マルチメディア通信と分散処理 40－5
（１９８９．１．１９）

# マルチメディア向け高速暗号方式

宝木 和夫* 佐々木 良一* 中川 聡夫**
＊ 日立製作所システム開発研究所 ＊＊ 日立コントロールシステムズ

ISDN等において、マルチメディアデータの盗取、改ざん等に対する情報セキュリティを確保するため、大型計算機やワークステーションのソフトで高速な暗号化を行う対称鍵暗号アルゴリズムを開発した。本アルゴリズムは、ソフトでの高速高安全な暗号化を実現するため、1回の基本マシーン命令で32ビット単位の換字、転置を無駄なく行う点が特徴である。本開発により、例えば、2050/32のソフトで暗号化する場合、1ビット処理基調のDES(米国標準方式)に比べ約65倍(270Kbps以上)の高速化を得る見通しを得た。さらに、本アルゴリズムの応用として、電子認証(ディジタル署名等)で必要な圧縮暗号化を高速に行える(10Kビットのデータを0.04秒以内に圧縮する)見通しも得た。

## Multi-Media Encryption Algorithm

Kazuo Takaragi* Ryoichi Sasaki* Fusao Nakagawa**

* Systems Development Laboratory, Hitachi, Ltd.
1099 Ohzenji Asao-ku Kawasaki-shi, Kanagawa-ken, 215 Japan.

** Hitachi Control Systems, Ltd.
5-2-1 Omika-cho, Hitachi-shi, Ibaraki-ken, 319-12 JAPAN

Fast encipherment algorithm of symmetric cryptosystem is developed to secure the multi-media communications. In the algorithm, a substitution-permutaion network is constructed only by using the fast basic operations of 32-bits machine to enable efficient software executions in main frame computers and workstations. The encryption/decryption speed is about 270 kbps, sixty five times as fast as DES, using the workstation Hitachi 2050/32. Safety in view of randomness is numerically proved. Furthermore, a fast hush function, the execution time is less than 0.04 seconds for 10K bits data, is presented as an application of this algorithm.

## 1．はじめに

ISDN、LAN等において、通信データの盗み見や改ざん等に対する情報セキュリティを確保するため、マルチメディア向きの高速暗号アルゴリズムを開発した。

マルチメディアとは、従来は個別に取り扱われてきたコンピュータ・コード化されたデータやファクシミリ等のイメージ情報、電話の音声等の情報を複合して取り扱うようにした一種の複合メディアである。今回、開発したマルチメディア向きの高速暗号アルゴリズムとは、このマルチメディアのデータをISDNやLAN等を使って伝送する場合に、送信側で暗号化して送り、受信側で元の平文データに復号するという暗号変換、復号変換を行う対称鍵(慣用)暗号アルゴリズムである。

本アルゴリズムは、情報セキュリティシステムを構築するうえで使用できる要素技術の一つとして位置付けられ（図１参照）、Hisecurity-Multi2（略称、Multi2）と呼ぶ。

今回、マルチメディア向きの高速暗号アルゴリズムを開発した背景は次のとおりである。

(1)　近い将来、ISDNやLANの進展に伴い、文書、画像、音声等を統合化したメディアの通信、つまり、マルチメディアの通信が普及すると予想されている。

(2)　衛星通信やISDNの普及ならびにLANの大規模化に応じ、これら通信データの盗取や改ざん等に対するセキュリティ対策が必要となり暗号化のニーズが生じると考えられている。

そこで、次を目的として、マルチメディア向きの暗号アルゴリズムを開発した。

(1)　マルチメディア通信の高速化に容易に対処できるようワークステーションレベルの計算機のソフトでも高速に暗号化できるアルゴリズムを開発する。

(2)　開放型ネットワークの暗号通信にも適用し易いよう、DES(米国の標準方式)[1]やFEAL-8[2]のように、暗号アルゴリズムが知られても秘密鍵を秘匿するだけで暗号通信の安全性が保たれるような安全性の高い暗号アルゴリズムを開発する。

