擬似ランダムな出力へと変換することにより、暗号化された出力シンボル(encrypted output symbol) C_i 910が、生成される。

【0073】

そのような変換オペレーションは、キーストリーム904の制御の下で平文入力シンボル908の置換を定義する。変換テーブル906は、 n 個の要素のベクトルとして表されることができ、そして平文入力シンボル908の変換は、変換テーブル906の p 番目の要素をルックアップすることにより行われることができる。暗号化された出力シンボル C_i を仮定すると、逆変換は、逆の置換のテーブルを作成することにより、あるいはシンボル C_i を含む入力求めてテーブルをサーチすること、およびそのインデックスを p として戻すことにより、そのいずれかで行われることができる。

【0074】

一般的に言って、 n 個の平文シンボルの組では、 $n!$ （階乗）の可能な置換が、存在する。置換は、すべてのそのような置換の組からランダムに選択され、そして平文入力シンボル P_i 908を暗号化された出力シンボル C_i 910（暗号文とも称される）に変換するために変換テーブル906として使用されることができ、入力ストリームの中の各平文シンボルについて、擬似ランダムに選択された変換テーブルが、選択される。そのときには、暗号化されたシンボル C_i 910を調べ、そしてそれが、特定の平文シンボルに対応することを知っている攻撃者は、他の平文シンボルと、対応する暗号化されたシンボルとの間の対応について依然として何も知らないままである。すなわち、攻撃者が、確認することができるすべての情報は、暗号化されたシンボルを変更することは、彼らが知っているものとは異なる平文シンボルを与えることになるが、そのようになる他の平文シンボルではないことである。それ故に、擬似ランダムに選択された変換テーブルは、平文入力シンボルと、暗号化された出力シンボル（暗号文）との間の関係を明らかにせず、そして攻撃者は、暗号文シンボル変換に対するどのような単一の平文シンボルの知識も活用することができない。

【0075】

セキュリティ保護された電話バンキングについての一例においては、電話からセキュリティデバイスによって受信される各DTMFトーンは、デジタル平文シンボルへと変換され（またはそれに関連づけられ）る。次いで、平文シンボルは、暗号化されたシンボルを得るために（キーストリームからの1つまたは複数の擬似乱数に基づいて得られる）変換テーブルによって変換される。次いで、暗号化されたシンボルは、（デジタル形式で、または暗号化されたシンボルに対応するDTMFトーンとしてのいずれかで）セキュリティサーバに送信され、ここでそれは、逆変換テーブルによって暗号解読される。逆変換テーブルは、同じキーストリームを生成するセキュリティデバイスと、セキュリティサーバとの両方において同期化された暗号ジェネレータを有することにより、生成され、または得られることができる。一例においては、暗号ジェネレータは、同じシード(seed)（例えば

、セッションキーなど)を使用することにより同期化されることができる。

【0076】

一例においては、複数の変換テーブルは、あらかじめ生成され、かつ/またはセキュリティデバイスおよび/またはセキュリティサーバによって記憶されることができる。オンザフライで新しい変換テーブル(すなわち、入力シンボルの置換)を生成することではなくて、変換テーブルは、あらかじめ生成され、そして記憶されることができる。キーストリーム904の擬似ランダムな数/シンボルは、暗号化されるべき各平文シンボルについてのあらかじめ生成された変換テーブルのうちの1つを選択するために使用されることができる。あらかじめ生成された変換テーブルは、 n 個の平文シンボルの組についてのあらゆる置換、または置換のサブセットを定義することができる。

【0077】

別の例においては、使用される変換テーブルは、変換テーブルを形成するためにキーストリームを使用することと、擬似ランダムにシンボルをシャッフルすることにより、オンザフライで生成されることができる。これらのソリューションは、 $n!$ 個のテーブルが存在することになるという意味で同等であり、そして、これらのテーブルのうちの1つを選択するために必要とされるキーストリームの量は、シャッフルすることによりそのようなテーブルを作成するために必要とされる量と同じであることに注意すべきである。

【0078】

図10は、平文シンボル(plaintext symbol)を暗号化されたシンボル(encrypted symbol)に変換するための、シンボルごとの変換テーブル(a symbol-to-symbol translation table)1002の一例を示している。この例においては、16個の平文シンボルが、異なる暗号化されたシンボルへと変換される。2進表現は、この例においては、16個の平文シンボルが、4-ビットの暗号化されたシンボルを使用して、暗号化されることができることを単に示すために示される。より多くの、あるいはより少ない数の平文シンボルが暗号化されるべきであるという他の例においては、異なる数のビットが、各シンボルについて使用されることができる。例えば、256個までの平文シンボルの場合には、8ビットが、各暗号化シンボルを生成するためにキーストリームから抽出されることができる。

【0079】

別の特徴は、特定の変換テーブル内で、平文シンボルと、暗号化されたシンボルとの間の1対1の対応を提供する。すなわち、どの2つの平文シンボルも、特定の変換テーブル内で、同じ暗号化されたシンボルには変換されない。これにより、暗号解読デバイスは、暗号化されたシンボルをその元の平文シンボルへと正確に暗号解読することができるようになる。

【0080】

暗号解読デバイスにおいては、シンボルごとの逆変換テーブル(a symbol-to-symbol reverse translation table)は、暗号化デバイスのシンボルごとの変換(symbol-to-symbol translation)を逆にし、そしてそれによって受信された暗号化シンボルを暗号解読するように生成されることができる。

【0081】

図11は、どのようにして平文シンボル1102が、暗号化されたシンボル1106を得るために異なる変換テーブル1104を使用して暗号化されることができるのかの一例を示している。各平文シンボル P_0 、 P_1 、 P_2 、 P_3 、... P_i について、おのおのシンボルの異なる置換を有する異なる変換テーブル1104が、暗号化されたシンボル C_0 、 C_1 、 C_2 、 C_3 、... C_i を得るために使用される。

