

メディア処理向け再構成可能アーキテクチャでの動画像復号処理の実現

密山 幸男^{†a)} 高橋 一真[†] 今井林太郎[†] 橋本 昌宜[†]
尾上 孝雄[†] 白川 功^{††}

Application Design of Multi-Standard Decoder on Media-Centric Reconfigurable Architecture

Yukio MITSUYAMA^{†a)}, Kazuma TAKAHASHI[†], Rintaro IMAI[†],
Masanori HASHIMOTO[†], Takao ONOYE[†], and Isao SHIRAKAWA^{††}

あらまし 面積効率の向上を目指したヘテロジニアス構造を有する粗粒度再構成可能アーキテクチャは、アプリケーション分野を特化することで高性能化と小面積化を実現することができる。そこで我々は、対象アプリケーションをメディア処理に特化したヘテロジニアス粗粒度再構成可能アーキテクチャARAMを開発してきた。本論文では、ARAMによって複数の動画像復号処理を実現できることを示すため、MPEG-2 デコーダ、MPEG-4 デコーダ、H.263 デコーダを設計対象として、各処理過程のマッピングについて述べる。更に、動画像復号処理の高性能化要求に対して、ARAMのスケラビリティと動画像復号処理の画素並列性を用いた性能拡張について述べる。またフィルタバンクのマッピングについて述べ、動画像復号処理以外にも適用できる機能拡張性を示す。キーワード 再構成可能アーキテクチャ、マッピング、動画像復号処理、画素並列処理

1. まえがき

汎用性を目指して高い柔軟性をもつ FPGA (Field Programmable Gate Array) に対して、回路の柔軟性を一部犠牲にして、より高い性能と小面積化を実現する粗粒度再構成可能アーキテクチャが数多く提案されている [1]。同種の機能セルを規則的に配置したホモジニアス構成を採用する再構成可能アーキテクチャは、アプリケーション分野を問わない汎用性をもちながら、並列性の高いデータパスの実現に適しているが、その汎用性に起因するハードウェアコストが課題となっている。これに対し、対象とするアプリケーション分野を限定し、面積効率の向上を図ったヘテロジニアス構成をもつ粗粒度再構成可能アーキテクチャが注目され

ている。

これまで我々はアプリケーション分野をメディア処理に特化することで、動画像復号処理を高い面積効率で実現するヘテロジニアス粗粒度再構成可能アーキテクチャ(ARAM: Area-conscious Reconfigurable Architecture for Media processing)を開発した [2]。ARAMを構成する再構成可能セルアレーは、機能セルの比率と配置を動画像復号処理の実装に適したヘテロジニアス構造とし、演算器や配線資源を限定することでハードウェアコストの大幅な削減を実現している。

本論文では、[2]で開発したARAMが複数の動画像復号処理を選択的に実現できることを示すため、MPEG-2 デコーダ、MPEG-4 デコーダ、H.263 デコーダを設計対象として、Variable Length Decoder (VLD)、Inverse Quantizer (IQ)、Inverse Discrete Cosine Transformer (IDCT)、Motion Compensator (MC) のマッピングについて述べる。面積効率を優先するため、再構成可能セルアレー上への制御データフローグラフのマッピングを体系的に定義した“マクロセル”を用い、すべてのマッピングは手作業で行っ

[†] 大阪大学大学院情報科学研究科, 吹田市
Graduate School of Information Science and Technology,
Osaka University, Suita-shi, 565-0871 Japan

^{††} 兵庫県立大学大学院応用情報科学研究科, 神戸市
Graduate School of Applied Informatics, University of
Hyogo, Kobe-shi, 650-0040 Japan

a) E-mail: mituyama@ist.osaka-u.ac.jp

た．また，再構成可能セルアレーのスケラビリティと，動画像復号処理の画素並列性を利用した性能拡張について述べる．更に，オーディオ処理でも用いられるフィルタバンクのマッピングについて述べ，ARAM が動画像復号処理以外にも適用できることを示す．

2. メディア処理向け再構成可能アーキテクチャ

2.1 アーキテクチャの概要

2.1.1 全体構成

ARAM [2] の全体構成を図 1 に示す．再構成可能セルアレーはプロセッサとの協調動作によりアプリケーションを実現する．再構成可能セルアレーの構成情報はオンチップの構成情報メモリに格納し，その他の処理データは外部メモリに格納する．

2.1.2 機能セル

再構成可能セルアレーは，表 1 に示す 4 種類のセルで構成され，各セルは図 1 (c) に示すように機能部と配線部で構成される．

配線部の構造は各種セルで共通となっている．図 2 に示すように，セルの上下左右に各 2 本，計 8 本の配線を有し，配線間スイッチにより配線接続と信号の方向を指定できる．配線部は以下の特長をもつ．

- 隣接セル間はスイッチを介さず直接接続されるが，スイッチを経由したセルの貫通接続によって離れたセル間でも通信できる．
- 8 ビット幅のデータ線の他に，キャリアやイネーブルのための 1 ビット幅のフラグ線を有する．メモリセルは他のセルの 4 倍のサイズになるため，四つの配線部を接続した構成になる．メモリセルの配線部は付録 1.1 に示す．

機能部の構造は 4 種類のセルで大きく異なる．各種セルの機能と入出力ポートを表 1 に示す．入出力ポートは，メモリセルを除いて，図 2 に示すワード線 (A0-A3, B0-B3) から選択することができる．ここで，一例として基本セルの機能部の詳細を図 3 に示し，乗算セル，レジスタセル，メモリセルの機能部の詳細は付録 1.2 で述べる．基本セルの機能部は，入出力レジスタ，ALU，シフタで構成され，表 2 に示す構成情報によって入出力レジスタ，ALU，シフタ，フラグ出力，ワード入出力の機能を設定することができる．

表 1 4 種類のセル
Table 1 Four types of reconfigurable cells.

