C-007

Stream cipher engine の最適設計

Optimum Design of a Stream cipher engine

大隅 裕介† 横山 温子† 深瀬 政秋† 佐藤 友暁‡
Osumi Yusuke Yokoyama Atsuko Fukase Masa-aki Sato Tomoaki

1.はじめに

至る所で形成されるユビキタスネットワークは、第三者による攻撃や侵入、盗聴や情報漏洩等の危険性の増大を引き起こす。そのため、低消費かつ信頼性の高い情報セキュリティ技術の必要性が大きな意味を持つようになった。そこで、著者らはユビキタスネットワークに貢献しうるマルチメディアストリーム暗号エンジン“Stream cipher engine”を開発してきた [1, 2]。

Stream cipher engine の開発目標は実用的な暗号強度でメディアストリームの小電力高速処理を可能とすることで、その要はストリーミング用バッファのサイズである。Stream cipher engine では register file をバッファとし、そのサイズに応じた乱数発生器を組み込む。Register file サイズを増せば暗号強度は増し、書き込み回数が減るのでスループットの向上につながる。一方、register file の増加は遅延と電力の増加を引き起こすことが予想される。そこで、本研究では Stream cipher engine の最適設計に取り組む。

2.Stream cipher engine

Stream cipher の基本原理は、図 1 に示すように RNG によって生成された擬似乱数を用いたランダムアドレッシングによる転置暗号方式である。RNG を 2 つにし、それぞれが発生させる乱数の周期を変えることで、二重鍵方式にした。2 つにすることで暗号強度の向上を狙う。

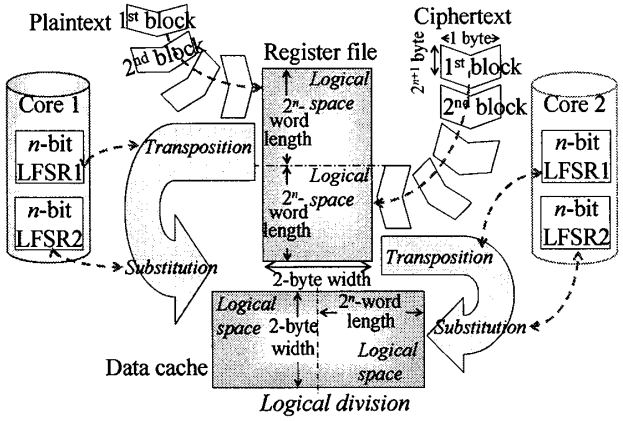

図 1 暗号化の仕組み

暗号化の場合、RNG から出力された乱数と Register File のアドレスを同期させ、転置暗号化し Hidable Unit により更に転置暗号化する。同様に復号化の場合は、RNG から出力された乱数と Data Cache のアドレスを同期させ、Hidable Unit による転置復号化する。Register File に格納する際の乱数と Hidable Unit で転地する際の乱数はそれぞれ独立した 2 つの RNG から異なった周期の乱数にした。

3.SIMD (Single Instruction/Multiple Data)

音声や画像などのマルチメディア・データに対する処理は、固定的なフォーマットのデータに対して、同じ種類の演算を繰り返し適用することが多い。そこで、1 つの命令で多量のデータに対して同じ種類の演算を一斉に行うようにして、プロセッサ全体のデータ処理能力を高めるために用意されたのが SIMD 命令である。

SIMD 命令を実装するためには、データを格納するための比較的大量のレジスタ（データ供給が滞らないようにするため）が必要な為、今回の Stream cipher engine には十分なレジスタを実装させ、SIMD 命令を実現させた。また、パイプライン処理と組み合わせることで処理能力拡大を狙った。

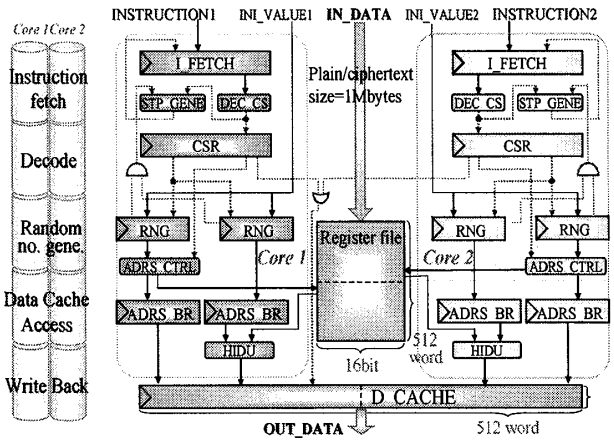

図 2 ストリーム暗号エンジンの構成

4.ストリーム暗号エンジン

図 2 にストリーム暗号エンジンの全体構成を示す。ストリーム暗号エンジン“Stream cipher engine”は 2 つの独立した CORE と共通の Register File、Data Cache からなり、各

† 弘前大学 Hiroskai University
‡ 弘前大学情報処理センター Computer and Network Systems Center, Hirosaki University

