

## 動的リコンフィギャブルプロセッサにおける可変クロック機構の導入

天野 英晴<sup>†</sup> 安達 義則<sup>†</sup> 堤 聡<sup>†</sup> 石川健一郎<sup>†</sup>

<sup>†</sup> 慶應義塾大学理工学部 〒223-8522 横浜市港北区日吉 3-14-1

E-mail: [†drp@am.ics.keio.ac.jp](mailto:†drp@am.ics.keio.ac.jp)

あらまし 動的リコンフィギャブルプロセッサは、一つのタスクを複数のコンテキストを切り替えながら解くことで高い面積効率を実現する。しかし、全体の実行周波数が最大遅延を持つコンテキストによって制限されるため、コンテキストの遅延時間に差がある場合に、損失が大きくなる。そこで、コンテキスト毎に実行クロックを可変にする方法を提案し、NEC エレクトロニクス社の DRP-1 への実装手法を示す。いくつかのアプリケーションによって評価した結果、可変クロック機構は、全体の動作周波数が低く、それぞれのコンテキストの遅延時間にばらつきが多い場合に効果的であり、例外を除き、10%-50% の性能向上を確認した。

キーワード リコンフィギャブルプロセッサ, DRP, 動的再構成

### A context dependent clock control mechanism for dynamically reconfigurable processors

Hideharu AMANO<sup>†</sup>, Yoshinori ADACHI<sup>†</sup>, Satoshi TSUTSUMI<sup>†</sup>, and Ken'ichiro ISHIKAWA<sup>†</sup>

<sup>†</sup> Dept. of ICS, Keio University 3-14-1 Hiyoshi, Yokohama, 223-8522 Japan

E-mail: [†drp@am.ics.keio.ac.jp](mailto:†drp@am.ics.keio.ac.jp)

**Abstract** Dynamically Reconfigurable Processors improve the area-efficiency by executing a task with multiple hardware contexts. The maximum operational frequency is limited with a context which has the largest delay time, and it causes a certain overhead when each context has various delay time. A context dependent variable clock method, which changes the clock in order to fit the current operational context, is proposed for NEC electronics' DRP-1. Performance evaluation using several applications reveals that the proposed method improves the performance from 10% to 50% with an exception.

**Key words** Reconfigurable Processor, DRP, Dynamic Reconfiguration

#### 1. はじめに

動的リコンフィギャブルプロセッサは、粗粒度のプロセッシングエレメント (PE) の機能と相互接続を動作中に変更することで、高い面積効率を実現可能であり、広い応用分野で利用可能な新しいアーキテクチャとして注目されている [11]。

なかでも NEC エレクトロニクス社の DRP [1] などのマルチコンテキスト型は、一つのタスクに対して複数のコンテキストを切り替えて用いることで面積効率を改善することができる。様々なアプリケーションの実装を通して、コンテキスト切り替えを用いない場合に比べて 4 倍から 14 倍の面積効率を実現することが可能であることが明らかになった [9]。しかし、一方で、評価結果から、かなり工夫して実装しても、利用 PE 数、クリティカルパスはコンテキスト間で相当の差が生じることも明らかになった。この場合、最大動作周波数は、クリティカルパスの長い特定のコンテキストに制約されてしまう。

そこで、本稿では、コンテキストによって、動作周波数を変更するコンテキスト依存型可変クロック機構を提案する。現在の DRP-1 は、多数の PE と分散メモリを用いた並列処理によって、ハイエンドの DSP に比べて数倍の性能を実現することができる。しかし、アプリケーションの多くは、全てのコンテキストで高い並列性を維持することが難しく、この点が性能向上の障害となっている。ここで、並列性の低いコンテキストの多くは、最大遅延時間が小さく、本来高い周波数で動作可能である。このような場合、コンテキスト依存型可変クロック機構を導入することで、並列性の低いコンテキスト実行を高速なクロックで実行することが可能となり、結果として性能をさらに改善することができる。通常、動作クロックを可変にすると、入出力等、他モジュールとのインタフェースが問題になるが、この点は入出力コンテキストの分離 [10] によって解決することができる。