【0082】

組の中のシンボルの数が少ない場合には、そのようなシンボルのすべての置換をリストアップし(すなわち、あらかじめ生成し)、そしてそれらの置換から変換テーブルを選択するために(キーストリームからの)インデックスを使用することを、可能とすることができる。例えば、12個の可能なシンボルの組の場合には、生成される可能な置換の数は、 $12!$ 、すなわち479,001,600である。置換を適切に選択するためには、3

2 ビットのキーストリームは、バイアスのない変換テーブルとして1つの置換を選択することで十分とすることができる。しかしながら、このアプローチは、組の中のシンボルの数が、増大するにつれて非効率的になる。例えば、256個の可能なシンボルの組の場合には、生成される可能な置換の数は、 $256!$ 、すなわち 8.5×10^{506} であり、これは、変換テーブルとして置換のうちの1つを選択するために、擬似ランダムなキーストリームから1684ビットを超過して必要とすることになる。

【0083】

図12は、1組の n 個のシンボルについての複数の可能な置換から変換テーブルを選択するためのアルゴリズムを示しており、ここで n は、正の整数である。この例においては、0と、 $2^k - 1$ との範囲に一樣に分散された k ビットの長さ（例えば、8ビット、32ビットなど）の擬似ランダムなキーストリーム値(pseudorandom keystream values)を与える暗号ジェネレータが、使用されることができる。キーストリームは、擬似乱数 w を得るために使用される1202。 $n!$ は、 2^k に均一に分割することができないので、擬似乱数 w は、バイアスを導入することなく直接使用されることができない。この理由のために、最大のしきい値 P_{max} (maximum threshold P_{max})は、 2^k よりも小さい $n!$ の最大の倍数(the largest multiple of $n!$ that is less than 2^k)として定義される。もし擬似乱数 w がこの最大のしきい値 P_{max} よりも小さい場合は、そのときには、それは、バイアスを導入することなく使用されることができる。そうではなくて、もし擬似乱数 w がこの最大のしきい値 P_{max} 以上である場合は、それは、捨てられ、そして新しい擬似乱数 w が、最大のしきい値 P_{max} よりも小さい擬似乱数 w が得られるまで、選択される1204。

【0084】

$w = w \bmod n!$ であるように、擬似乱数 w は($n!$)を法として割られる(the pseudorandom number w is divided modulo $n!$ so that $w = w \bmod n!$)1206。それ故に、バイアスされない擬似乱数 w は、置換（すなわち、変換テーブル）を得るために使用されることができる0から $n!$ の範囲の中で得られる。

【0085】

あらかじめ生成された置換を記憶することと、擬似乱数 w を使用することにより1つのそのような置換を選択することとではなくて、1つの特徴は、変換テーブルを生成するために基本の置換(base permutation)のシンボルをシャッフルすることにより、置換を生成することを提供する。基本の置換ベクトル P は、 $P = [0, 1, 2, \dots, n-1]$ となるようにシンボルの組のすべての値で初期化される1208。次いで、シンボルシャッフリングアルゴリズム1210は、擬似乱数 w を使用して、基本の置換ベクトル P の中でシンボルをシャッフルするために使用される。

【0086】

シンボルシャッフリングアルゴリズム1210の一例は、カウンタ i から $n-1$ を初期化し、ここで n は、組の中のシンボルの数である。カウンタ $i \geq 0$ である間に、擬似乱数 $w = w / (i+1)$ であり、変数 $j = w \bmod (i+1)$ であり、そして置換ベクトル P の値は、 $P_t[i] = P_{t-1}[j]$ であり、そして $P_t[j] = P_{t-1}[i]$ であるようにシャッフルされる。他のシンボルシャッフリングアルゴリズムも、本開示の特徴を逸脱することなく使用されることができることに注意すべきである。

【0087】

ひとたび、置換ベクトル P が、シャッフルされた後に、その置換ベクトル P は、入力シンボルストリームを暗号化するために、例えば、変換テーブルとしてそれを使用することができる任意のアプリケーションに対して提供される1212ことができる。

【0088】

図13は、単一の平文シンボルを暗号化するために複数の変換テーブルを使用することにより、シンボル認証を達成することができる別の暗号化スキームを示すブロック図である。すなわち、平文入力シンボル P_{i1302} は、第1の暗号化された出力シンボル C_{i1310} を得るために、第1の暗号ジェネレータ1308から得られる第1のキースト

リーム $S_i' 1306$ に基づいて生成され、または選択されることができる変換テーブル $A_1 1304$ によって暗号化される。次いで、第1の暗号化された出力シンボル $C_i' 1310$ は、第2の暗号化された出力シンボル $C_i 1318$ を得るために使用される、第2の暗号ジェネレータ 1316 から得られる第2のキーストリーム $S_i 1314$ に基づいて生成され、または選択されることができる第2の変換テーブル $A_2 1312$ に対する入力としての役割を果たす。このようにして、リダンダンシーが、第1の暗号化された出力シンボル $C_i' 1310$ を認証するために使用されることができる。すなわち、シンボル $C_i' 1310$ と $C_i 1318$ とを一緒に使用することにより、シンボル $C_i 1318$ は、 $C_i' 1310$ を認証する。それ故に、攻撃者が、例えば、シンボル $C_i' 1310$ を変更する際に成功する場合には、それは、シンボル $C_i 1318$ によって適切に認証されないことになる。

【0089】

図14は、どのようにして複数の変換テーブル 1404 および 1406 が、対応する1対の暗号化されたシンボル 1408 を得るために各平文シンボル 1402 を暗号化するために使用されることができるのかを示している。変換テーブル 1404 および 1406 は、各平文シンボル P_i を1対のシンボル C_i' / C_i へと暗号化するために擬似ランダムに選択され、かつ/または生成されることができることに注意すべきである。

