Technical Reports on Information and
Computer Science from Kochi
Vol. 1 (2009), No. 9

暗号化アルゴリズム DES の FPGA化による性能評価

山口 良典　松永 惇弥　村岡 道明
高知大学理学部情報科学コース　村岡研究室

1. まえがき

高速化の見通しが出来た暗号化アルゴリズム DES を FPGA に実装することで製品化に向けた性能評価を行った。また、更なる高速化を実現するために、暗号化アルゴリズム DES のパイプライン化についても検討した。

2. 研究目的

高速化の見通しがついたハードウェア DES を FPGAに書き込む為に、FPGA化を行う際の課題を明確にし解決する。高速化を目的としたパイプライン化については、従来研究室で行われていたパイプライン化手法とは異なった、新たなパイプライン化手法で高速化を行う。

3. 研究内容

3.1 ハードウェア化部分の検討

図１．ハードウェア化の検討

従来と同様に暗号化処理部のハードウェア化を行い、その次に処理時間の多い鍵生成部についても検討した。鍵生成部については、鍵データを内部に保存し、FPGAのスイッチ操作で呼び出すという手法を用いれば、FPGA化は必要ないと考えていたが、ビット数が多い初期値をスイッチにより一度に与えるという方法は、論理合成で失敗してしまい、出力がGNDから出力されてしまうという不具合が起こった。その結果、本手法では新たに鍵生成部の FPGA 化を行うことを決めた。

3.2 FPGA化

- 課題

今回の FPGAは入力に最大 16bit しか割り当てられず、入力データを一度に与えることが出来ないという問題がある。暗号化処理部のハードウェアのアルゴリズムを下に示す。

- 解決方法

入力データ 64bit と鍵データ 64bit を 16bit ずつに分け、スイッチの切り替えで入力を行った。

入力・鍵データがすべて入力されたら暗号化開始という回路を作成した。

4. シミュレーション結果

表１．タイミングシミュレーション比較

周波数[MHz]	20MHz	40MHz	50MHz	80MHz
暗号処理16回の実行時間[nsec]	89762.391	44887.865	35913.643	22447.780
暗号化処理16回のサイクル数	1796サイクル	1796サイクル	1796サイクル	1796サイクル
DES暗号16回の実行時間[nsec]	89961.761	44987.934	35980.93	22487.5
DES暗号16回のサイクル数	1800サイクル	1800サイクル	1800サイクル	1800サイクル

本手法が従来手法より時間が掛かっていることが分かる。これは、従来手法に比べて入力に時間が掛かるためで、従来手法は鍵データ生成のため FIFOを用いて鍵データを生成していたが、今回用いた FPGAで従来手法を行うには、FIFOへの入力で２サイクル、更に暗号化対象データの入力に４サイクル掛かるので、結果的に 1796+2+4 = 1802 サイクル必要ということになる。つまり、FPGAに実装するという視点から考えると、サイクル数は従来手法より２サイクル速くなる。

5. パイプライン化

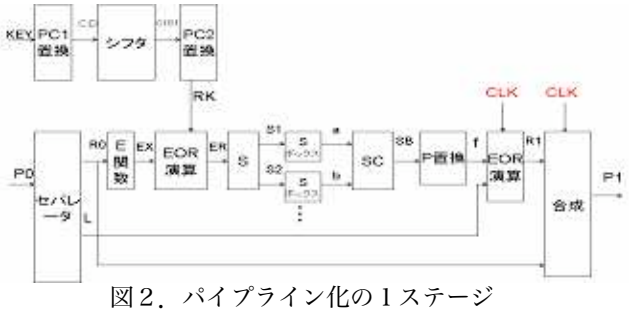

図２．パイプライン化の１ステージ

従来の DES は 16 回の繰り返しのために、３ビットカウンタと４ビットカウンタを使っているため、多くの繰り返し制御のレジスタが必要となっている。しかし今回のパイプライン化では、鍵生成部と暗号化処理部を 16 段用意することにより、記憶させておく必要がないため、大量のレジスタが削減することが出来る。その結果、全体の実行時間が飛躍的に向上する。また DES に関らず、暗号化処理は一度に複数を暗号化するので、この手法は適しているといえる。

6. 結果

表２．処理時間比較

	動作周波数	暗号化処理30万回のサイクル数	暗号化処理30万回の処理時間	処理速度の比較
ソフトウェア処理	—	—	15.613sec (a)	1 (a/a)
従来のハードウェア化	80MHz	33600004	0.42sec (b)	37.2倍 (a/b)
本研究のパイプライン化	10MHz	300,015	0.03sec (c)	614.3倍(a/c) 14倍(b/c)

処理時間は 0.03 秒となり、これはソフトウェアでの処理と比べ 614 倍、従来のハードウェア化と比べると 14 倍の高速化を行った。

7. まとめ

暗号化アルゴリズム DES の FPGA 化を行った。従来提案されていた暗号化処理部に加え、鍵生成部を付け加えた。今回 FPGA には入出力に制限があり、新たなモジュールを付け加えるという対応をとった。パイプライン化については、従来のパイプライン化とは異なった新たな手法でアプローチすることで、10MHz においてソフトウェアに比べ 614.3 倍、ハードウェア化に比べ 14 倍という飛躍的な高速化を行うことが出来た。今後の課題として、10MHz より高い周波数に対応できるように改良することが課題として挙げられる。また、入出力方法に関しても、PC から入力を与えるなど、パイプライン化に適した方法を検討していく必要がある。