(3)　既存の暗号アルゴリズムDESやFEAL-8に比べて、安全性および高速性において同等あるいはそれ以上のアルゴリズムを開発する。

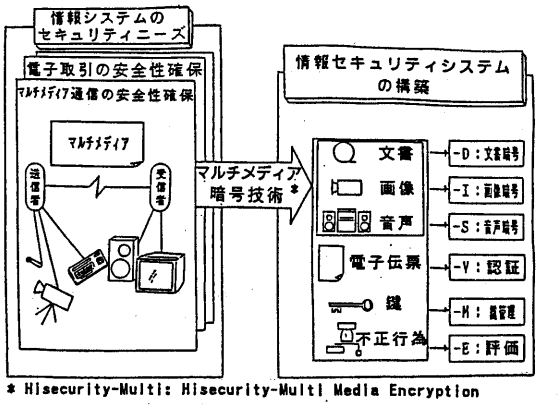


図１　情報セキュリティシステムにおけるマルチメディア暗号の位置付け

上記の項目を満足するような暗号アルゴリズムはメッセージ認証コード等に用いられる圧縮暗号(ハッシュトータル)の作成にも応用できることが知られている。そこで、本開発では、上記暗号アルゴリズムの開発とともに電子認証[3]を実現するうえで必要な圧縮暗号を高速に生成する技術を開発することも狙いとした。

DESは、LSI化向けの1ビット単位の処理に、FEAL-8は、8ビットマイクロコンピュータ向けの8ビット単位の処理等に特徴がある。

今回、提案するマルチメディア暗号アルゴリズムMulti2は、32ビットマイクロコンピュータのソフトにより、高速(270kbps以上)にデータの暗号化、復号化を行うものである。本暗号は対称鍵(慣用)暗号の一種であり、RSAに代表される非対称鍵(公開鍵)暗号には属さない。

本暗号アルゴリズは、大型計算機やクリエイティブワークステーション2050等を使ったコンピュータ間通信において、伝送路のデー

タが盗聴されたり、改ざんされたりすることを防止する目的で用いることができる。また、本アルゴリズムを暗号ブロックフィードバックモード(本文で説明)で使用することにより、高速に(270Kbps以上)圧縮暗号(ハッシュトータル)を生成することができ、電子認証技術の一要素として使用できる。

具体的には、業務処理用大容量通信、情報提供サービス通信、オフィスＯＡ通信等において、データ盗聴等の不当行為に対する情報セキュリティを確保したり、通信データの正当性を保証するための認証サービスを提供するうえで用いることができる。

## 2．基本的な考え方

対称鍵(慣用)暗号アルゴリズムは、使用される環境によって次の2種類に分類される。

(1)　送信者と受信者との間で、暗号アルゴリズムを秘匿して通信を行なう。つまり、もし悪意の第三者がいて暗号化された通信文を盗聴しても、第三者は暗号アルゴリズムが判らないので暗号文を解読することができない。

(2)　送信者と受信者との間で、暗号鍵(対称鍵暗号なので復号鍵としても使える)を秘密に持ち合い通信を行なう。暗号アルゴリズムは公知のものを用いる。つまり、もし悪意の第三者がいて暗号化された通信文を盗聴しても、第三者は暗号鍵が判らないので暗号文を解読することができない。ただし、第三者は暗号アルゴリズムは知っている。DESやFEAL-8等がこの分類に属する暗号アルゴリズムである。

ここで、開発する暗号アルゴリズムMulti2は上記(2)に属するものである。つまり、暗号アルゴリズムは皆がよく知っているものを用い、ある送信者と受信者が暗号通信をする必要が生じたときには、この二者間での秘密の暗号鍵を用いる。この方式は、開放型システム間結合(OSI)における暗号通信に適している。なんとなれば、本方式ではシステムに属するメンバー全員が同一の暗号ソフト／ハードを用いれば良いので、暗号通信の環境を比較的容易に設定できるからである。

ここでは、DESやFEAL-8よりも速度、安全性の面で優れた暗号アルゴリズムを開発することを目的とする。暗号アルゴリズムの基本形として、換字･転置(SP:substitution permutation)ネットワーク[4]を採用する。

