CAMELLIA H/W 実装評価報告

東芝

2001-01-12

目次

1　はじめに

本ドキュメントは, ブロック暗号アルゴリズム Camellia のハードウェア実装評価について記述したものである.

Camellia は, 256/192/128 ビットの暗号化鍵を持つ 128 ビットブロック暗号であるが, 評価は, 256 ビット鍵の暗号化回路に対して行った.

評価は, verilog で設計し, 同一の記述に対して合成条件で速度優先の条件と面積優先の条件で合成することにより行った。

2　ハードウェア設計の方針

設計方針を以下に示す.

1. ループアーキテクチャを採用する.
2. パイプラインアーキテクチャは採用しない.
3. 置換表はテーブルを用い, 論理合成ツールを用いて最適化を行う.
4. コントロール回路は鍵生成及び暗号化回路の全てのタイミング信号を生成する.

図 1に評価アルゴリズムを示す. 左側が暗号化アルゴリズムで, 右側が鍵スケジュールアルゴリズムである. 図中において, F は, F 関数, FL は, FL 関数, FLi は, FL^{-1} を表している.

図 1において, 点線は, 1 段のループで処理される境界を示している. 鍵長が 256 ビットの場合は, 28 クロックで暗号化処理が完了し, 6 クロックで鍵スケジュールが完了する.

各ループでの処理内容を表 1に示す.

3　回路構成

本評価に使用した評価回路の構成を図 2に示す.

評価回路は, 暗号化回路, 鍵スケジュール回路, コントロール回路で構成される.

- 暗号化回路

  暗号化回路は, 一段分の暗号化のための回路である.

  暗号化回路は, 入力された平文を保持する入力レジスタ (Input(M) Register), 暗号化された暗号文を保持する出力レジスタ (Output(C) Register), 暗号化を実現する組み合わせ回路 (Encryption Logic), ループの処理結果を一時的に保持する一時レジスタ (Temporary Register) で構成される.

- 鍵スケジュール回路

  コントロール回路からの制御信号に従い副鍵を生成する回路. 256 ビット鍵の場合は, 6 クロックで K_A, K_B の生成を完了する.

  鍵スケジュール回路は, 入力された鍵を保持する入力レジスタ (Input(KR/KL) Register), 鍵スケジュールの中間結果を保持するレジスタ (KA/KB Register), 鍵スケジュールを実現する組み合わせ回路 (KeyScheduling Logic), 鍵スケジュールのループの処理結果を一時的に保持する一時レジスタ (Temporary Register) で構成される.

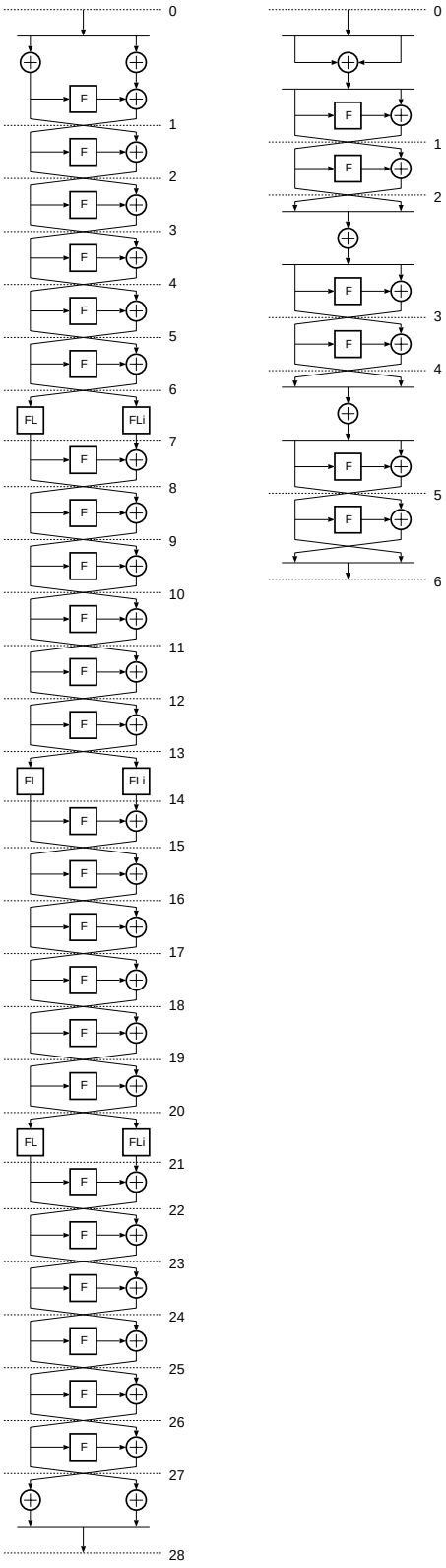
0
F
1
2
3
4
5
6
FL
FLi
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28