【0090】

図15は、どのようにして2つの変換テーブルが、平文シンボル P_n を1対のシンボル C_n' および C_n へと変換し、または暗号化するために使用されることができるのかの一例を示している。例えば、第1の平文シンボル $P_n = 「5」$ の場合には、第1の変換テーブル $A_1 1502$ は、第1の出力シンボル $C_n' = 8$ を提供する（すなわち、「5」が、「8」に変換される）。第1の出力シンボル「8」は、そのときには、第2の出力シンボル $C_n = 7$ を得るために、第2の変換テーブル $A_2 1504$ に対する入力としての役割を果たすことができる（すなわち、「8」が、「7」に変換される）。第2の出力シンボル C_n は、第1の出力シンボル C_n' に基づいて生成されたので、冗長なシンボル C_n および C_n' は、認証のために使用されることができる。いずれかまたは両方のシンボルが、送信中に攻撃者によって変更される場合、そのときには認証は、失敗する。例えば、 C_n' が、「8」から「4」へと攻撃者によって修正される場合、シンボル C_n' および $C_n = 「47」$ を受信する受信者は、 $C_n = 「7」$ が、 $C_n' = 「8」$ であり、「4」ではないことを意味すべきであることを見出すことになる。

【0091】

第2の平文シンボル $P(n+1)$ は、第1の平文シンボルと、第2の平文シンボルとが、同じである場合でさえ、完全に異なる変換テーブルを有することができる。例えば、第2の平文シンボル $P(n+1) = 「5」$ の場合には、第1の変換テーブル $B_1 1506$ は、第3の出力シンボル $C(n+1)' = 「*」$ を提供する（すなわち、「5」は、「*」に変換される）。第3の出力シンボル $C(n+1)' = 「*」$ は、そのときには第4の出力シンボル $C(n+1) = 「1」$ を得るために第2の変換テーブル $B_2 1508$ に対する入力としての役割を果たすことができる（すなわち、「*」は、「1」に変換される）。上記のように、シンボル対 $C(n+1)'$ と $C(n+1)$ との冗長な使用は、認証の一形式としての役割を果たすことができる。

【0092】

図16は、一例による、平文暗号化を実行するための一方法を示している。1組の n 個のシンボル内で定義される複数の入力シンボルが、得られる 1602 。異なるシンボルごとの置換を定義する複数の変換テーブルからの、擬似ランダムに選択された変換テーブルが、暗号化されるべき入力シンボルのおのおのについて得られる 1604 。入力シンボルは、各入力シンボルを個別に暗号化するために入力シンボルのおのおのについてそれらの対応する変換テーブルを使用して対応する出力シンボルへと変換される 1606 。次いで、出力シンボルは、暗号解読デバイスに対して送信される 1608 ことができる。

【0093】

そのような方法の一例において、第 1 の平文シンボルが得られ、ここで第 1 の平文シンボルは、1 組の中の n 個のシンボルのうちの 1 つとすることができる。 n 個のシンボルを n 個のシンボルの異なる置換へと変換する第 1 の変換テーブルが、得られる。第 1 の変換テーブルは、 n 個のシンボルをシャッフルするために擬似乱数を使用することにより擬似ランダムに生成されることができる。次いで、第 1 の平文シンボルは、第 1 の変換テーブルを使用して第 1 の出力シンボルへと変換される。

【0094】

n 個のシンボルを第 1 の変換テーブル以外の n 個のシンボルの異なる置換へと変換する第 2 の変換テーブルが、得られることができる。第 1 の出力シンボルは、第 2 の変換テーブルを使用して第 2 の出力シンボルへと変換される。次いで、暗号化されたシンボルが、第 1 および / または第 2 の出力シンボルに基づいて送信される。

【0095】

図 17 は、どのようにして暗号化されたシンボル C_i が、単一の平文シンボルを得るために 1 つまたは複数の逆変換テーブルを使用することにより暗号解読されることができるのかを示すブロック図である。すなわち、暗号化された入力シンボル C_i 1702 は、第 1 の暗号解読された出力シンボル C_i' 1710 を得るために、第 1 の暗号ジェネレータ 1708 から得られる第 1 のキーストリーム S_i' 1706 に基づいて生成され、または選択されることができる第 1 の逆変換テーブル A_1 1704 によって暗号解読されることができる。次いで、第 1 の暗号解読された出力シンボル C_i' 1710 は、平文出力シンボル P_i 1718 を得るために使用される、第 2 の暗号ジェネレータ 1716 から得られる第 2 のキーストリーム S_i 1714 に基づいて生成され、または選択されることができる第 2 の逆変換テーブル A_2 1712 に対する入力としての役割を果たす。

【0096】

例えば、 $C_i = (x, y)$ である代替コンフィギュレーションにおいては、暗号化されたシンボルおよび y は、平文出力シンボル P_i を得るために暗号化されたのと逆の順序で暗号解読されることができる。

【0097】

図 18 は、一例による、シンボル暗号解読を実行するための一方法を示している。1 組の n 個のシンボル内で定義される複数の（暗号化された）入力シンボルが、得られる 1802。異なるシンボルごとの置換を定義する複数の逆変換テーブルからの、擬似ランダムに選択された逆変換テーブルが、暗号解読されるべき入力シンボルのおのおのについて得られる 1804。各入力シンボルを個別に暗号解読するために入力シンボルのおのおのについてそれらの対応する逆変換テーブルを使用して入力シンボルに対応する出力シンボルへと変換する 1806。

【0098】

そのような方法の一例においては、第 1 の暗号化されたシンボル（入力シンボル）が、得られ、ここでその第 1 の暗号化されたシンボルは、1 組の中の n 個のシンボルのうちの 1 つである。 n 個のシンボルを n 個のシンボルの異なる置換へと変換する第 1 の逆変換テーブルも、得られる。第 1 の逆変換テーブルは、 n 個のシンボルをシャッフルするために擬似乱数を使用することにより擬似ランダムに生成されることができる。第 1 の暗号化されたシンボルは、第 1 の逆変換テーブルを使用して第 1 の出力シンボルへと変換される。 n 個のシンボルを第 1 の変換テーブル以外の n 個のシンボルの異なる置換へと変換する第 2 の逆変換テーブルが、得られる。第 1 の出力シンボルは、第 2 の逆変換テーブルを使用して第 2 の出力シンボルへと変換される。次いで、平文シンボルが、第 1 および / または第 2 の出力シンボルに基づいて得られることができる。