換字･転置ネットワークの一例を図２に示す。換字･転置ネットワークは、与えられた平文データに対し、暗号鍵をパラメータとして換字･転置を繰り返し行ない、一見ランダムな暗号文データに変換するものである。

送信者は、通信したい平文データに対し、この暗号化処理を行ない、その結果として得られた暗号文を受信者に送る。受信者は送信者のものと同じ暗号鍵を持っており、前記換字･転置の処理を逆に行なうことによって元の平文を得る。

この換字・転置ネットワークはDESやFEAL-8のように強い暗号アルゴリズムを構成する

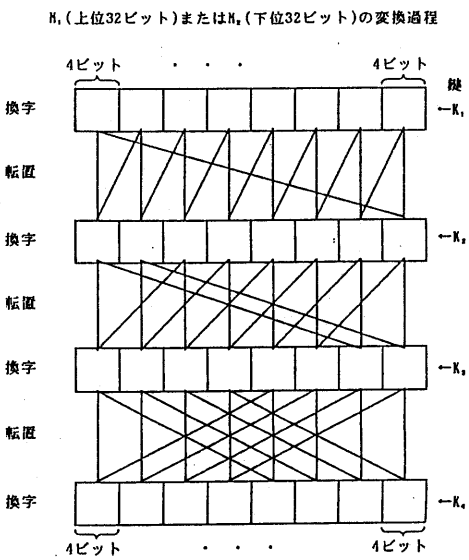


図２　換字・転字ネットワークの一例

うえで有用であると考えられている。そこで、本稿でも、換字･転置ネットワークを用いてDESやFEAL-8よりも速く、かつ、安全な暗号アルゴリズムを開発することを目的する。つまり、次の点に着目して暗号アルゴリズムを構成する。

(1) 本暗号アルゴリズムMulti2の基本構造として、換字･転置ネットワークを採用する。

(2) なるべく、32ビットマイクロプロセッサの基本マシーン命令1回当りでの32ビット単位の換字、あるいは、転置が効率よく行なえるようにする。これにより、1ビット単位で換字または転置を行なうDESや8ビット単位で処理を行なうFEAL-8に比べて、全体としての暗号変換速度をより速いものにする。

## 3．アルゴリズム

Multi2はDES、FEAL-8と同様に、64ビットの暗号鍵に鍵スケジュールと呼ばれる操作を施し（図５参照）、32ビット×8の実効鍵 $k_1$ ～ $k_8$ を生成し、この実効鍵を用いて暗号処理を行う。暗号処理は基本的に鍵スケジュールと同じ構成であるが、この処理は４種類の基本関数 $\pi_1$, $\pi_2$, $\pi_3$, $\pi_4$ から成っており、平文をM、暗号文をCとすると、