種類	機能	入出力
基本セル	加減算，シフト，MUX, AND, OR	2 × 8-bit 入力，8-bit 出力
乗算セル	乗算	2 × 8-bit 入力，2 × 8-bit 出力
レジスタセル	2-port 16-word レジスタ	(read) 4-bit RA, 8-bit RD (write) 4-bit WA, 8-bit WD
メモリセル	1-port 256-word レジスタ	(read) 8-bit RA, 8-bit RD (write) 8-bit WA, 8-bit WD

RA: read address, RD: read data
WA: write address, WD: write data

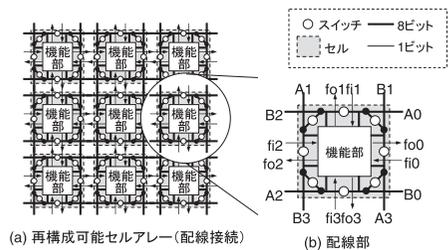


図 2 機能セルの配線部
Fig. 2 Interconnect part of reconfigurable cell.

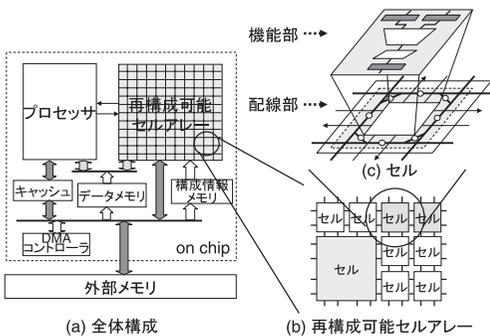


図 1 全体構成
Fig. 1 Overall architecture.

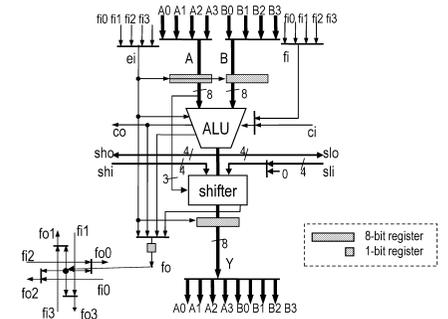


図 3 基本セルの機能部の構成
Fig. 3 Organization of function part of basic cell.

表 2 基本セルの構成情報
Table 2 Configuration data for basic cell.

ビット数	ユニット	構成
6	入出力レジスタ	パイプライン, ライトイネーブル, 定数, 未使用 (パイパス)
4	ALU	キャリー (f_i or c_i) 付き加算, 減算 論理演算 (AND, OR, or MUX), シフトタイプ (拡張または協調)
3	シフタ	シフト幅 (-4 ~ +3 bit)
4	フラグ出力	出力選択 (f_{o0} ~ f_{o3})
4	フラグ入力	入力選択 (f_i, e_i)
4	ワード入力	ポート A, B への入力選択
4	ワード出力	出力線の選択または Hi-Z 出力

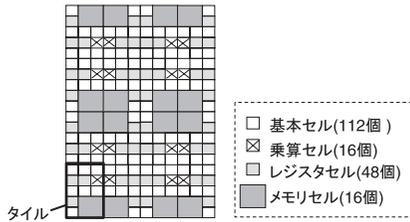


図 4 4 種類の機能セルによるヘテロジニアス構成
Fig. 4 Heterogeneous structure of 4 types of cells.

2.1.3 ヘテロジニアス構造

ARAM の再構成可能セルアレーは, 対象とするメディア処理を高いセル利用効率で実現するため, 以下の条件を満たすヘテロジニアス構造を有する. 再構成可能セルアレーのアレー構造と, 構成する各種セルの個数を図 4 に示す.

- 隣接する基本セルと乗算セルで 16 ビット乗算器を実現できる.
- 規則的配置に基づくスケラビリティを有するアレー構造とする.
- 動画像復号処理を実現する場合, 隣接する複数のメモリセルで 1 マクロブロック分のデコード中間データを格納できる.
- 制御回路や遅延調整回路を実現するレジスタセルをほぼ一様に分散配置する.

2.1.4 インタフェース

ARAM の再構成可能セルアレーは四つのインタフェースをもつ [2]. 以下ではそれぞれについて概説する.

ホストプロセッサインタフェース: 32 ビット幅のデータバスと, 1 ビットの制御信号線で構成される. データバスの各 8 ビットに対して, 再構成可能セルアレー (図 1) の左側にある基本セルのワード線 (A2, B2) から接続する. 同様に, 制御信号線に対してはフ

表 3 構成情報
Table 3 Configuration data size of 4 types of cells.

種類	構成情報 (bit)			個数
	機能部	配線部	レジスタ	
基本セル	29	12	16	112
乗算セル	25	12	16	16
レジスタセル	28	12	128	48
メモリセル	43	48	(2,048)	16

ラグ線 (f_{i2}, f_{o2}) から接続する.