本稿では、入出力の分離法を示した後、DRP の演算用コンテキストに対して可変クロックを用いる手法を述べ、実アプリケーション

ジョンを用いてその効果を評価する。

## 2. 対象システム

### 2.1 動的リコンフィギャラブルプロセッサ

動的リコンフィギャラブルプロセッサは、現在の SoC の問題点の解決法として注目されている。

メディア処理や通信制御では、MPEG や JPEG などのコード圧縮および復号、DES、AES などにより暗号化されたコード、Viterbi、Turbo などのエラー修正用コードの取り扱いなど、強力な演算能力を必要とされる処理を含んでいる。一方で、システム管理、ユーザインタフェース機能も必要であり、データ処理の一部にも比較的演算能力を要しないが複雑な処理が混在している。SoC(System on a Chip) は、このような性質に適合するように、演算能力は必要としないが複雑度の高い処理を担当する組み込み用 CPU と、強力な演算能力を要する処理をハードウェア化したモジュールを同一チップ上に混載することにより、低コスト、高性能、低消費電力の ASIC を実現している。

しかし、専用ハードウェアは、特定の処理に特化してしまうため、他の用途に転用することができず、用途別に様々な構成のハードウェアを混載した ASIC 開発が必要となる。しかも、JPEG2000、Turbo コードなど新しく複雑なコード化方式が次々に登場し、これに合わせて次々と新たな専用ハードウェアを混載する必要が生じ、このためのコストが大きな問題となっている。ここで、動的リコンフィギャラブルプロセッサは、専用ハードウェアに代わる効率的な柔軟性を持ったアーキテクチャである。構成は柔軟に変更できるため、新しい技術も簡単に採り入れることが可能である。開発コストを削減すると共に、単一のチップを大量に生産すれば良いことからチップ自体のコストの削減も期待できる。柔軟性の持ったハードウェア自体の構造が決まっていれば、新しいプロセスにもいち早く対応することができる。また、場合によっては製品の配布後もハードウェアの構造を変更することで、発生した問題点や新たな機能付加に対処することができる。

本稿では、このような目的を考慮し、組み込み用プロセッサと NEC エレクトロニクス社により開発された DRP-1 [1] の PE アレイのユニットである Tile 複数個を組み込み用 CPU に接続したシステムを想定する。

### 2.2 DRP の構成

DRP は、粗粒度のリコンフィギャラブルデバイスであり、Tile と呼ばれるリコンフィギャラブルユニットを構成単位とする。各 Tile は図 1 に示す通り、 $8 \times 8$  の Processing Element(PE) アレイ、状態制御を行なうシーケンサである State Transition Controller(STC) から構成される。また、 $8\text{bit} \times 256$  エントリのメモリ (VMEM) 8 セットとメモリコントローラ 2 セットを Tile の左右にもち、 $8\text{bit} \times 8192$  エントリのメモリ (HMEM) 4 セットとコントローラを Tile の上下にもつ。

各 PE は図 2 に示すように、8bit の ALU、シフトや簡単な論理演算を行なう Data Management Unit(DMU)、8bit の Flip Flop、レジスタファイルから構成される。コンフィギュレーション時には命令データが命令メモリに書き込まれ、実行中に STC

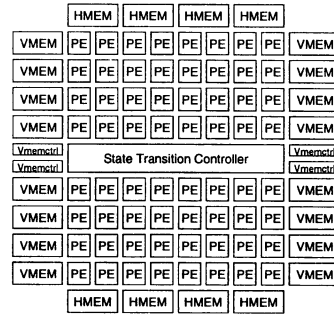


図 1 Tile の構造

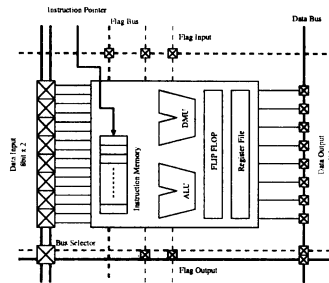


図 2 PE の構造

が発行した命令ポインタを命令メモリが受け、利用する命令データをロードすることでハードウェア構成を変更する。

DRP のプロトタイプチップ DRP-1 は、 $4 \times 2$  個の Tile から構成される Core の周辺部に 32 ビットの乗算器を 8 セット、メモリモジュール、PCI バスインタフェース、SDRAM/SRAM/CAM インタフェースを搭載したシステム LSI となっており、単独で PCI バスへの接続や外部メモリの制御が可能である。