$$C = \pi_4\pi_3\pi_2\pi_1\pi_4\pi_3\pi_2\pi_1(M) \quad \cdots(1)$$

で表される。$\pi_4\pi_3\pi_2\pi_1$ を１段の処理とし、これを２段繰り返すことによって複雑な暗号関数を実現している（図３）。

４種類の基本関数の具体的な処理を図４に示す。なお図中

$$\mathrm{Rot}_n(x)$$

とあるのは、nビット左巡回シフトを、

$$+$$

は、$2^{32}$ を法とした加算を、

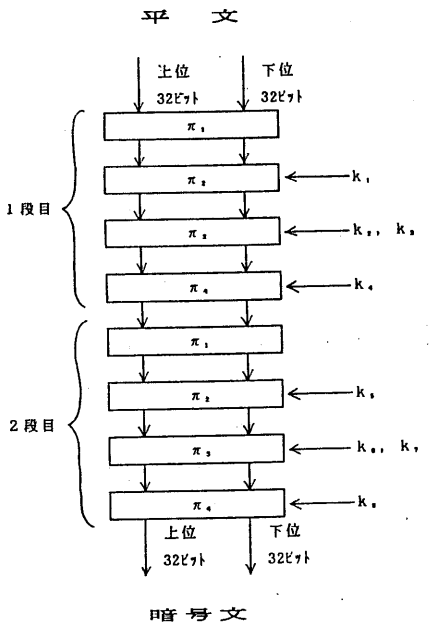


図３　暗号処理フロー

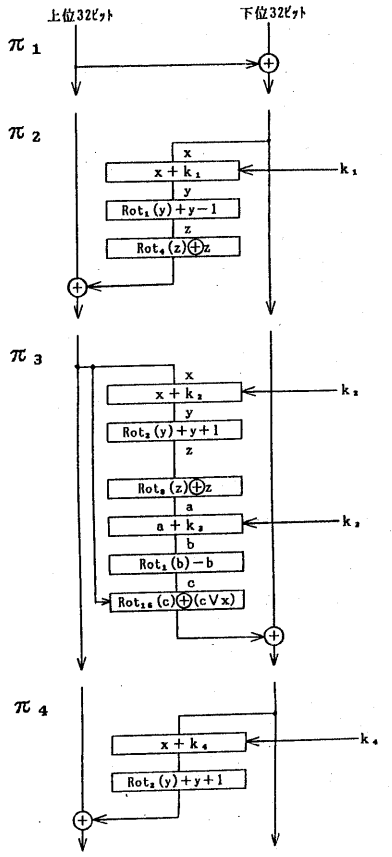


図４　暗号処理における基本関数

$\vee$

は、ビットごとの論理和を、

$\oplus$

は、ビットごとの排他的論理和を示す。

また、図５に鍵スケジュール部の処理を示す。図中、$J_1$～$J_8$は鍵スケジュールを行うために必要な定数であり、システムにより一意に定められる。

なお、図３～図５において、データの処理単位は32ビットとなっている。

ここで、各関数$\pi_i$は、

$\pi_i \cdot \pi_i(x) = x$, $(i = 1 \sim 4)$ ・・・・(2)

が成立するようなインボリューションである。

よって、復号処理は次式で与えられる。

$C = \pi_1 \pi_2 \pi_3 \pi_4 \pi_1 \pi_2 \pi_3 \pi_4(M)$ ・・・(3)

今回開発した暗号アルゴリズムは図２に示すような換字、転置の繰り返しを行なっている。

つまり、図４の基本関数のうち、

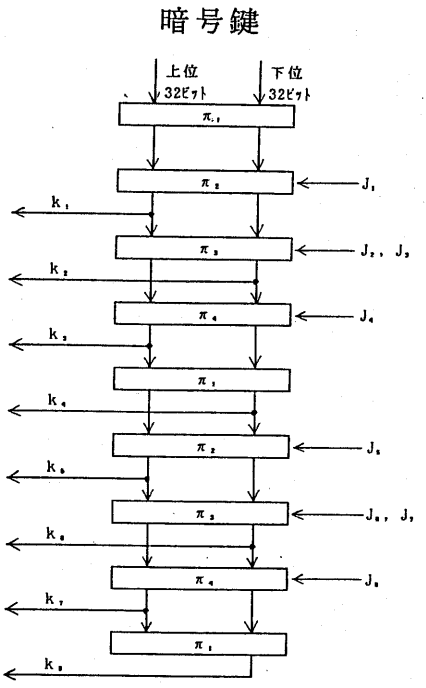


**図５　鍵スケジュール**

$x \leftarrow x + Ki$ ・・・・・・・・・・・・・・・・・・(4)

$x \leftarrow Rot^{m}(x) + (x) + \alpha$ ・・・・・・・・・・(5)

（ただし、$m = 1$ or $2$）

は、それぞれ、32ビットのデータを4ビットずつのブロックに分割したとき、1ビットの変化が1ブロック長分波及するような換字処理を32ビット分一斉に行っていると見ることができる。

また、図４の基本関数のうち、

$x \leftarrow Rot_4(x) \oplus x$ ・・・・・・・・・・・・・(5)

$x \leftarrow Rot_8(x) \oplus x$ ・・・・・・・・・・・・・(6)

$x \leftarrow Rot_{16}(x) \oplus (x \vee y)$ ・・・・・・・・(7)

