システムＬＳＩ設計技術　108－2
(2003. 1. 28)

Reconfigurable-processor core の実装実験とその評価

上野 貴史　志賀 裕介　今中 晴記　工藤 健慈　関根 優年

東京農工大学工学研究科電子情報工学専攻

〒184-8588　東京都小金井市中町 2-24-16

E-mail : { ueno , you , imanaka , k2x , sekine }@sekine-lab.ei.tuat.ac.jp

あらまし：ハードウェアの高速性とソフトウェアの柔軟性を併せ持った、再構成可能なプロセッサの実装実験とその評価をした。プロセッサコア部と拡張回路を別々の FPGA に実装し、拡張回路には Haar 関数による離散 Wavelet 変換を行う回路を用いた。再構成可能プロセッサの拡張方法、実装実験手法、及び拡張回路の設計法について紹介する。

キーワード：リコンフィギュラブル、プロセッサ、FPGA、ウェーブレット変換

Implementation and evaluation of reconfigurable-processor core

Takashi UENO , Yusuke SIGA , Haruki IMANAKA , Kenji KUDO , and Masatoshi SEKINE

Department of Electrical and Electronic Engineering,
Tokyo University of Agriculture & Technology

2-24-16 Nakamachi, Koganei, Tokyo, 184-8588, Japan

E-mail : { ueno , you , imanaka , k2x , sekine }@sekine-lab.ei.tuat.ac.jp

Abstract : We experiment reconfigurable-processor core that combines the rapidity of hardware and the pliability of software and its evaluation. Processor core and extended circuit implement different FPGA. Extended circuit implement an haar wavelet transform circuit. We introduce how to extend reconfigurable-processor, experiment of implementation, and design of extended circuit.

Keyword : reconfigurable, processor, FPGA, Wavelet transform

1．はじめに

システムの機能を実現する方法として主に２つがあげられる。汎用プロセッサを用いてソフトウェアで実現する方法と、専用のハードウェアを用いて実現する方法である。前者は、複数の機能を実現する場合や、システムの変更時などにハードウェアを変更することなく柔軟に処理することができる。一方後者は、特定用途向けに設計されているため、特定の処理を高速に行うことができる。

今日、両者の長所を活かした再構成可能なプロセッサが注目されている。その方法として、演算回路を再構成可能部分に構成する[1]ものや、LSI 内に多数のプロセッサを用意し、その間の接続をクロックサイクル毎に変更する[2]ものや、多数の演算処理回路を実行する処理命令ごとにネットワークトポロジーを変えハードを再構築する[3]ものや、UCI 拡張、DSP 拡張、ハードウェア・エンジン拡張、VLIW コプロセッサ拡張の４種類の拡張方法がある[4]ものなどがある。

本稿では、基本的な処理を行うプロセッサコア部分と、拡張処理を行う拡張部分を、再構成可能な LSI である FPGA にそれぞれ実装し、その評価をする。拡張処理部分には、画像処理などの前処理として用いられる、Haar の離散ウェーブレット変換を行う回路を実装する。ウェーブレット変換を行う回路として、部分的に画像変換を行う Haar-Wavelet 変換チップ[5]などがある。

2．再構成可能プロセッサ

再構成可能プロセッサは、ジャンプや分岐、アドレス計算など柔軟性が求められる処理をプロセッサコア部分で実行し、複雑な演算など高速性が求められる処理を拡張回路で実行することを目的としている。

2．1　プロセッサコア

プロセッサコアの主なアーキテクチャーを以下にあげる。

- RISC 型プロセッサであり、すべての命令を１クロックサイクルで実行する
- 命令長、データ幅は共に３２ビット
- 汎用レジスタは３２本
- 命令数は４３個（基本３９個＋拡張４個）※表１
- 命令形式は３種類であり内部回路が複雑にならないようにする
- ５段のパイプライン構造（フェッチ、デコード、実行、メモリアクセス、ライトバック）

表１　命令セット

ロード命令	LB,LBU,LH,LHU,LW
ストア命令	SB,SH,SW
ALU 命令 （イミディエト） （レジスタ）	 ADDI,SLTI,ANDI,ORI,XORI,LUI ADD,SUB,SLT,AND,OR,XOR,NOR
シフト命令	SLL,SRL,SRA,SLLV,SRLV,SRAV
ジャンプ命令	J,JAL,JR,JALR
分岐命令	BEQ,BNE,BLEZ,BGTZ, BLTZ,BGEZ,BLTZAL,BGEZAL
拡張命令	KD,KDR,KE,KER