【0099】

図 19 は、一例による暗号化モジュールを示すブロック図である。暗号化モジュール 1902 は、キーストリームジェネレータ 1906 に対してシードを供給するように構成された処理回路 1904 を含むことができる。キーストリームジェネレータ 1906 は、処理回路 1904 に送られる擬似ランダムな数またはシンボルのキーストリームを生成する

。処理回路 1904 に結合された入力インターフェース 1908 は、平文シンボルストリームを受信することができる。平文シンボルストリームを暗号化するために、処理回路 1904 は、変換テーブルジェネレータ 1910 から変換テーブルを得るためにキーストリームから得られる擬似乱数を使用するように構成されていることができる。変換テーブルジェネレータ 1910 は、変換テーブルを提供するために擬似ランダムな、バイアスされない方法で、例えば、基本のテーブルのシンボルをシャッフルし、かつ / または組み合わせるために擬似乱数を使用するように構成されていることができる。次いで、処理回路 1904 は、第 1 の平文シンボルを暗号化されたシンボルストリームの第 1 の暗号化されたシンボルへと変換するために変換テーブルを一度使用する。暗号化されたシンボルストリームは、処理回路 1904 に結合された出力インターフェース 1912 を介して送信されることができる。平文シンボルストリームの中の各平文シンボルについて、異なる変換テーブルが、そのシンボルを変換するために生成され、そして使用されることができる。

【0100】

図 20 は、一例による暗号解読モジュールを示すブロック図である。暗号解読モジュール 2002 は、キーストリームジェネレータ 2006 に対してシードを供給するように構成された処理回路 2004 を含むことができる。キーストリームジェネレータ 2006 は、処理回路 2004 に送られる擬似ランダムな数またはシンボルのキーストリームを生成する。処理回路 2004 に結合された入力インターフェース 2008 は、暗号化されたシンボルストリームを受信することができる。暗号化されたシンボルストリームを暗号解読するために、処理回路 2004 は、逆変換テーブルジェネレータ (reverse translation table generator) 2010 から変換テーブルを得るためにキーストリームから得られる擬似乱数を使用するように構成されていることができる。逆変換テーブルジェネレータ 2010 は、変換テーブルを提供するために擬似ランダムな、バイアスされない方法で、例えば、基本のテーブルのシンボルをシャッフルし、かつ / または組み合わせるために擬似乱数を使用するように構成されていることができる。次いで、処理回路 2004 は、第 1 の暗号化されたシンボルを平文シンボルストリームの第 1 の平文シンボルへと変換するために逆変換テーブルを一度使用する。平文シンボルストリームは、処理回路 2004 に結合された出力インターフェース 2012 を介して送信されることができる。

【0101】

暗号化モジュール 1902 と、暗号解読モジュール 2002 とが、それぞれシンボルを適切に暗号化し、そして暗号解読するために、それらは、同じキーストリームジェネレータを有し、そして相補的な変換テーブルジェネレータを有することができる。キーストリームジェネレータ 1906 と 2006 とを同期化するために、共通のシードが、暗号化モジュールと、暗号解読モジュールとの間の特定の通信セッションについて (例えば、セキュリティ保護された認証スキームによって) 確立されることができる。例えば、セッションキーは、キーストリームジェネレータ 1906 と 2006 とについてのシードとして使用されることができる。

【0102】

ここにおいて説明される例のうちのいくつかは、DTMF トーンの暗号化について言及しているが、ここにおいて説明される暗号化方法は、送信された情報をセキュリティ保護するために多くの他のタイプの通信システムを用いてインプリメントされることができる。

【0103】

図 1 ~ 18 に示される 1 つまたは複数のコンポーネント、ステップ、および / またはファンクションは、本発明を逸脱することなく、単一のコンポーネント、ステップ、および / またはファンクションへと並べ換えられ、かつ / または組み合わせられ、あるいはいくつかのコンポーネント、ステップ、および / またはファンクションへと分離されることができる。追加の要素、コンポーネント、ステップ、および / またはファンクションもまた、本発明を逸脱することなく、加えられることができる。図 1、2、3、5、7、9、13、17、19 および / または 20 に示される装置、デバイス、および / またはコンポーネ

ントは、図 2、4、6、8、10、11、12、14、15、16 および / または 18 に説明される 1 つまたは複数の方法、機能、またはステップを実行するように構成されていることができる。

【0104】

当業者は、さらに、ここにおいて開示される例に関連して説明される様々な例示の論理ブロック、モジュール、回路、およびアルゴリズムステップが、電子的なハードウェア、コンピュータソフトウェア、またはそれらの両方の組合せとしてインプリメントされることができることを理解するであろう。ハードウェアとソフトウェアとのこの交換可能性を明確に説明するために、様々な例示のコンポーネント、ブロック、モジュール、回路、およびステップは、それらの機能に関して、上記に一般的に説明されている。それらの機能がハードウェアとしてインプリメントされるか、あるいはソフトウェアとしてインプリメントされるかは、特定のアプリケーションと、全体システムに課される設計制約条件と、に依存する。

【0105】

上記コンフィギュレーションは、単なる例にすぎず、そして本発明を限定するものとして解釈されるべきでないことに注意すべきである。これらの例の説明は、例示的であり、請求の範囲の範囲を限定すべきでないように意図される。したがって、本教示は、他のタイプの装置、および多数の代替案に対しても容易に適用されることができ、修正、および変形は、当業者にとって明らかであろう。

以下に、本願出願の当初の特許請求の範囲に記載された発明を付記する。

[1] 暗号化デバイス上で動作する方法であって、

複数の入力シンボルを得ることと、

暗号化されるべき前記入力シンボルのおのおのについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを得ることと

、
各入力シンボルを個別に暗号化するために、前記入力シンボルのおのおのについてそれらの対応する変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへ変換することと、