は、それぞれ、

(a)　4ビット左循環シフトを行うという転置を行った後、さらに換字を行うという処理、

(b)　8ビット左循環シフトを行うという転置を行った後、さらに換字を行うという処理、

(c)　16ビット左循環シフトを行うという転置を行った後、さらに換字を行うという処理を示している。

この処理フローにおいて最初の左半分または右半分の32ビットのデータに着目すると、図２と同様に、それらのいかなるビットの変化も最後の32ビットのデータすべてに影響を与えることが分かる。これにより、本実施例は、高度なランダム性を持つ暗号変換を行うという効果が得られることが分かる。

## ４．処理速度

Multi2、DES、FEAL-8の暗号プログラムを、ワークステーション2050/32(CPU：MC68020、20MHz）上で、Ｃ言語を用いて開発し、処理速度を実測した。その結果を表１に示す。

表１において、Multi2の暗号化速度は278kbpsとなっているが、Ｃ言語でなくアセンブラで暗号プログラムを作った場合、さらに高速な結果が得られると予想される。

表1　処理速度の比較

| | | Multi2 | FEAL-8 | DES |
|---|---|---|---|---|
| 鍵スケジュールを毎回行う場合 | 処理速度(kbps) | 136 | 68 | 2.8 |
| | 速度比 | 49 | 24 | 1 |
| 鍵スケジュールを最初だけ行う場合 | 処理速度(kbps) | 278 | 133 | 4.3 |
| | 速度比 | 65 | 31 | 1 |

## 5．暗号破りに対する安全性の検討

一般的に、絶対安全な暗号アルゴリズムは存在し得ないと考えられているが、以下の評価尺度により、本方式がどの程度安全であるかという目安が得られる。

### 5.1　平文－暗号文の組が与えられたときの暗号鍵のしらみつぶし探索方法

本暗号アルゴリズムMulti2は、32ビットマイクロコンピュータで実行する場合、基本マシン命令換算で60ステップ程度(2段処理の場合)を要するが、仮にスーパーコンピュータで1 $\mu$secに1000個の暗号鍵を試すと仮定すると、真の暗号鍵を発見するまで、平均して次の時間がかかる。

鍵長64ビットの場合：

$$2^{63}/1000\,\mu\text{sec} \fallingdotseq 290 \text{ year}$$

鍵長128ビットの場合:

$$2^{127}/1000\,\mu\text{sec} \fallingdotseq 5.4\times10^{21} \text{ year}$$

したがって、このしらみつぶし探索方法によっては本暗号アルゴリズムはまず破られない。

### 5.2　暗号文を入手されたときの近似攻撃

第二次世界大戦中から採用されている暗号破りの方法がある。この方法は、予め文字、あるいは、単語が出現する頻度の分布を統計にとっておき、入手した暗号文の文字列パターンの頻度分布とのマッチングをとるものである[5]。例えば、平文にビット'0'の連続が多くあると予め分かっているとき、暗号文の方も何かある一定のランダムパターンが繰り返し表われる。したがって、暗号文を入手して分析するだけで、平文の方が推定される。このような統計的近似攻撃を避けるため、暗号化された64ビットの暗号ブロックを一旦フィードバックして、次の64ビットの入力データとの排他的論理和を取るというCBC(Cipher Block Chaining)モードを使用することが奨められる(図6)。