DRP-1 は内部のメモリに最大 16 コンテキスト分の情報を蓄えることが可能で、クロックサイクル毎にコンテキスト切り替えが可能なマルチコンテキストデバイスである。

## 3. 可変クロック機構の導入

### 3.1 導入の必要性

DRP は分散メモリ構成であり、主として VMEM に格納されたストリームデータに対して、なるべく多数の PE で並列演算を行うことで高い性能を実現する。この場合、多くのアプリケーションでは、クリティカルパスを細かく切って、コンテキストの切り替えを増やすよりも、VMEM から読み出して一連の演算を行って、結果を VMEM または中間結果のレジスタに格納するまでを 1 ステップとし、この数ステップを 1 つのコンテキストに割り当てる手法を取った方が全体の性能が高くなる [9]。この場合、それぞれのコンテキストでの利用 PE 数およびクリティカルパスは、かなりばらついてしまう。図 3 に JPEG エンコードにおける DCT(Discrete Cosine Transform) 演算を DRP 上に実装した場合のそれぞれのコンテキストにおけるクリティカルパスを示す。この例では、全体の動作周波数は、最大のクリティカルパスを持つコンテキスト 7 の動作する 16MHz に制限されてしまう。それでも、多数の PE を用いた並列処理により 225MHz

で動作する TI の 8way VLIW 型 DSP TMS320C6713 の約 3 倍の性能を達成することができる [2]。しかし、コンテキスト 7 以外は、より高速なクロックで動作させることが可能であり、コンテキスト毎に最適なクロックで動作させることにより、さらに性能を向上させることが可能である。

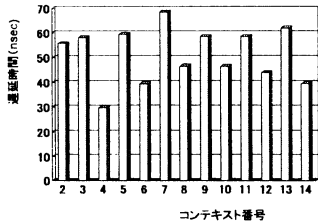


図 3 DCT の演算用コンテキストの遅延時間

### 3.2 コンテキスト依存型可変クロック機構

クロックの周期を自由に変える機構は非同期回路の東データ方式などで検討されている、ハンドシェイク信号の遅延を可変にする技術に近い。Speculative Completion [3] では数種類の遅延を、回路入力に応じて切り替えることにより遅延時間を最適化する。マルチコンテキスト型 FPGA におけるコンテキストの切り替えに可変遅延回路を利用した非同期 WASMII [4] では、この機構を用いて 30% 程度の性能向上を実現している。しかし、遅延線は、インパータの直列接続で実現する場合は、多くのゲートとマルチプレクサを必要とし、しかも正確なタイミングを実現することが困難である。

そこで、ここでは、簡単な図 4 に示すプリセッタブルカウンタで遅延回路を実現する。カウンタの大元の動作周波数  $f_{max}$  は、クロック可変の精度を決定する。すなわち、周波数が低いと可変の刻みが粗くなってしまうが、あまり高い周波数を用いると消費電力および実現性の点で問題が生じる。そこで、現在の DRP のアプリケーションの動作周波数を考慮して、200MHz のクロックを想定することとした。この場合、7bit の桁数で、640nsec (約 1.5MHz) までの範囲を 5ns の精度でカバーすることができる。5ns の刻み幅による損失がどの程度になるかは、それぞれのコンテキストの遅延時間に依存する。この点は後に実アプリケーションを用いて評価する。

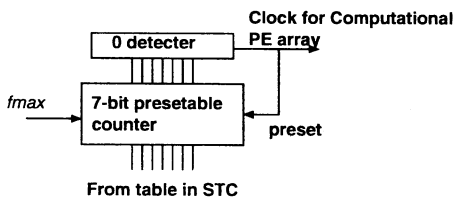


図 4 クロック可変機構

次に、DRP のコンテキストを制御する STC の状態遷移テーブルを拡張し、それぞれのコンテキストに対するクロックプリセット値を入れておく。このプリセット値がカウンタの初期値として設定され、カウンタがカウントアップして最大値に達すると、一定の遅延の後に 1 クロックパルスが発生し、これが PE の動作クロックとして用いられる。同時にプリセット値がカウンタにセットされるが、コンテキストが切り替わる際に STC から新しい値に更新される。この手法は、ゲート数も少なく、200MHz 程度の動作周波数であれば、無理なく実装が可能である。