2．2　プロセッサコアと拡張回路の接続方法

プロセッサコアと拡張回路の接続方法を２つ用意している。２つの拡張方法とも、プロセッサコア部分と拡張回路部分は、同じ周波数で動作し、位相も同位相で動作する。これによってパイプラインを崩すことなく動作させることができる。

拡張方法の１つは、図１に示すように実行ステートとメモリアクセスステートの一部を拡張回路で動作させる方法である。これは KE,KER 命令に相当する。

動作の流れは、命令メモリより読み込まれたインストラ

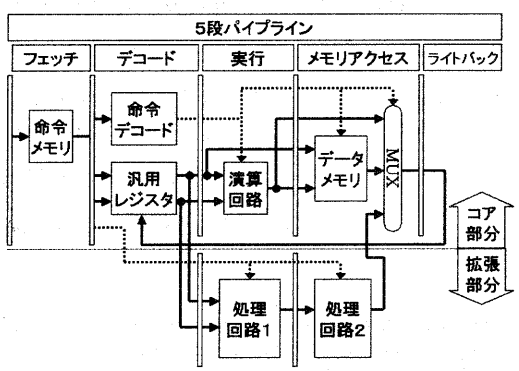

図１　拡張回路１の接続

クションを、デコーダーが６ビットのOP（オペコード）をデコードし、KE,KER 命令の時、オペランド（５ビット、５ビット）で指定された汎用レジスタの値（３２ビット）２つを拡張回路に送り、それと同時に残りのオペランド（即値・１６ビット）を拡張回路の制御用に拡張回路に送る。拡張回路は、実行ステートのクロックの立ち上がりでこれらの信号を読み込む。読み込まれたデータはユーザー任意の処理回路（図の処理回路１に相当）により実行され、メモリアクセスステートのクロックの立ち上がりで処理結果が保持される。この保持された処理結果（３２ビット）をプロセッサコアに送る（図の処理回路２は通過）。そして、KER 命令の時、ライトバックステートのクロックの立ち上がりでプロセッサコアの汎用レジスタに拡張回路の処理結果が書き込まれる。

拡張回路はユーザー任意の回路であるため、その自由度は広く、図の処理回路１で処理したデータをさらに処理回路２で処理してからプロセッサコアに送る方法や、クロックの立下りを利用することもできる。

接続方法のもう１つは、図２に示すようにデコードステートと実行ステートとメモリステートの一部を拡張回路で動作させる方法である。これは KD,KDR 命令に相当する。

動作の流れは、命令メモリより読み込まれたインストラクションをフェッチステートに設けられた簡易デコーダーによって、拡張回路転送命令（KD,KDR 命令）であるか判別し、拡張回路転送命令であれば拡張回路に送る。拡張回路は、デコードステートのクロックの立ち上がりでデータを読み込む。読み込まれたデータはユーザー任意の処理回路により順次実行され、メモリアクセスステートのときに、その結果（３２ビット）をプロセッサコアに送る。

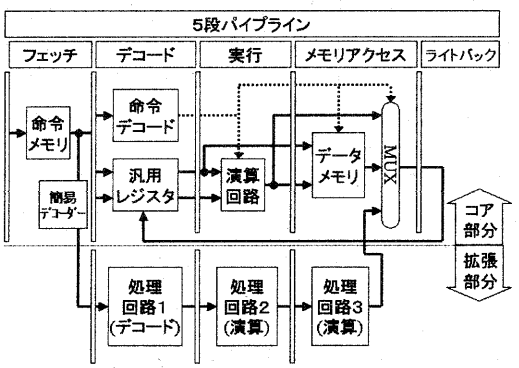

図２　拡張回路２の接続

そして、KDR 命令の時、ライトバックステートのクロックの立ち上がりでプロセッサコアの汎用レジスタに拡張回路の処理結果が書き込まれる。この接続方法もユーザー任意の回路であるため、入力に対する出力のタイミングとバス幅の制約を除けば、自由に設計することができる。

