2S-07

FPGA ベース並列マシン RASH における TMTO 法暗号解析の実装 (1)
～実装手法～

浅見廣愛†，飯田全広‡，中島克人†，森伯郎†，佐藤裕幸†，高橋勝己†
†三菱電機（株），‡三菱電機エンジニアリング（株）

1 はじめに

タイムメモリトレードオフ解読法(Time-Memory Trade-Off Cryptanalysis，以下 TMTO 法と略す)[1]は鍵探索時の計算量が全数探索より少なく、記憶量がテーブルルックアップ法より少ないという特徴を持つ有効な暗号解読法である。我々は専用 LSI を用いた TMTO 法による暗号解析装置の提案[2]を行ってきたが、TMTO 法では事前計算フェーズと鍵探索フェーズで異なるロジックが要求されるため、これらの機能を包含する LSI を構成する必要があった。

一方、我々は FPGA(Field Programmable Gate Array)を多数用いた可変構造型計算機として、FPGA ベース並列マシン RASH(Reconfigurable Architecture based on Scalable Hardware)を試作し[3]、DES(Data Encryption Standard)を始めとする秘密鍵暗号の鍵探索処理が高速に行えることを実証した[4]。

FPGA は回路を変更できるという特徴を持つため、フェーズ毎に回路機能を変更する TMTO 法に適した LSI である。今回、CAM(Contents Addressable Memory)の代わりに SDRAM を搭載した RASH 用ドータカードを用いて、TMTO 法による DES 暗号の鍵探索を実現したので報告する。

2 TMTO 法暗号解析

2.1 TMTO 法の原理

TMTO 法は、鍵を入力値、暗号文を出力値とする関数 g を想定し、関数の出力値 g(X)から入力値 X を確率的な探索で求める方法である。この原理を示したのが図 1 であり、Y_0 を取得暗号文、X を求める鍵とした場合、以下の手順で X を求める。

(1) **テーブル作成：**初期値(A_0,B_0)を入力とした関数 g の出力値を再帰的に入力値として用いてテーブル（A_3,B_3）を作成する。

(2) **鍵探索：**図 1 のように Y_0 から Y_1,Y_2 を作成し、A_3,B_3 との一致検索を行う。一致したものがある場合、初期値から再度演算することにより X を求める。例えば、Y_1 と A_3 が一致した場合、X は A_1 になる。

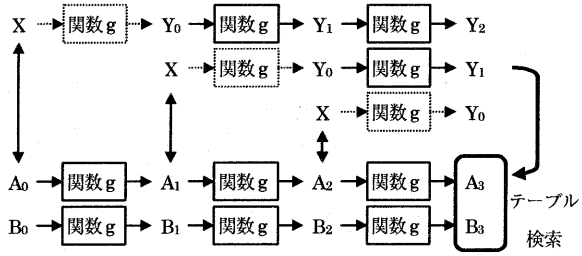

図 1 TMTO 法の原理

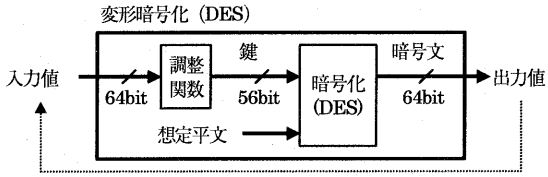

図 2 変形暗号化

実際の関数 g は、図 2 のように、平文を想定平文と称する 1 ブロック（DES の場合は 64bit）の入力として固定し、暗号化の前段に調整関数を加えた「変形暗号化」を関数として定義する。調整関数はビット長の調節と変換を行う関数であり、DES の場合は入力値 64bit を 56bit に変換する。

2.2 TMTO 法のパラメータ

TMTO 法においては、「変形暗号化」関数はランダム関数とみなすことができる。このため、探索表もしくは事前計算中の中間結果に全ての値が出現する保証はなく、探索成功確率は 100%にならない。また、出現する値の偏りを防ぐため、複数種類の調整関数を用いる必要がある。これらを考慮して、TMTO 法では次のパラメータを決定する。

まず、ASIC による専用マシンを提案した[2]と同じ k=2,u=1 という値を選択し、探索成功確率を 80%とした。これらの k,u を満たす L,M,T の値として、以下の値を選択した。

- $L=2^{22}$ （4M 枚の探索表＝4M 種類の調整関数）
- $M=2^{14}$ （探索表 1 枚は 16K エントリ）
- $T=2^{21}$ （探索表の 1 エントリを得るため、変形暗号化を 2M 回繰り返す）

3 RASH のハードウェア構成

図 3 に RASH のハードウェアの構成を示す。
RASH の基本構成（1 ユニット）は最大 6 枚の演

Implementation Time-Memory Trade-Off Cryptanalysis on FPGA-based Parallel Machine "RASH" (1) - the Implementation method -

Hiroai Asami, Masahiro Iida, Katsuto Nakajima, Hakuro Mori, Hiroyuki Sato, Katsumi Takahashi.
Mitsubishi Electric Corporation & Mitsubishi Electric Engineering Co.,LTD.
5-1-1 Ofuna, Kamakura, Kanagawa 247-8501, Japan