外部メモリインタフェース: 32 ビット幅のデータバス, 24 ビット幅のアドレスバス, 1 ビット幅のイネーブル制御信号線で構成される. データバスとアドレスバスの各 8 ビットに対して, アレー上側と下側にある全セルのワード線 (A1, B1 または A3, B3) から接続する. 同様に, イネーブル制御信号線には, フラグ線 (f_{i1}, f_{o1} または f_{i3}, f_{o3}) から接続する.

データメモリインタフェース: コンテキスト切替時にレジスタ初期値をアレーの行または列単位で書き込むため, 96 ビット幅の三つのデータバスで構成される. 他と同様に, アレーの上側のワード線 (A1, B1), 右側と左側の基本セルのワード線 (A0, B0 または A2, B2) から接続する.

構成情報メモリインタフェース: コンテキスト切替時に構成情報メモリから構成情報を書き込むため, 492 ビット幅のバスで構成される. アレー上下方向に 12 本ある 41 ビット幅のコンフィギュレーション専用配線に接続する.

2.2 動的再構成機能

ARAM は動的再構成機能を有しており, 各セルの機能部と配線部の構成, 基本セルと乗算セルの入出力レジスタの初期値, レジスタセル内の 16 ワードレジスタの初期値を書き換えて, 複数のコンテキストを時分割実行する. ここで, 構成情報とレジスタ初期値は, それぞれ構成情報メモリとデータメモリに保持している. 1 コンテキスト分の構成情報の詳細は表 3 に示すようになっており, 合計で 16,752 ビットとなる. なお, メモリセル内の 256 ワードのデータは動的再構成時には書き換ええないため [2], コンテキストには含まない. 再構成可能セルアレーに対してコンテキストは並列に書き込まれ, 32 サイクルで切替が完了する.

2.3 マクロセルを用いたアプリケーション設計

ヘテロジニアス構成の再構成可能セルアレーでは, アプリケーションをマッピングする際に, 各機能セルの比率や位置関係といった固有の性質による制約を考

表 4 マクロセル
Table 4 Macro cells.

(a) セル単体によるマクロセル			
処理	使用セル (個数)	面積*	概要
遅延 (8-bit)	基本セル (1) or 乗算セル (1)	1	入出力レジスタによる 1 cycle 遅延
	レジスタセル (1)	1	1-16 cycles 遅延
遅延 (1-bit)	基本セル (1)	1	1 bit シフトのシフトアウトによる 1-8 cycles 遅延
平均	基本セル (1)	1	加算後に 1 ビット右シフト
累算	基本セル (1) or 乗算セル (1)	1	出力レジスタを利用
マスク演算	基本セル (1)	1	マスク値との論理和
比較	基本セル (1)	1	定数 (最大値のビット反転) との加算結果のキャリーアウトで判定
符号反転	基本セル (1)	1	符号ビットを示すフラグ入力により 0 との加減算を切替
アップ/ダウンカウンタ	基本セル (1)	1	加減算によるインクリメント/デクリメント. イネーブル制御が可能
制御信号	レジスタセル (1)	1	レジスタセルのフラグ出力
	基本セル (1)	1	基本セルからのキャリーアウト, シフトアウト, 符号信号
アドレス変換	レジスタセル (1)	1	入力アドレスでアドレス変換テーブルを参照
	メモリセル (1)	4	入力アドレスでアドレス変換テーブルを参照
(b) 複数セルの組合せによるマクロセル			
処理	使用セル (個数)	面積*	概要
加減算 (9-32-bit)	基本セル (2-4)	2-4	隣接する複数の基本セルの連携によるビット幅の拡張
可変長シフト (9-32-bit)	基本セル (2-4)	2-4	隣接する複数の基本セルの連携によるビット幅の拡張 (-4 ~ +3-bit shift)
乗算 (16 × 8-bit)	基本セル (2), 乗算セル (2)	4	乗算と加算の組合せで実現
乗算 (16 × 16-bit)	基本セル (5), 乗算セル (4)	9	乗算と加算の組合せで実現
テーブル (16, 24, 32-bit)	レジスタセル (2-4)	2-4	隣接する複数のテーブルによるビット幅の拡張
	メモリセル (2-4)	8-16	隣接する複数のテーブルによるビット幅の拡張
ビット結合	基本セル (3)	3	マスク演算でビット抽出した結果同士の論理和をとる
制御回路	レジスタセル (1), 基本セル (1)	2	レジスタテーブルとカウンタによる 16 状態のステートマシン (シーケンサ)
アドレス生成	基本セル (2 以上)	2	カウンタ出力から基本セル (シフタ, AND, OR など) で生成
ビットストリームシフタ	基本セル (3)	3	隣接する基本セルの連携とレジスタイネーブル制御によるシフト幅の拡張
飽和处理	基本セル (2)	2	大小比較の結果で MUX を制御.
ミスマッチ制御	基本セル (6)	6	累算結果の最下位ビットとビット結合し, 制御信号で出力制御

*) 基本セルの面積を 1 としたときのマクロセルの面積.

慮する必要がある. 既存のアーキテクチャでは, データフロー記述からの動作合成や, 制御データフローグラフの機能セルへの自動マッピングツールが用意されているが, 実装効率を向上させるためには手作業によるマッピングの変更が必要である [3].