３．実装実験

３．１　HwModule

今回実験するにあたり、当研究室の PCI ボード（HwModule）を使用する。HwModule のブロック図を図３に、HwModule の写真を図４に示す。

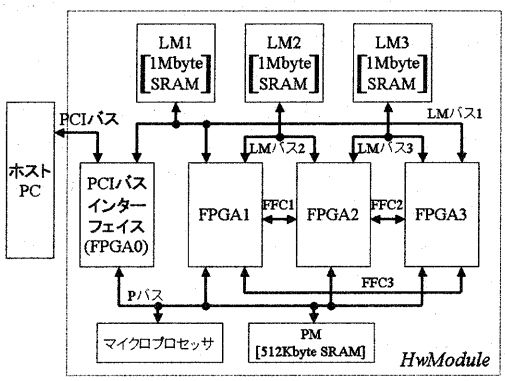

図３　HwModule のブロック図

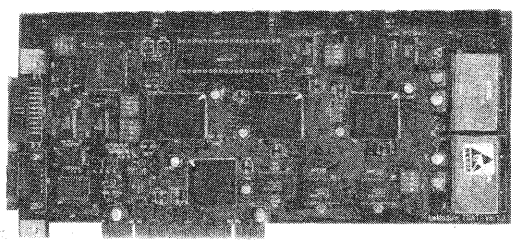

図４　HwModule の写真

３．１．１　HwModule の構成

HwModule は FPGA、メモリ、マイクロプロセッサ、PCI バスインターフェイスで構成される。FPGA が複数あり、このため FPGA が動作中であっても、動作していないFPGAの回路を書き換えることが可能となっている。FPGA は最大２０万ゲート規模のものである。メモリは LM（ローカルメモリ）バス側に３個、マイクロプロセッ

サのプログラムが置かれているPM（プロセッサメモリ）が１個ある。すべてのメモリがアクセスタイム１２ns のSRAM であり、容量はLM が１個１Mbyte、PM が５１２kbyte である。バス幅はLM バス側のデータバスが３２bit、P バス側のデータバスが１６bit、FPGA 間をそれぞれ結んでいるバス（FFC）が１８bit である。

３．１．２　HwModule の速度

HwModule 全体は、PCI バスのクロック（３３MHz）に同期して動作する。このため、各FPGA に供給されるクロックは３３MHz であるが、FPGA 内部のDLL 機能を使って、２倍の６６MHz で動作させることも可能である。また、この機能を使って位相をずらすことも可能であり、これによって３３MHz のクロック１クロック内に、メモリアクセスが可能である。

３．１．３　FPGA の回路の書き換え

FPGA の回路の書き換えは、マイクロプロセッサのI/O ポートに接続されたJTAG ポートを用いて行う。回路データは、主メモリからPM に送られた後、プログラムによりFPGA に書き込まれる。FPGA の書き換え速度は、８８０ms である。

３．２　プロセッサコア

プロセッサコア部分は、HwModule のFPGA1 に実装する。拡張回路部分は、FPGA2 に実装する。

FPGA1 内部は、以下の４つのモジュールで構成される。（図５）

- プロセッサコア
- ローカルメモリアクセスコントローラ
- LM バスブリッジ
- P バスインターフェイス

３．２．１　プロセッサコア

２章で述べたプロセッサコア部分である。

命令メモリをLM1 に、データメモリをLM2 に割り当てることにより、命令とデータを別々にパイプラインを構成することができ、パイプラインの崩れを押さえパフォーマンスの向上が期待できる。プロセッサコア、拡張回路共に３３MHz のクロックに同期して動作し、１クロックでメモリアクセスが可能である。

拡張回路への接続は、拡張回路１で１１２bit（デコードステート８０bit＋メモリアクセスステート３２bit）、拡張回路２で６４bit（フェッチステート３２bit＋メモリアクセスステート３２bit）必要である。拡張回路１で、デコードステートとメモリアクセスステートの接続を共有すれば８０bit あれば十分である。メモリアクセスステートで、拡張回路の処理結果をプロセッサコアに送るバスは、FFC1（FPGA1－FPGA2 間結線）とFFC2（FPGA2－FPGA3 間結線）とFFC3（FPGA3－FPGA1 間結線）を使い、残りはLM バス２のデータバス及びアドレスバスを使う。これによって２つのハザードが生じる。（１つは、データメモリへのアクセスとプロセッサコアから拡張回路へのアクセス。もう１つは、プロセッサコアから拡張回路へのアクセスと拡張回路からプロセッサコアへのアクセス。）