を備える方法。

[2] 暗号化されるべき前記入力シンボルのおのおのについて前記擬似ランダムに選択された変換テーブルを得ることは、

第 1 の入力シンボルについての、擬似ランダムに選択された第 1 の変換テーブルを得ることと、

第 2 の入力シンボルについての、擬似ランダムに選択された第 2 の変換テーブルを得ることと、

を含み、

そして、前記入力シンボルのおのおのについてそれらの対応する変換テーブルを使用して前記入力シンボルを対応する出力シンボルへ変換することは、

前記第 1 の変換テーブルを使用して前記第 1 の入力シンボルを第 1 の出力シンボルへ変換することと、

前記第 2 の変換テーブルを使用して前記第 2 の入力シンボルを第 2 の出力シンボルへ変換することと、

を含む、

[1] に記載の方法。

[3] 前記複数の入力シンボルは、1 組の N 個のシンボルによって定義される、但し、N は、正の整数であり、変換テーブルは、前記 N 個のシンボルの置換である、[1] に記載の方法。

[4] 暗号化されるべき前記入力シンボルのおのおのについて、前記擬似ランダムに選択された変換テーブルを得ることは、

暗号化されるべき第 1 の入力シンボルについての第 1 の擬似乱数を得ることと、

前記組の N 個のシンボルの異なる置換を得るために前記組の N 個のシンボルをシャッフルすること、および前記第 1 の入力シンボルについての前記変換テーブルとしてその置換を使用することと、

を含む、

[3] に記載の方法。

[5] 前記第 1 の擬似乱数は、

前記第 1 の入力シンボルについての擬似乱数を生成することと、但し、前記擬似乱数は k ビットの長さであり、k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを決定することと、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨てることと；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を得ることと；

前記第 1 の擬似乱数を得るために前記許容可能な擬似乱数のモジュロ N の階乗を除算することと；

によって得られる、

[4] に記載の方法。

[6] 前記組の N 個のシンボルをシャッフルすることは、

前記組の N 個のシンボルのうちのすべてのシンボルを用いて置換ベクトル P を初期化することと、

前記第 1 の擬似乱数に基づいて前記置換ベクトルの中の前記シンボルをシャッフルすることと、

を含む、

[4] に記載の方法。

[7] 前記出力シンボルを暗号解読デバイスに送信すること、

をさらに備える [1] に記載の方法。

[8] 暗号化されるべき前記入力シンボルのおののおについて、異なるシンボルごとの置換を定義する前記複数の変換テーブルから、第 2 の擬似ランダムに選択された変換テーブルを得ることと、

各入力シンボルをさらに個別に暗号化するために、前記出力シンボルのおののおについてそれらの対応する第 2 の変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへ変換することと、

をさらに備える [1] に記載の方法。

[9] 複数の入力シンボルを得るための手段と、

暗号化されるべき前記入力シンボルのおののおについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを得るための手段と、

各入力シンボルを個別に暗号化するために、前記入力シンボルのおののおについてそれらの対応する変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへ変換するための手段と、

を備える暗号化デバイス。

[10] 第 1 の入力シンボルについての、擬似ランダムに選択された第 1 の変換テーブルを得るための手段と、

第 2 の入力シンボルについての、擬似ランダムに選択された第 2 の変換テーブルを得るための手段と、

前記第 1 の変換テーブルを使用して前記第 1 の入力シンボルを第 1 の出力シンボルへ変換するための手段と、

前記第 2 の変換テーブルを使用して前記第 2 の入力シンボルを第 2 の出力シンボルへ変換するための手段と、

をさらに備える [9] に記載のデバイス。

[1 1] 暗号化されるべき第 1 の入力シンボルについての第 1 の擬似乱数を得るための手段と、

前記組の N 個のシンボルの異なる置換を得るために前記組の N 個のシンボルをシャッフルし、そして前記第 1 の入力シンボルについての前記変換テーブルとしてその置換を使用するための手段と、

をさらに備える [9] に記載のデバイス。

[1 2] 前記第 1 の入力シンボルについての擬似乱数を生成するための手段と、但し、前記擬似乱数は k ビットの長さであり、k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを決定するための手段と、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨てるための手段と；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を得るための手段と；

前記第 1 の擬似乱数を得るために前記許容可能な擬似乱数のモジュロ N の階乗を除算するための手段と；

をさらに備える [1 1] に記載のデバイス。

[1 3] 前記出力シンボルを暗号解読デバイスに送信するための手段、

をさらに備える [9] に記載のデバイス。

[1 4] 暗号化されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の変換テーブルから、第 2 の擬似ランダムに選択された変換テーブルを得るための手段と、

各入力シンボルをさらに個別に暗号化するために、前記出力シンボルのおののについてそれらの対応する第 2 の変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへ変換するための手段と、

をさらに備える [9] に記載のデバイス。

[1 5] 入力シンボルストリームを受信するための入力インターフェースと、

前記入力インターフェースに結合された処理回路と、

を備え、

前記処理回路は、

複数の入力シンボルを前記入力インターフェースから取得し、

暗号化されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを取得し、そして

各入力シンボルを個別に暗号化するために、前記入力シンボルのおののについてそれらの対応する変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへ変換する、

ように構成されている、

暗号化デバイス。

[1 6] 前記出力シンボルを送信するための、前記処理回路に結合された出力インターフェース、

をさらに備える [1 5] に記載のデバイス。

[1 7] 前記複数の入力シンボルは、1 組の N 個のシンボルによって定義されている、但し、N は正の整数であり、変換テーブルは、前記 N 個のシンボルの置換である、[1 5] に記載のデバイス。

[1 8] 前記処理回路に結合されたキーストリームジェネレータと、なお、前記キーストリームジェネレータは、暗号化されるべき第 1 の入力シンボルについて前記キーストリームジェネレータから第 1 の擬似乱数を取得するように構成されている；

前記処理回路に結合された変換テーブルジェネレータと、なお、前記変換テーブルジェネレータは、前記組の N 個のシンボルの異なる置換を取得するために、前記組の N 個のシ

ンボルをシャッフルするように構成され、そして前記第 1 の入力シンボルについての前記変換テーブルとしてその置換を使用する；

をさらに備える [1 7] に記載のデバイス。

[1 9] 前記処理回路は、さらに、

前記第 1 の入力シンボルについての擬似乱数を生成し、なお、前記擬似乱数は k ビットの長さであり、 k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを決定し、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨て；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を取得し；そして