ARAM の再構成可能セルアレーは, 面積効率を優先するため配線資源が大幅に限られているため, マッピングはより複雑になる. しかしながら, 対象アプリケーション分野を限定しているため, 頻繁に用いられるマッピングを “マクロセル” として定義しておくことができる. 現時点ではマッピングの自動化ツールは整備されていないが, マクロセルを用いることでマッピング作業の効率化を図り, 手作業によるマッピングであってもアプリケーション設計が可能である.

マクロセルは, 表 1 に示すセルの基本機能に基づいて構成される. 表 4 に示すメディア処理向けマクロセルは, セル単体によるマクロセルと, 複数セルの組合

せによるマクロセルに分けられる. メディア処理向けマクロセルの例として, ビットストリームシフタ, 飽和处理, ミスマッチ制御のマクロセルの構成を付録 2. で説明する.

マクロセルには, 組合せ回路だけでなく, 複数サイクルで動作する順序回路も用意されている. 複数サイクルで動作するマクロセルを用いる場合, 並列するパス上にタイミング調整用の遅延バッファを挿入する必要があり, 遅延サイクル数に応じて基本/乗算セルとレジスタセルを使い分ける. 動作周波数を向上させるためのパイプラインレジスタも同様に, セルの入出力レジスタやレジスタファイルを用いる.

2.4 動的再構成のためのコンテキスト分割

コンテキストの切替頻度が高くなると処理時間のオーバーヘッドが無視できなくなる. そこで, 以下の基準に従ってコンテキスト分割を決定する.

- 制御データフローグラフからマッピングに必要な

なセル数を見積もり、使用セル数が配置配線のための余裕をもつ（例えば、全セル数の 60%以下）ように分割する。

- 各コンテキストでの処理サイクル数に対して、コンテキスト切替オーバーヘッドが比較的小さく（例えば、各コンテキストでの処理サイクル数の 10%以下）になるようにする。

- メモリセルのデータはコンテキスト切替の前後で保持されるため、コンテキスト間で共有する中間データを格納する。ただし、データサイズやアクセス頻度に応じてデータメモリに格納する。

- メモリセルの初期化はシステム起動時のみ可能であり、あるコンテキストでテーブルとして使用するメモリセルは、他のコンテキストでは使用しない。

3. 動画像復号処理のマッピング

3.1 複数動画像復号処理マッピングの概要

本論文で対象とする符号化方式のビットストリームは、複数の階層で構成されており、そのうち下位層であるマクロブロック層とブロック層がビットストリームの大半を占める。復号処理は、各マクロブロックの先頭にあるヘッダの解析と、ヘッダ解析結果に基づく VLD, IQ, IDCT, MC の処理で構成される。ヘッダ解析は符号化方式に依存し、復号処理における制御情報を得る複雑な処理であるため、ホストプロセッサで実行する。一方、VLD, IQ, IDCT, MC の各処理は、符号化方式間で似通っているが、演算量が大きいため再構成可能セルアレーで実行する。そこで、ARAM では、図 5 に示すようなパイプライン処理を行う。図中の網掛けの部分ビットストリームにアクセスする処理であり、ホストプロセッサと再構成可能セルアレーで競合が発生しないようになっている。処理手順について以下に示す。

(1) マクロブロック (i) のヘッダ解析終了後、再構成可能アレーで VLD 以後の処理を開始。

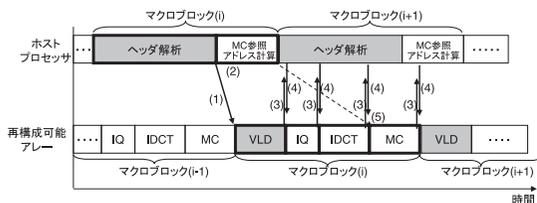


図 5 動画像復号処理スケジューリング
Fig. 5 Process scheduling of MPEG-2 decoder.

(2) ホストプロセッサは引き続き MC 参照アドレス計算を実行。

(3) 再構成可能アレーからホストプロセッサへは割込み信号によって各処理過程の終了を通知。

(4) ホストプロセッサから再構成可能アレーの動的再構成を制御。

(5) MC 参照アドレス計算は、再構成可能アレーで MC の処理開始までに完了。

以下では各処理の再構成可能セルアレー上へのマッピング方針について述べる。

3.2 VLD のマッピング

VLD の入出力データ構成と処理内容を、それぞれ表 5 と図 6 に示す。再構成可能セルアレーのコンテキスト切替が完了したあと、再構成可能セルアレーはマクロブロックヘッダ部より後のビットストリームを外部メモリから入力してメモリセルに格納する（図 6 の 2 行目）。その後、6 ブロック分の VLD を実行し、処理終了時はホストプロセッサに対して割込み信号で通知する。図 7 に、VLD のタイミングチャートを示す。

図 6 の 4~6 行目は、図 8 に示す制御データフローグラフで表現され、これをマクロセルを用いてマッピングすることで、再構成可能セルアレー上に VLD を実現することができる。符号長検出テーブルと可変長符号テーブルは、そのサイズや内容は符号化方式によって異なるが、テーブルサイズに応じてメモリセルとレジスタセルを組み合わせるマッピングされる。ま

表 5 VLD の入出力データ構成
Table 5 Input/output data format of VLD.