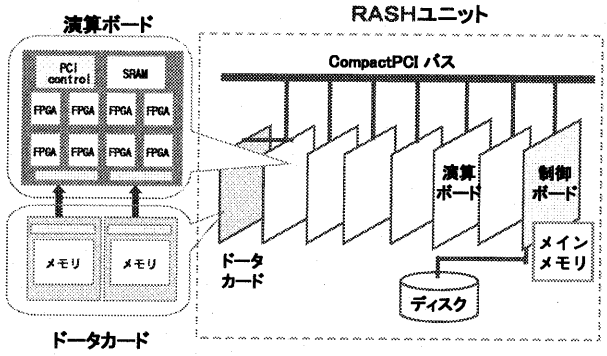

図 3 RASH の構成

算ボードと、メモリ搭載ドータカード（以降、単にドータカードと呼称）、制御ボード、ディスク等から成る。演算ボードは CompactPCI 基板上に 8 個の SRAM タイプの FPGA(10万ゲート相当)を搭載している。これらの FPGA は PCI バス制御用のコントローラにローカルバスで接続されており、さらに 32bit の信号線でメッシュ接続されている。また、演算ボード上には 2MB の SRAM が搭載されている。

演算ボード上にはドータカードを直接接続するためのコネクタが用意されており、2 枚までのドータカードが搭載できる。ドータカードには、128MB の SDRAM が 2 つ搭載され、2 個の FPGA で 1 つの SDRAM を共有する。

制御ボードは Pentium MMX(233MHz)を搭載した市販のボードであり、演算ボードへのデータの分配等を行う。ユニット内の制御ボード、各演算ボード間は CompactPCI バスで接続され、複数ユニット間は Ethernet で接続が可能である。

4 回路実装

実際の TMTO 法の処理は、テーブル作成を行う事前計算フェーズと、暗号文入手後に暗号化鍵の探索を行う鍵探索フェーズの 2 つに分けられる。各フェーズにおける FPGA 上の回路構成を図 4 に示す。

4.1 事前計算フェーズ

事前計算フェーズでは、事前計算回路で DES の F 関数 8 段パイプライン回路を用いて初期値を変形暗号化することによりテーブル作成を行う。事前計算フェーズではドータカードを使用しない。

4.2 鍵探索フェーズ

鍵探索フェーズでは、鍵探索回路で入手暗号文から再帰的に変形暗号化を行い、ドータカード上のテーブルとの一致検索を行う。ドータカード上のテーブルとのメモリバンド幅の制約から、FPGA 内には DES の F 関数 1 段回路を 2 個搭載することとした。

4.3 ヒットマップ

鍵探索フェーズでのテーブル検索は、CAM を用いずに、ドータカード上の RAM を用いる。本実装では、図 5 のようにテーブルからハッシュ表を作成

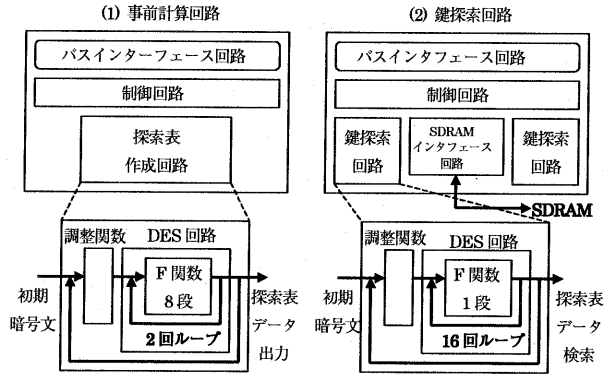

図 4 事前計算回路と鍵探索回路の構成

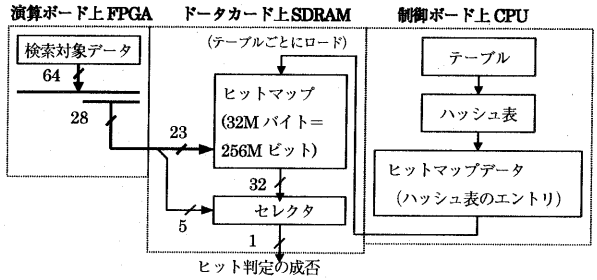

図 5 ヒットマップによる検索

し、ハッシュのエントリ表だけを「ヒットマップ」としてドータボード上の RAM に収め、ヒット判定のみを高速に行う。ヒット判定に成功したものは、制御ボード上の CPU を使い時間をかけて正規の検索を行う。

5 まとめ

以上、RASH を TMTO 法による DES 暗号解析に適用した場合の実装手法について報告した。今回の実装に関する性能評価については、文献[5]にて報告している。

今後は、これらの結果を踏まえて、RASH による TMTO 暗号解析の更なる性能向上の検討を行う予定である。

参考文献

[1] M.E.Hellman, "A cryptanalytic time-memory trade-off," IEEE Transaction on Information Theory, Vol.IT-26, No.4, pp-401-406, 1980.

[2] 高橋, 他: "タイムメモリトレードオフ解読法に基づく暗号強度評価装置の実現性について," 情報処理学会論文誌, Vol.40, No.8 pp-3318-3328, 1999-8.

[3] 中島, 他: "FPGA ベース並列マシン RASH の概要", 第 58 回情処全国大会, 1H-08, 1999-3.

[4] 浅見, 他: "FPGA ベース並列マシン RASH での DES 暗号解析処理の改良",情報処理学会論文誌：ハイパフォーマンスコンピューティングシステム, Vol.41,No.SIG 5(HPS 1), pp-50-57, 2000-8.

[5] 飯田, 他: "FPGA ベース並列マシン RASH における TMTO 法暗号解析の実装(2)－性能評価－", 第 62 回情処全国大会, 2S-08, 2001-3.