平成２２年度　特別実験報告書概要

課程，学籍番号，氏名	課程：　電気・電子工学課程，学籍番号：　B093341，氏名：古森　篤朗
工学分野名：情報通信システム	指導教員名：市川　周一　准教授（７系）

題　　目：和　ハッシュ関数 Luffa のハードウェア記述言語による実装と FPGA への最適化

（英 The VHDL implementation of hash function Luffa, and its optimization for FPGA）

Abstract

Hash function is a procedure that converts variable-sized data to a fixed-length value. Hash functions are widely used for digital signature, check-sum, and timestamps. In 2007, National Institute of Standards and Technology opened a public competition for new cryptographic hash algorithm SHA-3. On September 2010, the second round of the competition is completed, and 14 candidates are remaining. In this study, a function called Luffa was implemented in VHDL, which was optimized for Xilinx FPGA.

First of all, the reference VHDL design was implemented as a literal translation of the reference C code provided by the authors of Luffa. Then the optimized C code by the authors of Luffa was also implemented in VHDL. This implementation resulted in 1% reduction of frequency and 7% reduction of area.

Further optimization was attempted using Xilinx Unified Library, which is a collection of design elements optimized for Xilinx FPGA devices. By the optimization of Message Injection module, 10% reduction of frequency and 8% reduction of area were achieved. Optimization of Sub Crumb module achieved 38% increase of frequency and 1% increase of area. Optimization of Mix Word module achieved 30% increase of frequency and 10% reduction of area. Optimization of Constant Generator module achieved 8% reduction of frequency and 11% reduction of area. Though the combinations of these optimizations were also examined, it did not lead to further improvements.

概　　要

ハッシュ関数とは，可変長の入力データから固定長の値を生成する関数である．文書の署名，改竄の検出，タイムスタンプなどに広く用いられている．2007 年，アメリカの国立標準技術研究所は次世代の標準ハッシュ関数 SHA-3 の募集を始めた．2010 年 9 月現在，選考の第２ラウンドまでが終了しており，14 の候補が残っている．この中から Luffa という関数を VHDL で実装し，Xilinx 社の FPGA 用に最適化した．

まず，Luffa の製作者が提供している C 言語ソースコードを参考に，VHDL で実装した．さらに Luffa 製作者が提供する Optimized 版 C コードを参考に VHDL を改良し，論理合成を行ったところ，動作周波数は 1%減少し，面積は 7%減少した．

次に，Xilinx 社から提供されるデバイス依存ライブラリにある ROM, RAM, LUT 等のコンポーネントを使用し，モジュールごとに最適化を行った．Message Injection モジュールに改良を施したところ，動作周波数は最大で 10%減少し，面積は最大 8%減少した．Sub Crumb モジュールでは，動作周波数は最大 38%増加し，面積は最大 1%増加した．Mix Word では，動作周波数は最大 30%増加し，面積は最大 10%減少した．Constant Generator では，動作周波数は最大 8%減少し，面積は最大 11%減少した．これらの改良を組み合わせた設計も評価したが，面積と動作周波数の向上は得られなかった．

同じ VHDL ソースコードを，Virtex-4 FPGA で評価した．Optimized 版の動作周波数は 2%減少し，面積は 5%減少した．Message Injection モジュールでは，　動作周波数は最大 2%増加し，面積は最大 23%増加した．Sub Crumb では，動作周波数は最大 28%減少し，面積は最大 3%減少した．Mix Word では，動作周波数は最大 2%増加し，面積は最大 7%減少した．Constant Generator では，動作周波数は最大 2%増加し，面積は最大 1%減少した．提案した改良手法のうち，opt，MI_lut，MI_lut_l，MI_ram，CG_lut，CG_lut_l，CG_ram，CG_rom，SC_lut，SC_lut_l，SC_rom，MW_rom を組み合わせて合成を行ったところ，面積は個々の改良における最小面積よりも 5%小さくなった．動作周波数は，個々の改良における最高動作周波数と同じであった．

発表する際の課程を記入　　電気・電子工学　　課程　　発表番号　54

（学籍が他課程所属の学生も発表する課程を記入すること）