CORE はパイプライン 5 段で構成される暗号・復号専用プロセッサである。入力するデータは同じ初期値且つ Stream cipher engine により暗号化されたデータを用いる。RNG による擬似乱数生成に必要な初期値は公開鍵方式を用いて相手に送信する。なお、暗号化には共通鍵方式を用いる。

本研究における、最適設計とは図2においてRegister FileとData Cacheの最適容量を探ることである。バッファとしてのRegister Fileの容量を増やすことで暗号強度が増す。しかし面積の増大から、消費電力が増え、クロックスピードは減る事になる。従って、これらの因子のトレードオフに配慮した最適設計が必要である。

5.評価

最適設計の指針を得るために、Register File、Data Cache、消費電力、クロック周波数、面積の評価を行う。

評価の際、Register File size、Data Cache size を 32-word、64-word、128-word、256-word、512-word、とした Stream cipher engine1-5 を作成しそれぞれの消費電力、面積、Throughput、クロック周波数、Running Time を測定する。図 3 に Clock、図 4 に消費電力、図 5 に Running time、アクセス回数と Resister File size のグラフを示す。

それぞれを比較評価し、その結果から面積を 2.5×5.0 角チップとして作成する際の適する Register file size、Data Cache size を探る。

7.結び

最適設計として Stream cipher engine の RNG を 2 つにすることで二重鍵にし、その評価を行った。そしてトレードオフポイントを探った。

今後の課題として、暗号強度の測定を行い、それを考慮したトレードオフポイントを見つけること、そして、動作シミュレーションを行うことがあげられる。

8.謝辞

本研究は東京大学大規模集積システム設計教育研究センターを通し、シノプシス株式会社の協力で行われたものである。

9.参考文献

[1] 岩本祐顕、天間僚、武田宏樹、野田一訓、深瀬政秋、佐藤友暁、「マルチメディアストリーム暗号エンジン」平成 19 年度電気関係学会東北支部連合大会, p194, 2007 8 月.

[2] M. Fukase and T. Sato, “A Stream Cipher Engine for Ad-hoc Security,” Proc. of CIS’2007, pp. 902-906, 2007.

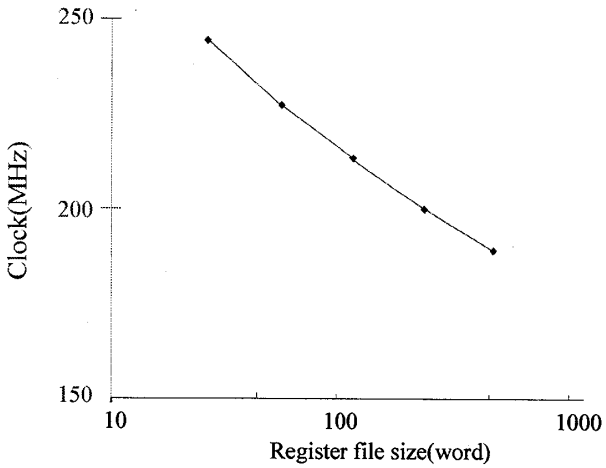

図 3 Register file size vs Clock

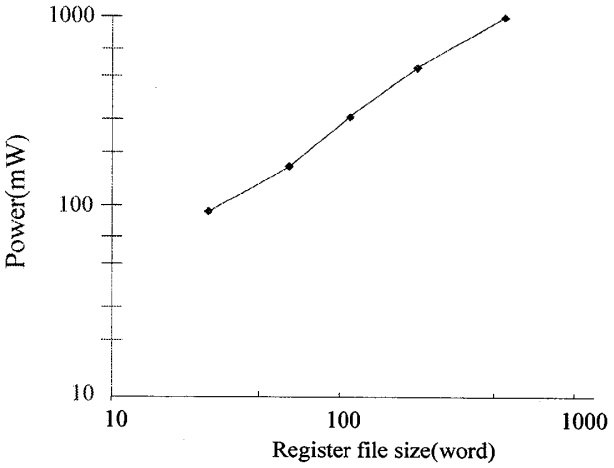

図 4 Register file size vs Power

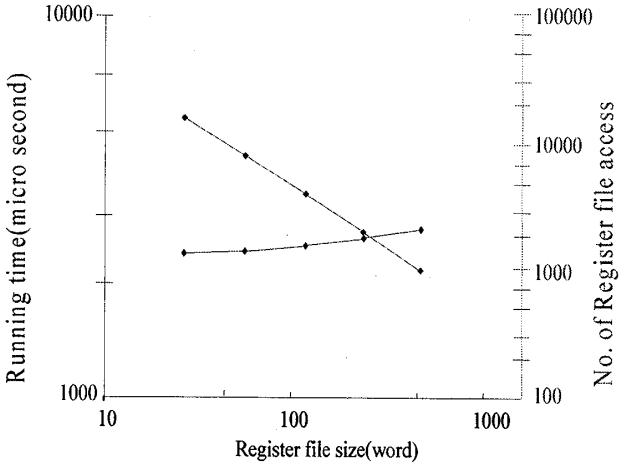

図 5 Register file size vs
Running time, No. of Register file access