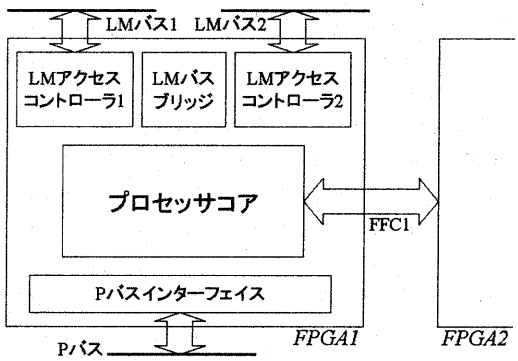

図５　FPGA1 内部モジュール

３．２．２　ローカルメモリアクセスコントローラ

ローカルメモリは複数のFPGA と接続しているため、ローカルメモリへのアクセスをコントロールする必要がある。LM バスの管理は、PCI バスインターフェイスが行っているので、ローカルメモリへのアクセスは、PCI バスインターフェイスにバス使用権を要求し、許可が下りてから行う。

３．２．３　LM バスブリッジ

LM バスブリッジは、ホストからの要求によりPCI バスインターフェイスがLM2 にアクセスするときに、LM バス１とLM バス２を結ぶ。

３．２．４　P バスインターフェイス

P バス経由で結ばれているマイクロプロセッサとのインターフェイス。

3．3　拡張回路

拡張処理部分には、画像処理などの前処理として用いられる、Haarの離散ウェーブレット変換を行う回路を実装する。

3．3．1　Haarの離散ウェーブレット変換

Haarのスケーリング関数$\phi_H(x)$と、Haarのウェーブレット関数$\phi_H(x)$は式（1）、（2）のように定義される。

$$\phi_H(x) = \begin{cases} 1 & (0 \leqq x < 1) \\ 0 & (\text{otherwise}) \end{cases} \qquad (1)$$

$$\phi_H(x) = \begin{cases} 1 & (0 \leqq x < 1/2) \\ -1 & (1/2 \leqq x < 1) \\ 0 & (\text{otherwise}) \end{cases} \qquad (2)$$

スケーリング関数$\phi_H(x)$をLowpass Filter、ウェーブレット関数$\phi_H(x)$をHighpass Filterとして、入力画像（Original image）の水平方向、垂直方向の順に1次変換を行うことで、1/4に圧縮された低周波成分（Scaling image）と高周波成分（Wavelet image）が得られる。（図6）

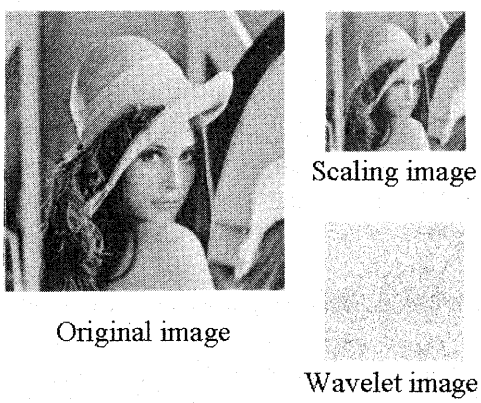

図6　Haar-Wavelet変換による圧縮

3．3．2　多重解像度表現

図6のように、ウェーブレット変換を1回行うことを1レベルの変換とし、ここで得られる原画像を1/4に圧縮した画像データをレベル1とする。ウェーブレット変換してできた低周波成分を、さらに変換するという作業を繰り返すことにより得られる信号の表現を多重解像度表現という。（図7）

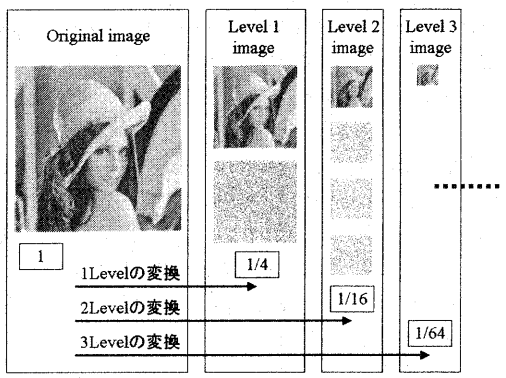

図7　多重解像度表現

3．3．3　ウェーブレット変換回路

拡張回路にウェーブレット変換回路を実装するにあたり、拡張回路1の接続方法で実現する。図1の処理回路1に1Levelの変換を行う回路を割り当て、処理回路2に1Levelの変換を行う回路と処理回路1の結果をそのまま送る回路を割り当てる。これによって1回の拡張命令で1Levelの変換と2Levelの変換（1Levelの変換＋1Levelの変換）が可能となる。3Levelの変換、4Levelの変換をするには、拡張命令を2回する。

今回、ウェーブレット変換をするにあたり、原画像は256色グレースケールを採用する。よって、拡張回路からの出力（32bit）には、4pixel分のデータを送ることができるので、原画像の最小の大きさは、1Levelの変換では16pixel（2pixel×8pixel）、2Levelの変換では64pixel（4pixel×16pixel）、3Levelの変換では256pixel、4Levelの変換では1024pixelとなる。（図8）