### 3.3 入出力コンテキストの独立

可変クロックを用いる場合の問題は、コンテキストによって次に発生するパルスのタイミングが異なるため、全体の同期やデータ転送が困難になる点である。この点は入出力コンテキストを演算用コンテキストから独立させることにより解決される [10]。

この手法では、図 5 に示すように、I/O ピンとの接続部と、それ以外を独立のコンテキストとして動作させる。VMEM/HMEM は全体を二つに分割し、これらのペアをダブルバッファとして用いる。

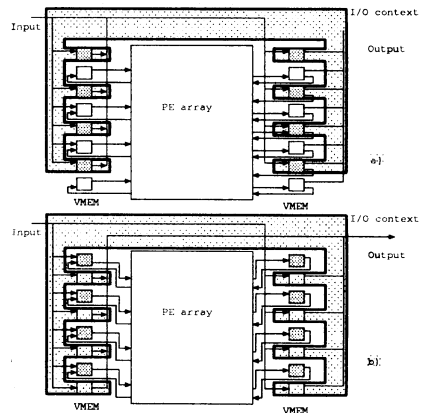


図 5 I/O コンテキストの独立

I/O 用コンテキストは、他のコンテキストと独立したクロックで独立動作し、独立したシーケンサ (I/O-STC) を持つ。I/O-STC は、演算用コンテキストのコントローラ STC との間で信号線を交換し、互いに相手が特定の状態に達するまで待ち合わせをすることを可能にする。まず図 5a) に示すように、PE アレイは半分の VMEM を用いて演算を行い、この間、I/O コンテキストは次のデータストリームの入出力を行う。演算が終了したら、VMEM の接続を図 5b) と入れ替え、再び入出力と演算を並行して行う。I/O コンテキストは、複数のステップを一つのコンテキストに割り付けることにより、単一コンテキストでの実現が可能である。したがって、I/O コンテキストはマルチコンテキスト構成にする必要はない。

上記の構成により、演算コンテキストと I/O コンテキストは、完全に同時動作するため、I/O 時間が演算時間よりも大きくなければ、これを完全に隠蔽することが可能である。さらに、I/O

O コンテキストのクロックと演算コンテキストのクロックを完全に独立にすることができる。このため、I/O コンテキスト用には、標準的なクロックを用いて、演算コンテキストには可変クロック機構を用いることが可能である。

#### 4. 評価

以下のアプリケーションについて、可変クロック機構を導入した場合の性能向上を評価した。

- DCT: JPEG エンコード時の離散コサイン変換 [2]。最大遅延は 62nsec で、並列性が高いため、TI の 8way VLIW 型の DSP TMS320C6713(225MHz) 動作の約 3 倍の性能を実現している。各コンテキストの遅延時間は比較的良く揃っている。
- IMDCT: MP3 デコード時の変形逆離散コサイン変換 [5]。最大遅延は 72nsec で大きいですが、並列性が大きいため、TMS320C6713 の 5 倍程度の性能を実現している。
- DWT: JPEG2000 デコード時の離散ウェーブレット変換 [7]。最大遅延は 56.7nsec で平均的であるが、PE 利用率が低い。
- MQC: JPEG2000 デコード時の算術符号化処理。最大遅延は 25.6nsec でかなり高速に動作するが、並列性は高くなく PE 利用率は低い。
- AES/CBC: Rijndael による AES の復号器 [6]。最大遅延は 15.7nsec。高速動作するがフィードバックがあり、PE 利用率が低い。
- SHA-1: IPsec で用いられる八種ベースの暗号化アルゴリズムの一つ HMAC-SHA-1-96 [8]。最大遅延は 29.4nsec で、かなり高速に動作する。

図 6 に現在の最大動作周波数で動作させた際の性能と比べて、可変クロック機構を導入した場合の性能向上率を示す。最大値と記されている値は、それぞれのコンテキストが最適な遅延時間で動作した時の性能向上率であり、この手法での性能向上の限界を示す。しかし、本稿で提案した可変クロックの刻み幅は、図 4 に示す回路の  $f_{max}$  により決定され、200MHz 利用時は 5nsec 刻みとなり、最大値に比べてその効果は小さくなる。