図 1: 評価アルゴリズム

表 1: 各ループにおける処理内容

1	XOR + F- 関数
2	F- 関数
3	F- 関数
4	F- 関数
5	F- 関数
6	F- 関数
7	FL- 関数
8	F- 関数
9	F- 関数
10	F- 関数
11	F- 関数
12	F- 関数
13	F- 関数
14	FL- 関数
15	F- 関数
16	F- 関数
17	F- 関数
18	F- 関数
19	F- 関数
20	F- 関数
21	FL- 関数
22	F- 関数
23	F- 関数
24	F- 関数
25	F- 関数
26	F- 関数
27	F- 関数
28	XOR

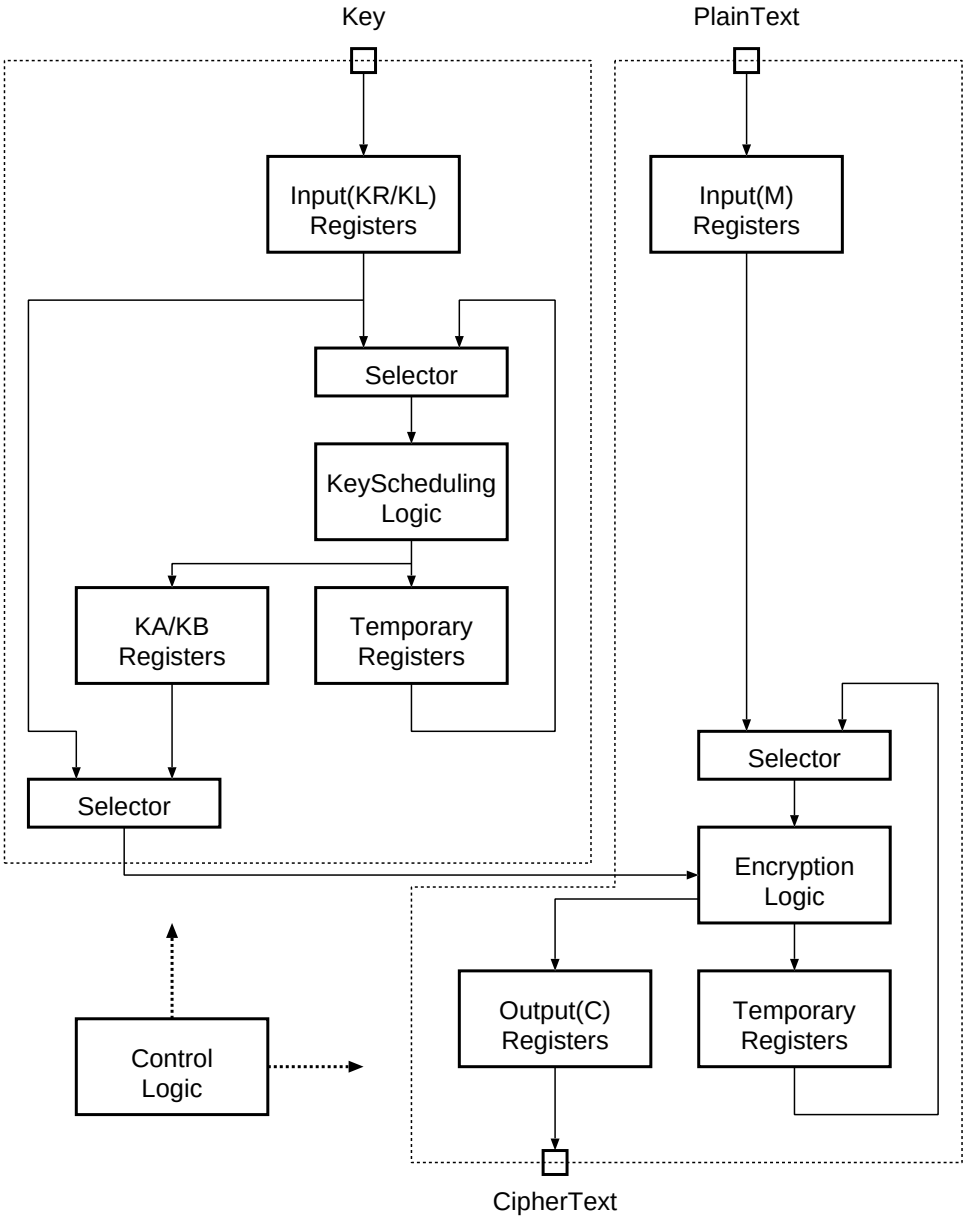
Key
PlainText
Input(KR/KL)
Registers
Input(M)
Registers
Selector
KeyScheduling
Logic
KA/KB
Registers
Temporary
Registers
Selector
Selector
Encryption
Logic
Output(C)
Registers
Temporary
Registers
Control
Logic
CipherText

図 2: 評価回路

- コントロール回路

  鍵生成の制御信号やタイミングの制御信号を生成する.