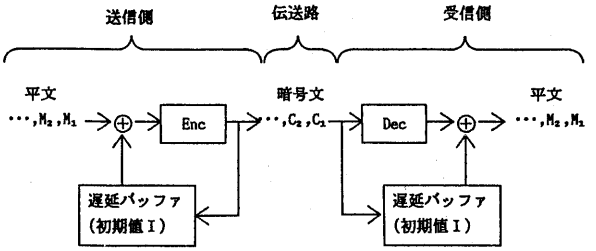


図6　CBCモードの構成

CBCモードによる暗号化を、例えば、10ブロックずつ繰り返すと640ビット単位のブロック暗号に拡張されるので、その分統計的推定は困難となる。

### 5.3　平文－暗号文の組が与えられたときの構造解析的アタック

平文－暗号文の組の入手の仕方としては、次の二通りがある。

(1)　解析者は自由に平文を与えられない場合。

(2)　解析者は自由に平文を与えることができ、かつ、対応する暗号文を得られる場合。

上記(1)または(2)の場合、5.1節で述べたしらみつぶし探索方法以外に暗号鍵を推定する方法として、直接次の方程式を解くことが考えられる。

$$C = f(K, M) \quad \cdots\cdots\cdots\cdots(9)$$

ここに、C ：暗号文･･･入力データ

K ：暗号鍵･･･未知数

M：平文　･･･入力データ

もし、(6)式の暗号関数fがKに関し、線形であり、正則行列A(M)とベクトルU(M)が存在して次式のように表現できたとする。

$$C = A(M)\cdot K + U(M) \quad \cdots(10)$$

このとき、

$$K = A(M)^{-1}\cdot(C - U(M)) \quad \cdots(11)$$

と解かれてしまう。

今回、提案する暗号アルゴリズムについては、暗号関数は線形になっていないので、(10)、(11)式のアタックは成立しない。

上記(2)の場合、平文、暗号文、暗号鍵の三者のパラメータの相互関係において、感度解析を行なうことによるアタックが成立し得る。このアタックの概略は次の通り。

(a) (6)式の計算式を詳細に分析し、暗号鍵のある特定ビットが'0'のときと'1'のときの、平文ー暗号文の統計的挙動の相違を予測する。

(b) 解析者(アタッカー)は、暗号鍵は未知の暗号装置に対し、実際に種々の平文を入力し、対応する暗号文をいくつも得る。そして、平文ー暗号文の統計的性質を分析することにより、暗号鍵のある特定ビットが'0'であるか'1'であるかを統計的に推定する。

このような感度解析によるアタックにも耐えられるようにするためには、上記(a)の統計的挙動の予測が困難であるようにしなければならない。そのためには、暗号鍵のすべてのビットが、平文、暗号文のすべてのビットと均等に干渉しあうように設計することが望ましい。

解読不可能な暗号アルゴリズム（理想アルゴリズム）では、暗号文と平文・暗号鍵との相関が全く無く、暗号文から平文や暗号鍵を推定することはできない。また、平文や暗号鍵を変化させた時の暗号文の各ビットが変化するかどうかもわからない。このとき、暗号文の各ビットが変化する確率は1/2である。

ここで、平文や暗号鍵を変化させたとき、暗号文が1ブロック当り何ビット変化するかという統計をとると、各ビットが変化する確率がそれぞれ1/2で、1ブロックが64ビットだから、変化ビット数は二項分布B(64,1/2)に従って分布する。また、64個の暗号ブロックのうち、第1ビット、第2ビット…が変化した回数も二項分布B(n,1/2)に従う。

しかし、実際の暗号アルゴリズムでは、暗号文と平文、暗号鍵との相関が全く無いということは有り得ない。そこで、暗号アルゴリズムの安全性の評価を、どの程度理想アルゴリズムに近いか、つまり平文・暗号鍵を変化させたときの暗号文の変化ビット数の分布がどの程度二項分布に近いかということにより行う[6]。

Multi2、DES、FEAL-8の三種類の暗号アルゴリズムについてこの評価を行った。評価結果を表2に示す。評価は、平文、暗号鍵を変化させたときの、ブロック内の変化ビット数、および第iビットが変化した回数について、標本数4096および32768の場合について行った。なお、DESは処理速度が遅いため標本数32768の場合の評価は行なっていない。

評価結果より、Multi2およびFEAL-8は、理想アルゴリズムとほとんど変わらないことがわかる。理想アルゴリズム以上の数値となる部分は、統計上の検定誤差であると考えられる。

## 6．圧縮暗号への応用

メッセージ通信において、メッセージ認証の機能を実現するため圧縮暗号が用いられる。

表2　ランダム度の比較

単位　%

| | | | Multi2 | DES | FEAL-8 | 理想アルゴリズム |
|---|---|---|---|---|---|---|
| 標本数4096 | ブロック間の相関 | 鍵指標 | 96.7 | 95.5 | 96.6 | 96.5 |
| | | 平文指標 | 96.5 | 95.7 | 96.9 | |
| | ビット間の相関 | 鍵指標 | 96.4 | 94.4 | 96.6 | |
| | | 平文指標 | 96.3 | 95.6 | 96.3 | |
| 標本数32768 | ブロック間の相関 | 鍵指標 | 98.8 | — | 98.8 | 98.8 |
| | | 平文指標 | 98.8 | — | 98.8 | |