入力	マクロブロックヘッダの後に続く 1 マクロブロック分のビットストリーム
出力	量子化 DCT 係数 (run/level/sign/eob の組合せ) (6 ブロック分)

```

0. func_vld()
1. {
2.   1 マクロブロック分のビットストリームをメモリセルに
   パッパリング
3.   while(6 blocks){
4.     ビットストリームに含まれる可変長符号の切り出し
5.     可変長符号テーブルを参照して量子化 DCT 係数
   (run/level/sign/eob) に変換
6.     生成した量子化 DCT 係数をメモリセルに格納
7.   }
8. }
    
```

図 6 VLD の処理内容
Fig. 6 Process of VLD.

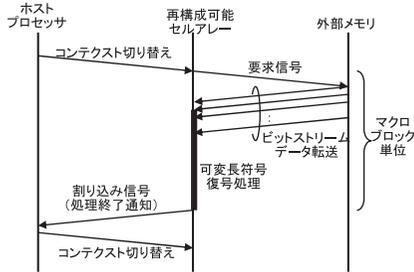


図 7 VLD のタイミングチャート
Fig. 7 Timing chart of VLD.

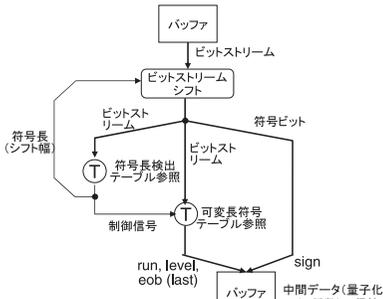


図 8 VLD の制御データフローグラフ
Fig. 8 Control data flow graph of VLD.

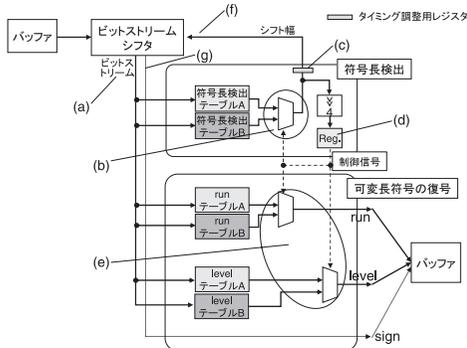


図 9 VLD のマッピング概要
Fig. 9 Outline of technology mapping result of VLD.

た、入力ビットストリームから可変長符号を切り出すためのビットストリームシフタは、マクロセルを用いてマッピングされる。ビットストリームシフタでは、1 サイクルで最大 8 ビットのビットストリームを読み込むため、一つの可変長符号の処理に必要なサイクル数は符号長によって異なる。例えば、MPEG-2 デコーダでは、最も長いエスケープコードの復号には 3 サイクル必要である。

ここで、マッピングの一例として、VLD のマッピ

表 6 IQ の入出力データ構成
Table 6 Input/output data format of IQ.

入力	量子化 DCT 係数 (run/level/sign/eob の組合せ) (6 ブロック分)
出力	8 × 8 DCT 係数行列 (6 ブロック分)

```

0. func_iq()
1. {
2.   while(6 blocks){
3.     メモリセルから run/level/sign/eob の読み出し
4.     level と 量子化係数との乗算, 飽和処理, ミスマッチ制御による DCT 係数の生成
5.     run の累算結果をインデックスとして逆スキャン用テーブルを参照し, メモリセルへの書き込みアドレスを生成
6.     生成した DCT 係数をメモリセルに格納
7.   }
8. }
    
```

図 10 IQ の処理内容
Fig. 10 Process of IQ.

ング概要を図 9 に示す。また、図中の記号を用いて、VLD の処理手順の詳細を以下に示す。

(1) 符号長検出テーブルの出力 (b) は、下位 4 ビットが符号長 (シフト幅) (c)、上位 4 ビットが可変長符号復号テーブル選択などの制御信号 (d) で構成される。

(2) 符号長が 8 ビット以下の場合、各テーブル A を参照し (b,e)、符号長と量子化 DCT 係数を出力する (e)。

(3) 符号長が 9 ビット以上の場合、符号長検出テーブル A からは、シフト幅が 8 ビット (c)、制御信号は各テーブル B を参照する情報が出力される (d)。次のサイクルで各テーブル B を参照し、符号長及び量子化 DCT 係数を出力する (e)。

(4) 符号長が 17 ビット以上の場合には更にもう 1 サイクルかけて量子化 DCT 係数を出力する。

以上のことから、図 6 に示す VLD の処理が、制御データフローグラフとマクロセルを用いてマッピングできることが分かる。

3.3 IQ のマッピング

IQ の入出力データ構成と処理内容を、それぞれ表 6 と図 10 に示す。

図 10 の 3~6 行目は、図 11 に示す制御データフローグラフで実現され、これをマクロセルを用いて再構成可能セルアレー上にマッピングすることで、IQ を実現することができる。ジグザグスキャン用テーブル (b) と量子化行列テーブル (d) は 64 エントリであるためそれぞれ 1 個のメモリセルにマッピングされ、累

表 8 MC の入力データ構成
Table 8 Input/output data format of MC.