前記第 1 の擬似乱数を取得するために前記許容可能な擬似乱数のモジュロ N の階乗を除算する；

ように構成されている、

[1 7] に記載のデバイス。

[2 0] 前記処理回路は、さらに、

暗号化されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の変換テーブルから、第 2 の擬似ランダムに選択された変換テーブルを取得し、そして

各入力シンボルをさらに個別に暗号化するために、前記出力シンボルのおののについてそれらの対応する第 2 の変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへ変換する、

ように構成されている、

[1 7] に記載のデバイス。

[2 1] プロセッサによって実行されるときに、前記プロセッサに、

複数の入力シンボルを取得させ、

暗号化されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の変換テーブルから、擬似ランダムに選択された変換テーブルを取得させ、そして

各入力シンボルを個別に暗号化するために、前記入力シンボルのおののについてそれらの対応する変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへ変換させる、

シンボルを暗号化するために動作可能な 1 つまたは複数の命令、を有する機械可読媒体。

[2 2] 前記複数の入力シンボルは、1 組の N 個のシンボルによって定義される、但し、 N は正の整数であり、変換テーブルは、前記 N 個のシンボルの置換である、[2 1] に記載の機械可読媒体。

[2 3] プロセッサによって実行されるときに、前記プロセッサに、さらに、

暗号化されるべき第 1 の入力シンボルについての第 1 の擬似乱数を取得させ、そして

前記組の N 個のシンボルの異なる置換を取得するために、前記組の N 個のシンボルをシャッフルさせ、そして前記第 1 の入力シンボルについての前記変換テーブルとしてその置換を使用させる、

1 つまたは複数の命令、を有する [2 2] に記載の機械可読媒体。

[2 4] プロセッサによって実行されるときに、前記プロセッサに、さらに、

前記第 1 の入力シンボルについての擬似乱数を生成させ、なお、前記擬似乱数は k ビットの長さであり、 k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを決定させ、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨てさせ、；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シ

ンボルについての異なる擬似乱数を取得させ；そして

前記第 1 の擬似乱数を取得するために前記許容可能な擬似乱数のモジュロ N の階乗を除算させる、

1 つまたは複数の命令、を有する [2 2] に記載の機械可読媒体。

[2 5] プロセッサによって実行されるときに、前記プロセッサに、さらに、

暗号化されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の変換テーブルから、第 2 の擬似ランダムに選択された変換テーブルを取得させ、そして

各入力シンボルをさらに個別に暗号化するために、前記出力シンボルのおののについてそれらの対応する第 2 の変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへ変換させる、

1 つまたは複数の命令、を有する [2 1] に記載の機械可読媒体。

[2 6] シンボルを暗号解読するための方法であって、

1 組の n 個のシンボル内で定義される複数の入力シンボルを得ることと、

暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを得ることと、

各入力シンボルを個別に暗号解読するために、前記入力シンボルのおののについてそれらの対応する逆変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへ変換することと、

を備える方法。

[2 7] 前記複数の入力シンボルは、1 組の N 個のシンボルによって定義され、但し、N は正の整数であり、逆変換テーブルは前記 N 個のシンボルの置換であり、

さらに前記方法は、

暗号解読されるべき第 1 の入力シンボルについての第 1 の擬似乱数を得ることと、

前記組の N 個のシンボルの異なる置換を得るために前記組の N 個のシンボルをシャッフルすること、および前記第 1 の入力シンボルについての前記逆変換テーブルとしてその置換を使用することと、

をさらに備える、

[2 6] に記載の方法。

[2 8] 前記第 1 の擬似乱数は、

前記第 1 の入力シンボルについての擬似乱数を生成することと、なお、前記擬似乱数は k ビットの長さであり、k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを決定することと、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨てることと；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を得ることと；

前記第 1 の擬似乱数を得るために前記許容可能な擬似乱数のモジュロ N の階乗を除算することと；

によって得られる、[2 7] に記載の方法。

[2 9] 暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の逆変換テーブルから、第 2 の擬似ランダムに選択された逆変換テーブルを得ることと、

各入力シンボルをさらに個別に暗号解読するために、前記出力シンボルのおののについてそれらの対応する第 2 の逆変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへ変換することと、

をさらに備える [2 6] に記載の方法。

[3 0] 1 組の n 個のシンボル内で定義される複数の入力シンボルを得るための手段と、暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換

を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを得るための手段と、

各入力シンボルを個別に暗号解読するために前記入力シンボルのおののおのについてそれらの対応する逆変換テーブルを使用して前記入力シンボルに対応する出力シンボルへと変換するための手段と、

を備える暗号解読デバイス。

[3 1] 前記複数の入力シンボルは、1組のN個のシンボルによって定義され、但し、Nは正の整数であり、逆変換テーブルは前記N個のシンボルの置換であり、

前記デバイスはさらに、

暗号解読されるべき第1の入力シンボルについての第1の擬似乱数を得るための手段と

、

前記組のN個のシンボルの異なる置換を得るために、前記組のN個のシンボルをシャッフルし、そして前記第1の入力シンボルについての前記逆変換テーブルとしてその置換を使用するための手段と、

を備える、

[3 0] に記載のデバイス。

[3 2] 前記第1の擬似乱数は、

前記第1の入力シンボルについての擬似乱数を生成するための手段と、なお、前記擬似乱数はkビットの長さであり、kは正の整数である；

前記擬似乱数が最大数Pmaxの範囲内にあるかどうかを、決定するための手段と、但し、Pmaxは、最大しきい値 2^k よりも小さいNの階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数Pmaxよりも大きい場合は、捨てるための手段と；

前記最大数Pmax以下である許容可能な擬似乱数が得られるまで、前記第1の入力シンボルについての異なる擬似乱数を得るための手段と；

前記第1の擬似乱数を得るために前記許容可能な擬似乱数のモジュロNの階乗を除算するための手段と；

によって得られる、

[3 1] に記載のデバイス。

[3 3] 暗号解読されるべき前記入力シンボルのおののおのについて、異なるシンボルごとの置換を定義する前記複数の逆変換テーブルから、第2の擬似ランダムに選択された逆変換テーブルを得るための手段と、