表 1に示した動作タイミングを図 3に示す。

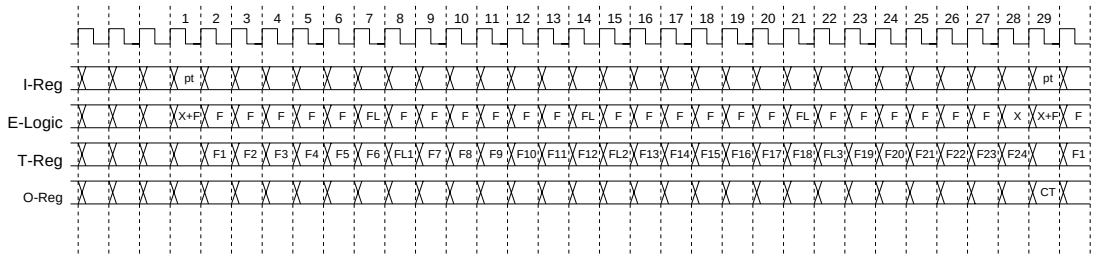

図 3: 評価回路の動作タイミング

図中において, I-Reg は, 暗号化回路の入力レジスタの出力, E-Logic は, 暗号化回路の暗号化ロジックの処理内容, T-Reg は, 暗号化回路の一時レジスタの出力, O-Reg は, 暗号化回路の出力レジスタの出力である.

'1' のタイミングで, I-Reg に平文 pt が保持され, その値に X+F(XOR と F 関数) の処理が行われ, その結果 F1 が次のクロックの立ち上がりで一時レジスタに保持される. '2' のタイミングで, 一時レジスタに保持されている値 F1 に対して F(F 関数) の処理が行われ, その結果 F2 が次のクロックの立ち上がりで一時レジスタに保持される. 同様に処理が進められ, '29' のタイミングで, 一時レジスタに保持されている値 F24 に対してその値に X(XOR) の処理が行われ, その結果の暗号文 CT が次のクロックの立ち上がりで出力レジスタに保持される. このような動作によって暗号化処理が行われる.

4　実装結果

Camellia のハードウェア実装評価結果を示す.

以上の設計方針のもと, Camellia を ASIC で実装した場合について評価を行った. 評価条件を表2に示す. 合成は, 速度優先の条件と面積優先の条件の 2 つで行った.

表 2: ハードウェア設計及び評価環境

記述言語	Verilog-HDL
シミュレータ	VCS5.1
デザインライブラリ	TOSHIBA 0.25 μ COMS　ASIC Design Library
論理合成ツール	Design Compiler. 2000.05-1
動作条件	0 〜 70℃, 3.3V ± 5 %

4.1　ハードウエア規模

ハードウェア規模の評価結果を以下に示す.

表 3: ハードウェア実装結果 (ハード規模) 単位： Gate

暗号アルゴリズム	暗号回路	鍵スケジュール	コントロール回路	TOTAL
Triple-DES *1	4,218	1,333	151	6,496
Triple-DES *2	2,011	1,088	134	5,111
Camellia *1	16,327	22,755	266	39,348
Camellia *2	9,668	13,304	141	23,124

*1 スピード優先にて論理合成

*2 規模優先にて論理合成

4.2　処理速度

スループットの評価結果を以下に示す.

クリティカルパスは, 1 クロックサイクルの最大遅延時間を表している. したがって, 暗号化に要する処理時間は, クリティカルパス×処理クロック数となり, スループットは, 暗号化のブロックサイズを処理時間で割った値となる. Triple-DES では, 処理に 59 クロック要し, Camellia では, 処理に 28 クロック要する.

なお, 数値は全て, ワースト条件での値である.

表 4: ハードウェア実装結果 (スループット)

暗号アルゴリズム	クリティカルパス	暗号化処理時間	スループット
Triple-DES *1	4.44ns	262.0ns	244Mbps
Triple-DES *2	7.10ns	419.9ns	153Mbps
Camellia *1	5.46ns	152.9ns	837Mbps
Camellia *2	11.51ns	322.3ns	397Mbps

*1 スピード優先にて論理合成
*2 規模優先にて論理合成

5　まとめ

本稿では, ハードウェア実装に対する評価結果を示した.

ハードウェア規模に関しては, Camellia は, Triple-DES に比べ 4.5 倍から 6.1 倍程度大きくなった. また, データスケジュール部と鍵スケジュール部では, 鍵スケジュール部の方が大きくなる. これは, 鍵スケジュール部がデータスケジュール部と同じ構成要素を持ち, かつ, 鍵の選択回路の規模が大きくなるためである.

スループットに関しては, Camellia は, Triple-DES に比べ 2.6 倍から 3.4 倍高速になった.