圧縮暗号とは、あるメッセージを、一定長の短いデータに暗号化したものである。つまり、圧縮暗号の関数をHとすると、

$$h = H(M) \quad \cdots\cdots\cdots\cdots\cdots\cdots (12)$$

と書け、通常、hのデータ長はメッセージMのデータ長より短い。さらに、ハッシュ関数は、同じ圧縮暗号hを出力データとして得られるような入力データとして異なったメッセージ$M_1$、$M_2$を見つけることは計算量的にたいへん困難なものでなければならない。圧縮暗号は、別名、ハッシュトータル、あるいは、一方向暗号、メッセージ認証コード(MAC: Message Authentication Code)とも呼ばれている。

上記の条件を満たすハッシュ関数の例を図7および図8に示す。図で、Encと書かれているところが、今回開発した暗号関数の部分Multi2である。

圧縮暗号の速度は、図7の場合、暗号通信の速度とほぼ同等であり、図8の場合、鍵スケジュールの処理が追加されるので、暗号通信の速度に比べ、ほぼ1/2となる。

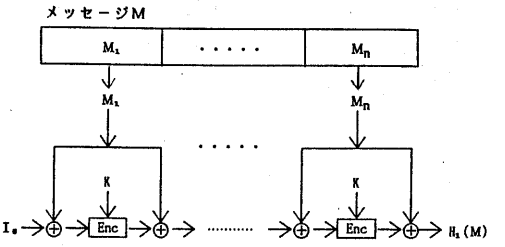


図7　ハッシュ関数の例（その1）

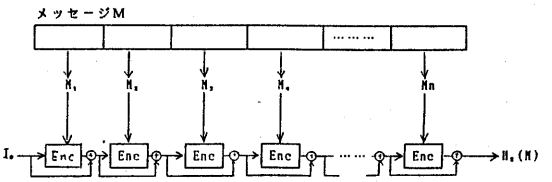


図8　ハッシュ関数の例（その2）

## 7．むすび

Multi2はDESの約65倍、FEAL-8の約2倍の処理速度を有し、二項分布への近似度に基づく安全性の評価に関してもDES、FEAL-8と比較して劣るという結果は得られなかった。このことからMulti2は、処理速度、安全性共に、有用な暗号アルゴリズムであると考えられる。特に、Multi2は32ビットコンピュータによる処理に向いており、ワークステーションや大型コンピュータのソフトウェアにより暗号化を行なう場合、CPUへの負荷が比較的少ないという長所がある。


## 謝辞

本研究の機会を与えて下さった㈱日立製作所システム開発研究所所長堂免信義氏をはじめ、本研究を行うにあたり御指導、御協力頂いた関係者各位に深謝申し上げる。


## 参考文献


(1)一松信監修、「データ保護と暗号化の研究」、日本経済新聞社、昭和58年、pp 73-121.

(2) A. Shimizu, S. Miyaguchi, "Fast data encipherment algorithm FEAL," Advances in Cryptology, EUROCRYPTO-87, Springer-Verlag Publication, 1988, pp 267-278.

(3) 宝木、白石、佐々木、「ICカード利用の電子取引用認証方式」、電学論C、Vol 107-C、No. 1、昭和62年、

(4) J. B. Kam, G. I. Davida, "Structured design of substitution-permutation encryption networks," IEEE Trans. Computer, Vol C-28, No. 10, October 1979, pp 747 - 753.

(5) Dorothy E. Denning, "Cryptography and data security," Addison - Wesley Publishing Company, January 1983.

(6) 宮口、平野、「暗号/認証アルゴリズム強度評価指標」、信学論、Vol J69-A、No. 10、昭和61年、pp 1252- 1259.