各入力シンボルをさらに個別に暗号解読するために、前記出力シンボルのおののおのについてそれらの対応する第2の逆変換テーブルを使用して、前記出力シンボルに対応する第2の出力シンボルへと変換するための手段と、

をさらに備える [3 0] に記載のデバイス。

[3 4] 入力シンボルストリームを受信するための入力インターフェースと、

前記入力インターフェースに結合された処理回路と、

を備え、

前記処理回路は、

1組のn個のシンボル内で定義される複数の入力シンボルを取得し、

暗号解読されるべき前記入力シンボルのおののおのについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを取得し、そして

各入力シンボルを個別に暗号解読するために、前記入力シンボルのおののおのについてそれらの対応する逆変換テーブルを使用して、前記入力シンボルに対応する出力シンボルへと変換する、

ように構成されている、

暗号解読デバイス。

[3 5] 前記複数の入力シンボルは、1組のN個のシンボルによって定義され、但し、N

は正の整数であり、逆変換テーブルは前記 N 個のシンボルの置換であり、

前記デバイスはさらに、

前記処理回路に結合されたキーストリームジェネレータと、なお、前記キーストリームジェネレータは、暗号解読されるべき第 1 の入力シンボルについて前記キーストリームジェネレータから第 1 の擬似乱数を取得するように構成されている；

前記処理回路に結合された逆変換テーブルジェネレータと、なお、前記逆変換テーブルジェネレータは、前記組の N 個のシンボルの異なる置換を取得するために、前記組の N 個のシンボルをシャッフルするように構成され、そして前記第 1 の入力シンボルについての前記逆変換テーブルとしてその置換を使用する；

を備える [3 4] に記載のデバイス。

[3 6] 前記複数の入力シンボルは、1 組の N 個のシンボルによって定義され、但し、N は正の整数であり、逆変換テーブルは前記 N 個のシンボルの置換であり、

前記処理回路は、さらに、

前記第 1 の入力シンボルについての擬似乱数を生成し、なお、前記擬似乱数は k ビットの長さであり、k は正の整数である；

前記擬似乱数が最大数 P_{max} の範囲内にあるかどうかを、決定し、但し、 P_{max} は、最大しきい値 2^k よりも小さい N の階乗の最大の倍数である；

前記擬似乱数を、もしそれが前記最大数 P_{max} よりも大きい場合は、捨て；

前記最大数 P_{max} 以下である許容可能な擬似乱数が得られるまで、前記第 1 の入力シンボルについての異なる擬似乱数を取得し；そして

前記第 1 の擬似乱数を得るために前記許容可能な擬似乱数のモジュロ N の階乗を除算する；

ように構成されている、

[3 4] に記載のデバイス。

[3 7] 前記処理回路は、さらに、

暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の逆変換テーブルから、第 2 の擬似ランダムに選択された逆変換テーブルを取得し、そして

各入力シンボルをさらに個別に暗号解読するために、前記出力シンボルのおののについてそれらの対応する第 2 の逆変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへと変換する、

ように構成されている、

[3 4] に記載のデバイス。

[3 8] プロセッサによって実行されるときに、前記プロセッサに、

1 組の n 個のシンボル内で定義される複数の入力シンボルを取得させ、

暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する複数の逆変換テーブルから、擬似ランダムに選択された逆変換テーブルを取得させ、そして

各入力シンボルを個別に暗号解読するために、前記入力シンボルのおののについてそれらの対応する逆変換テーブルを使用して、前記入力シンボルを対応する出力シンボルへと変換させる、

シンボルを暗号解読するための 1 つまたは複数の命令、を有する機械可読媒体。

[3 9] 前記複数の入力シンボルは、1 組の N 個のシンボルによって定義され、但し、N は正の整数であり、逆変換テーブルは前記 N 個のシンボルの置換であり、

そして前記機械可読媒体は、

プロセッサによって実行されるときに、前記プロセッサに、さらに、

暗号解読されるべき第 1 の入力シンボルについての第 1 の擬似乱数を取得させ、そして前記組の N 個のシンボルの異なる置換を取得するために、前記組の N 個のシンボルをシャッフルさせ、そして前記第 1 の入力シンボルについての前記逆変換テーブルとしてその置換を使用させる、

1つまたは複数の命令、をさらに備える、

[3 8] に記載の機械可読媒体。

[4 0] プロセッサによって実行されるときに、前記プロセッサに、さらに、

暗号解読されるべき前記入力シンボルのおののについて、異なるシンボルごとの置換を定義する前記複数の逆変換テーブルから、第 2 の擬似ランダムに選択された逆変換テーブルを取得させ、そして

各入力シンボルをさらに個別に暗号解読するために、前記出力シンボルのおののについてそれらの対応する第 2 の逆変換テーブルを使用して、前記出力シンボルを対応する第 2 の出力シンボルへと変換させる、