入力	8 × 8 予測誤差データ (6 ブロック分), 参照画素データ (MC 参照アドレスを用いてフレームメモリから読出し)
出力	復元画像データ (6 ブロック分)

```

0. func_mc()
1. {
2.   while(6 blocks){
3.     for(i=0; i<8; i++){ //行単位の処理
4.       ホストプロセッサから MC 参照アドレスを取得
5.       フレームメモリから参照画素データを取得し、レジスタセルにバッファリング
6.       参照画素データから予測画素データを生成
7.       メモリセルから予測誤差データを読み出し
8.       予測誤差データと参照画素データとの加算結果から復元画像データを生成し、レジスタセルにバッファリング
9.       復元画像データを外部メモリに出力
10.    }
11.  }
12. }
    
```

図 14 MC の処理内容
Fig. 14 Process of MC.

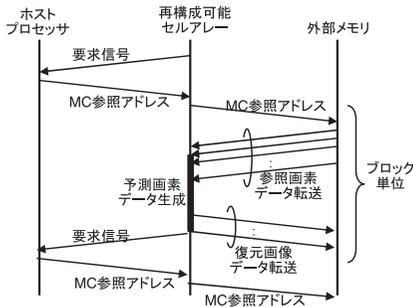


図 15 MC のタイミングチャート
Fig. 15 Timing chart of MC.

セッサインタフェースを介して転送され (図 14 の 4 行目), フレームメモリからの参照画素データの読出し (図 14 の 4 行目) や, 復元画像データの書込み (図 14 の 9 行目) は, 外部メモリインタフェースを介して実行される. この一連の処理はブロック単位で実行される.

図 14 の 6 行目は MC での演算の大半を占める半画素生成処理であり, 図 16 (a) に示す制御データフローグラフで表現される. 参照画素データを 1 ラインずつ読み込み, 2 個または 4 個の画素データから半画素データを算出する. $(h_x, h_y) = \{(1, 0), (0, 1), (1, 1)\}$ とすることで, 2 個の平均演算により, 図 16 (b) に示

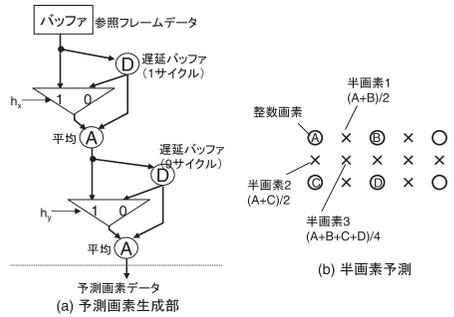


図 16 MC の予測画素生成部の制御データフローグラフ.
Fig. 16 Control data flow graph of half-pel manipulator in MC.

表 9 マッピング対象とした動画復号処理
Table 9 Implemented video decoders.

符号化方式	プロファイル	フレームサイズ	フレームレート	ビットレート
MPEG-2	Main	720 × 480	30 fps	6 Mbit/s
H.263	Baseline	352 × 288	30 fps	384 kbit/s
MPEG-4	Simple	352 × 288	30 fps	384 kbit/s

す半画素 1, 2, 3 をそれぞれ生成する. 遅延バッファや平均処理はマクロセルを用いて再構成可能アレー上に容易にマッピングされる.

MC では, 符号化方式に依存する動きベクトル計算をホストプロセッサで行うため, 再構成可能セルアレーで行う主な処理は符号化方式に大きく依存しない. また, 一つの画素データを 1 サイクルで生成することができるが, IDCT や IQ と同様に並列処理が容易であるため, 参照アドレスや参照画素データの入力に問題がなければ, 使用セル数に応じたスループットを得ることができる.

3.6 マッピング結果

以上で述べたマッピング方針に従い, 表 9 に示す MPEG-2 デコーダ, H.263 デコーダ, MPEG-4 デコーダの各処理過程を再構成可能セルアレー上にマッピングした. 各処理過程のマッピングに用いたセル数とその使用率 (面積割合) をコンテキストごとにまとめたものを表 10 に示す.

MPEG-2 デコーダは, VLD/IQ, IDCT (row), IDCT (column), MC の 4 コンテキストに分割した. VLD と IQ は同一コンテキスト上にマッピングするため, VLD で生成したデータはメモリセルに格納せず直接 IQ で処理を行い, IQ の処理結果をメモリセルに格納する. MC では双方向予測に対応するため, 予測

表 10 使用セル数
Table 10 Cell utilization.

コンテキスト		使用セル数				使用率**
		基本	乗算	レジスタ	メモリ	
MPEG-2	VLD/IQ	80	3	11	11	57.5%
	IDCT*	52	12	13	8	45.4%
	MC	87	0	15	4	49.2%
H.263	VLD/IQ	63	1	14	8	45.8%
	IDCT*	52	12	13	8	45.4%
	MC	87	0	15	4	49.2%
MPEG-4	VLD	83	0	13	9	55.0%
	IQ	36	4	0	9	31.7%
	IDCT*	52	12	13	8	45.4%
	MC	65	0	9	4	37.5%

*) 行方向と列方向に必要なセル数は同じ .

***) メモリセルの面積比率が他のセルの 4 倍であることを考慮 .

表 11 使用マクロセル
Table 11 Used macro-cell.