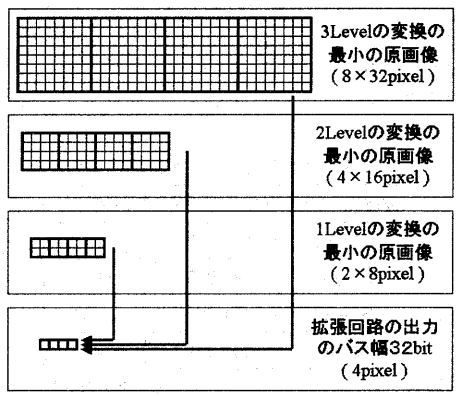

図8　各レベルの最小原画像の大きさ

3．4　プログラム

2 Level の変換を例にあげ主要部分の処理の流れを図９に示す。2 Level の変換の最小単位である４×１６pixel ごとに変換を行う。変換は、((縦のサイズ÷４）×（横のサイズ÷１６)) 回行う。

① ロード命令によりメモリから４×１６pixel の画像データ、１６WORD 分を汎用レジスタにロードする
② 拡張命令（KE）命令により汎用レジスタの値を拡張回路に送ると同時に拡張回路制御信号を送る
③ 拡張命令（KER）命令により汎用レジスタの値を拡張回路に送ると同時に拡張回路制御信号を送り、変換結果を汎用レジスタに書き込む
④ セーブ命令により汎用レジスタに格納された変換結果をメモリにセーブする

図９　処理プログラム

3．5　結果

3．5．1　実装シミュレーション結果

表２に実装シミュレーションによる各モジュールを Implementation した結果を示す。すべてのモジュールにおいて動作速度の３３MHz を満たしている。

表２　実装シミュレーション結果

モジュール	回路規模(gate)	最大処理速度(MHz)
FPGA1 全体	35,413	45.276
(プロセッサコア部分)	18,195	51.372
FPGA2(拡張回路)	17,716	46.106

3．5．2　処理結果

表３に５１２×５１２pixel グレースケールイメージを各処理レベルのウェーブレット変換に要した時間を示す。Scaling image と Wavelet image とも変換に要する時間は同じである。

表３　処理結果（512×512 pixel）

処理レベル	クロック数(clk)	時間(ms)
1Level	247,047	7.411
2Level	135,815	4.074
3Level	121,159	3.634
4Level	116,391	3.491

(1clk=30ns)

また今回の実験では、ホスト PC（６００MHz）上の主メモリにある Original image と画像サイズ及び変換レベルに応じたプログラムを HwModule 上の LM（ローカルメモリ）に送り、再構成可能プロセッサで実行した後、ホスト PC 上の主メモリに書き戻す作業を行った。５１２×５１２pixel グレースケールイメージを送るのに、２３ms かかった。また、プログラムを送るのに要した時間は、レベル４までの変換で０．２ms 以内であった。

3．5．3　ソフトウェアとの比較

３．５．２で示した結果と同じ処理をソフトウェアで実行した。６００MHz の PC を使用し、2 Level の変換を行った結果、ウェーブレット変換処理に要した時間は約１０．８ms であった。これは 6,480,000clk 分に相当し、再構成可能プロセッサによる処理（135,815clk）の約４８倍である。

4　むすび

本稿では再構成可能なプロセッサを用い、ウェーブレット変換をする拡張回路について、実装実験の手法とその結果について述べた。高速性と柔軟性は実現することができたが、設計・生産性の向上に向けたとりきめが今後必要である。

文　献

[1] 木田裕之，木村晋二，高木一義，あべ松竜盛，渡邉勝正 “再構成可能部を持つ Java プロセッサ” 1999 年電子情報通信学会総合大会, SA-2-6, Mar. 1999

[2] WWW サイト：http://www.nec.co.jp/press/ja/0210/1604.html

[3] WWW サイト：http://pcweb.mycom.co.jp/news/2002/09/10/07.html

[4] 田中茂 “システム LSI のプラットホーム事例” デザインウェーブマガジン vol.7, no.8, pp.42-53, Aug. 2002

[5] 仁木雅，関根優年，伊藤光，木場俊暁 "部分選択的に画像変換を行う Haar-Wavelet 変換チップ" 第５回システム LSI ワークショップ予稿集, pp.235-238, Nov, 2001