1つまたは複数の命令、を有する [3 8] に記載の機械可読媒体。

【誤訳訂正 3】

【訂正対象書類名】図面

【訂正対象項目名】図 1 2

【訂正方法】変更

【訂正の内容】

【図 1 2】

図 12

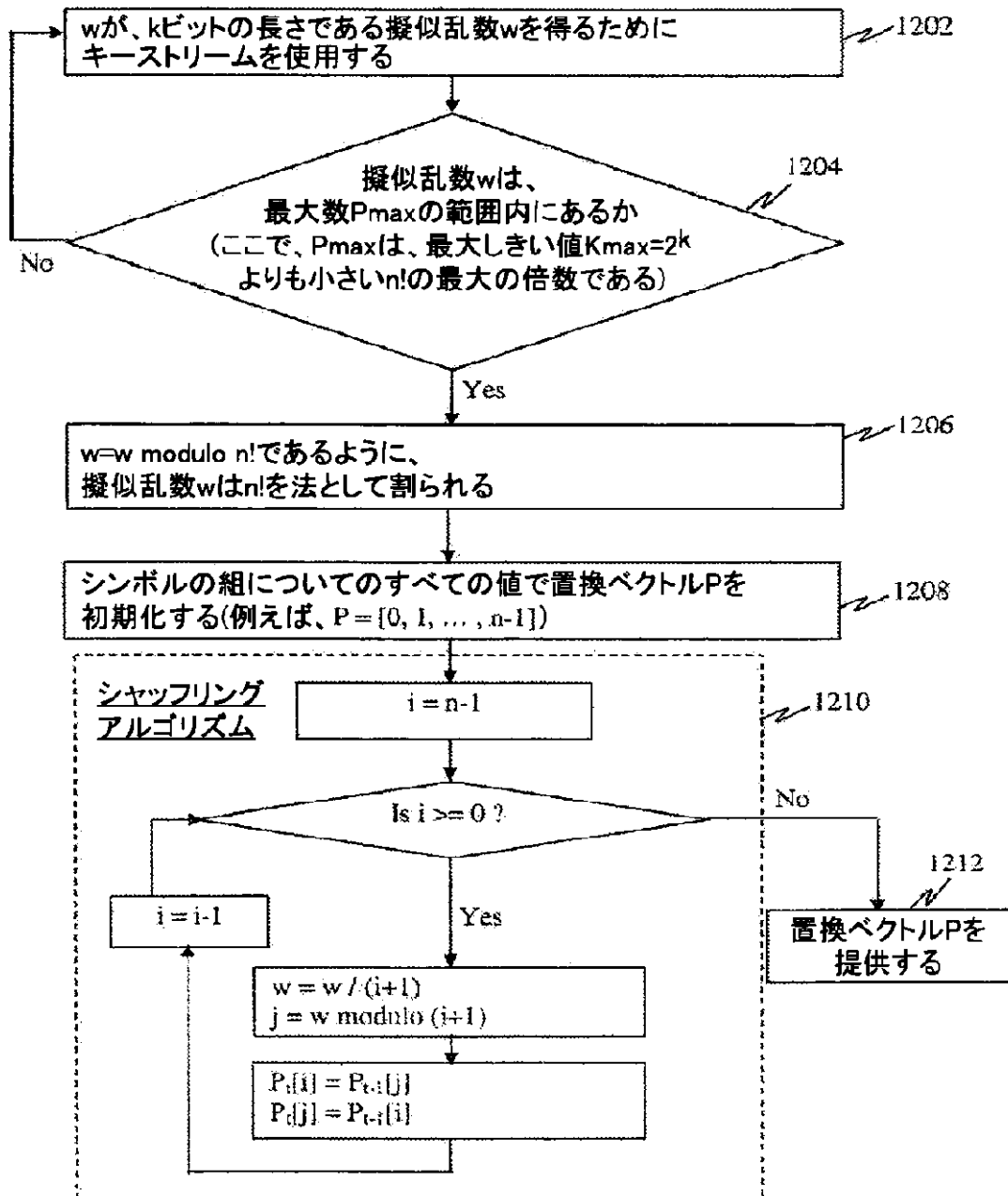


Figure 12

【誤訳訂正 4】

【訂正対象書類名】図面

【訂正対象項目名】図 1 6

【訂正方法】変更

【訂正の内容】

【図 16】

図 16

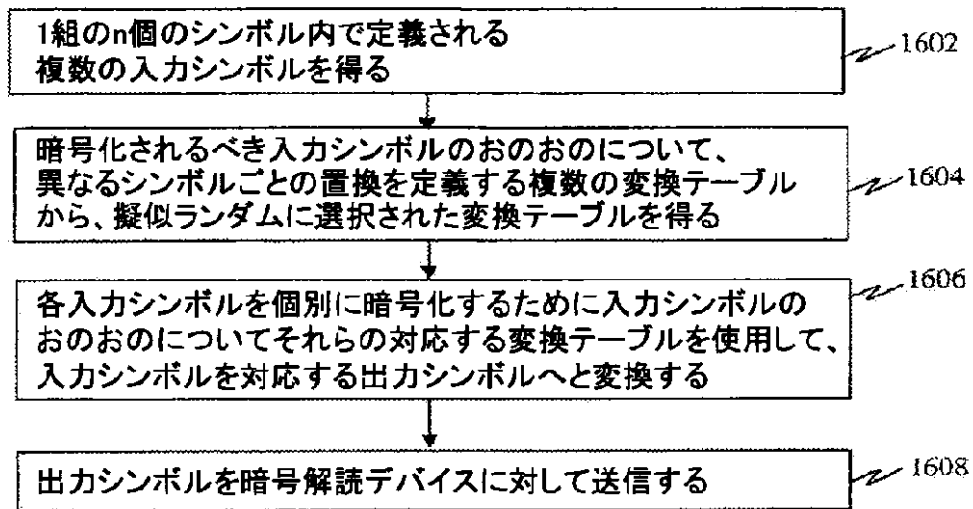


Figure 16

【誤訳訂正 5】

【訂正対象書類名】図面

【訂正対象項目名】図 18

【訂正方法】変更

【訂正の内容】

【図 18】

図 18

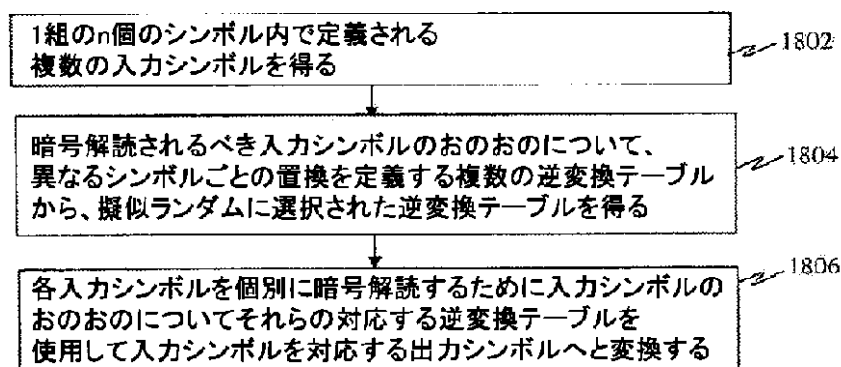


Figure 18