MPEG-2 デコーダ		
コンテキスト	使用マクロセル (数)	割合*
VLD/IQ	累算 (3), カウンタ (2), 制御信号 (3), 加算 (1), 乗算 (2), バッファ (3), テーブル (13), ビット結合 (2), 制御回路 (1), ビットストリームシフト (1), 飽和处理 (1), ミスマッチ制御 (1)	54.8%
IDCT	遅延 (2), カウンタ (1), 制御信号 (2), 乗算 (3), 加減算 (6), バッファ (3), テーブル (3), 制御回路 (1), アドレス生成 (1)	91.8%
MC	遅延 (4), 平均 (5), 累算 (3), カウンタ (5), 制御信号 (13), ビット結合 (9), 制御回路 (1), 加減算 (1), バッファ (5)	71.7%
H.263 デコーダ		
コンテキスト	使用マクロセル (数)	割合*
VLD/IQ	累算 (3), カウンタ (1), 制御信号 (9), 加算 (1), 乗算 (1), バッファ (3), テーブル (15), ビット結合 (1), 制御回路 (1), ビットストリームシフト (1), 飽和处理 (1), ミスマッチ制御 (1)	50.0%
IDCT	遅延 (2), カウンタ (1), 制御信号 (2), 乗算 (3), 加減算 (6), バッファ (3), テーブル (3), 制御回路 (1), アドレス生成 (1)	91.8%
MC	遅延 (4), 平均 (5), 累算 (3), カウンタ (5), 制御信号 (13), ビット結合 (9), 制御回路 (1), 加減算 (1), バッファ (5)	71.7%
MPEG-4 デコーダ		
コンテキスト	使用マクロセル (数)	割合*
VLD	累算 (2), カウンタ (1), 符号反転 (1), 制御信号 (17), 可変長シフト (3), バッファ (3), テーブル (15), ビット結合 (3), 制御回路 (1), ビットストリームシフト (1)	52.4%
IQ	累算 (1), カウンタ (1), 制御信号 (4), 加算 (1), 乗算 (2), バッファ (5), テーブル (2), ビット結合 (2), 飽和处理 (1), ミスマッチ制御 (1)	83.7%
IDCT	遅延 (2), カウンタ (1), 制御信号 (2), 乗算 (3), 加減算 (6), バッファ (3), テーブル (3), 制御回路 (1), アドレス生成 (1)	91.8%
MC	遅延 (2), 平均 (2), 累算 (2), カウンタ (4), 制御信号 (11), ビット結合 (7), 制御回路 (1), 加減算 (1), バッファ (3)	71.8%

*) 全使用セルに占めるマクロセルの割合 .

メモリセルの面積比率が他のセルの 4 倍であることを考慮 .

画素生成回路を二つ並列にマッピングしている .

H.263 デコーダは, MPEG-2 デコーダと同じコンテキスト構成とした . IQ のマッピングでは, 量子化行列やミスマッチ制御が MPEG-2 と比較して簡単であるため, 使用基本セル数は少ない . また, IDCT と MC のマッピングは MPEG-2 と共通である .

MPEG-4 デコーダの VLD では 3 種類のエスケープコードを扱うため, 他のアルゴリズムより処理が複雑になり, 使用セル数が多くなる . そこで, VLD と IQ を別コンテキストとして, 5 コンテキストに分割した . IDCT のマッピングは, MPEG-2 デコーダと共通である . MC には 4vector モードなど MPEG-2 デコーダにはない機能を有するが, 双方向予測が不要であるため, 結果的に MPEG-2 デコーダよりセル数が少なくなる .

また, マッピングに使用したマクロセルの内訳を表 11 に示す . IDCT のように比較的単純なデータパス回路の場合, 全使用セル中のマクロセルの比率が大きい . 一方で, VLD はセルの基本機能によるマッピングが多くなるため, マクロセルの割合が小さい . VLD 以外の処理ではマクロセルの割合は約 70%以上となっており, マクロセルが動画像復号処理の効率的なマッピングに有効であることが分かる .

4. 実装結果

4.1 配置配線

再構成可能アーキテクチャ上にアプリケーションを実装するため, 前述のマッピング後に配置配線を行う必要がある . そこで, FPGA 向け配置配線手法である VPR [5] に基づき, 自動配置配線ツールを作成した .

配置配線の例として, 各動画像復号処理で共通する IDCT について述べる . 図 13 に示した制御データフローグラフにデータ入出力用のメモリセル, タイミング調整用のレジスタなども含めたマッピング結果を図 17 (a) に示す . このマッピング結果に対して配置配線を行った結果, 図 17 (b) に示すような実装結果が得られた . 図中の数字は, 図 17 (a) と図 17 (b) で対応しており, 効率良く配置配線されていることが分かる .

次に, 動画像復号処理の実装例として, 表 10 で最もセル使用率の高い MPEG-2 デコーダについて, 各コンテキストの配置配線結果を図 18 に示す .

まず, 図 18 (a) に示す VLD/IQ では, アレー左上にある基本セルが外部メモリインタフェースとして機能し, 入力されたビットストリームをアレー左上のメ

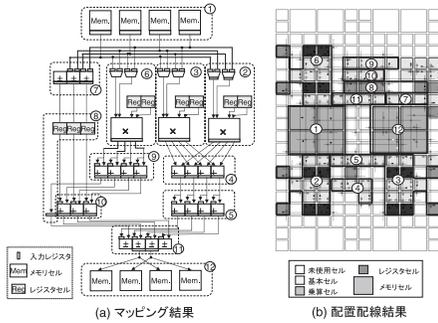


図 17 一次元 IDCT の配置配線結果
Fig. 17 Implementation result of 1-D IDCT.