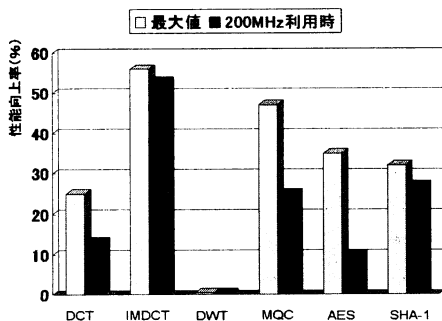


図 6 可変クロック機構による性能向上率

図 6 により、アプリケーションによって異なるが、全体としてある程度の効果が上がっていることが確認できる。IMDCT は、元々動作周波数が低い上に、それぞれのコンテキストの遅延時間の差が大きいことから 50%程度の性能改善を実現している。これに対して DWT は、ほとんど改善されない。これは、実行時間のほとんどを最大の遅延を持つコンテキスト 1 つが占めるためである。

AES, MQC は、最大値と 200MHz 利用時の差が大きい。これは、動作周波数が高いことから、5nsec で刻んだことによる損失が表面化したためである。SHA-1 は、同様に動作周波数が高いが、反復数の多いコンテキストのうち、いくつかの遅延時間が小さいため、かなりの効果が上がっている。全体をまとめると、DWT のような例外を除き、10%-50%の効果が有り、動作周波数が低く、それぞれのコンテキストの遅延時間の差が大きい設計ほど効果が上がっていることがわかる。

#### 5. おわりに

可変クロック機構の効果が少ない設計は、それぞれのコンテキストの遅延時間の差が少なく、遅延時間自体が小さいものである。すなわち、設計が優れている程効果が出ないことになる。逆に考えると、無理にチューニングをしない設計でもそれなりの性能を実現する技術であると考えられる。問題全体を解く場合に必要な消費エネルギーの低減、チップ全体として同時スイッチングノイズを低減する効果等についても期待できる。

#### 文 献

- [1] M. Motomura. A Dynamically Reconfigurable Processor Architecture. Microprocessor Forum, Oct. 2002.
- [2] M. Suzuki, et. al. "Stream Applications on the Dynamically Reconfigurable Processor". Proc. on IEEE IC-FPT2004, Dec. 2004.
- [3] S.M. Nowick. Design of a low-latency asynchronous adder using speculative completion, 1996.
- [4] Y. Adachi, K. Ishikawa, S. Tsutsumi, H. Amano. An implementation of the Rijndael on Async-WASMII. In Proc. on IEEE IC-FPT2003, pages 53-61, 2003.
- [5] Y. Yamada, K. Deguchi, N. Kaneko, H. Amano. Core Processor / Multicontext Device Co-design. In Proc. of Cool Chips VI, page 82.
- [6] 阿部、長谷川、黒瀧、天野、安生、粟島。"リコンフィギャラブルプロセッサ DRP-1 上での AES-CBC の実装"。第 4 回リコンフィギャラブルシステム研究会, 2004 年 9 月。
- [7] 出口、阿部、安生、粟島、天野。"DRP-1 上への JPEG2000 の離散ウェーブレット変換器と算術符号器の実装"。第 4 回リコンフィギャラブルシステム研究会, 2004 年 9 月。
- [8] 長谷川、安部、安生、粟島、天野。動的リコンフィギャラブルプロセッサを用いた IPsec アクセラレータの設計と実装。
- [9] 天野、阿部、出口。"動的リコンフィギャラブルプロセッサの基本的トレードオフの解析"。第 4 回リコンフィギャラブルシステム研究会, 2004 年 9 月。
- [10] 天野、阿部、出口、長谷川。"動的リコンフィギャラブルプロセッサにおける入出力と演算のバランス ~ 論理とデータ、どちらを動かすか? ~"。信学技法, CPSY, 2004 年 12 月。
- [11] 天野英晴。リコンフィギャラブルシステム最新情報。In SAC-SIS2004 チュートリアル資料, pages 1-9.