表 12 配置配線後のセル使用数
Table 12 Cell utilization after placement and routing.

コンテキスト		マッピング に使用	配線のみ に使用	使用数 (使用率) *
MPEG-2	VLD/IQ	138	81	219 (91.3%)
	IDCT	109	60	169 (70.4%)
	MC	118	82	200 (83.3%)
H.263	VLD/IQ	110	35	145 (60.4%)
	IDCT	109	60	169 (70.4%)
	MC	118	82	200 (83.3%)
MPEG-4	VLD	132	37	169 (70.4%)
	IQ	76	18	94 (39.2%)
	IDCT	109	60	169 (70.4%)
	MC	90	66	156 (65.0%)

*) メモリセルの面積比率が他のセルの 4 倍であることを考慮し、全セル数を 240 とする。

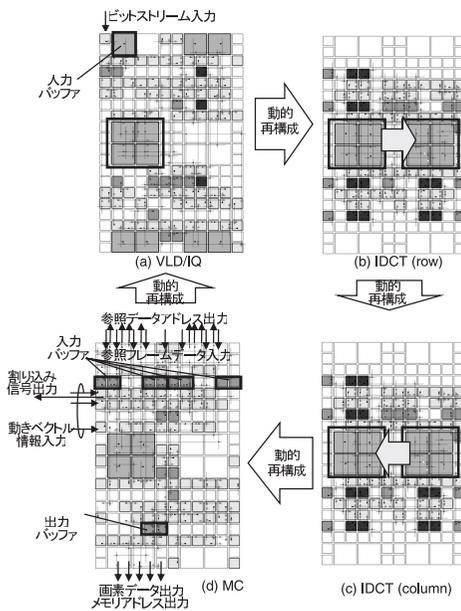


図 18 MPEG-2 デコーダの配置配線結果
Fig. 18 Implementation result of MPEG-2 decoder.

メモリセルに格納する。およそ下半分のセルアレーで VLD を実装しており、アレー下側の四つのメモリセルは符号長検出テーブルと復号化テーブルのマッピングしている。一方、上半分のセルアレーに IQ を実装しており、アレー右上の二つのメモリセルはジグザグスキャン用アドレステーブルと量子化行列のマッピングしている。VLD/IQ の処理結果はアレー中央左側に枠で囲んだ四つのメモリセルに格納する。

次に、図 18 (b) に示す IDCT (row) では、アレー中央左側の四つのメモリセルから IQ の処理結果を読み出し、1-D IDCT 演算を実行した後、アレー中央右側のメモリセルに結果を格納する。同様に図 18 (c) に

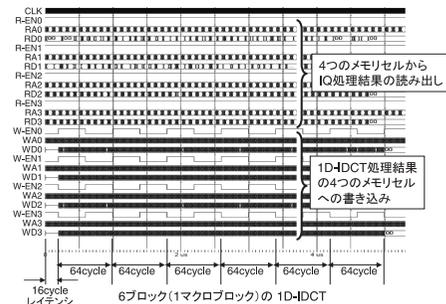


図 19 IDCT の配置配線結果を用いたシミュレーション結果 (動作周波数 80 MHz)

Fig. 19 Simulation wave form of IDCT.

示す IDCT (column) では、アレー中央右側のメモリセルからデータを読み出し、1-D IDCT 演算を実行した後、アレー中央左側のメモリセルに結果を格納する。最後に、図 18 (d) に示す MC では、まず、アレー左側にある基本セルがホストプロセッサインタフェースとして機能し、MC 参照アドレス情報が入力される。また、アレー上側にある基本セルが外部メモリインタフェースとして機能し、参照画素データが入力される。参照画素データから生成した予測画素データと、アレー中央左側のメモリセルから読み出した予測誤差データから、復元画像データを生成し、アレー下方にあるレジスタセルにバッファリングする。アレー下側にある基本セルも外部メモリインタフェースとして機能し、レジスタセルにバッファリングしている復元画像データを、フレームメモリに書き込む。

表 12 に示す配置配線結果から、MPEG-2 デコーダ、H.263 デコーダ、MPEG-4 デコーダが高いセル使用率を達成できていることが分かる。また、コンテキスト分割を決定する基準としてセル使用率を 60% とす

表 13 MPEG-2 MP@ML デコーダ実装例
Table 13 Implementation results of MPEG-2 MP@ML decoder.

	実装方式	面積 (mm ²)	テクノロジ	動作周波数 (MHz)	面積 (mm ²) (90 nm 換算)	面積性能効率 (pixels/sec·mm ² ·MHz)	備考
(a)	ARAM [2]	1.1 × 1.4	90 nm	80	2.64*	49,091	ARM9TDMI と協調動作 . MPEG-4 SP, H.263 Baseline の CIF, 30 fps デコード可能 . 74.2k gate .
(b)	ASIC [6]	15 × 15	0.5 μm	40	7.29	69,350**	70 MHz 動作時に 1152 × 1024 画素のリアルタイムデコードが可能
(c)	ASIC [7]	8.6 × 9.3	0.5 μm	54	2.59	74,093	MPEG Audio layer I/II デコーダも実装

*) ARM926EJ (I-cache 8KB, D-cache 8KB) の面積 1.1 mm²(90 nm) も含む .

**) 70 MHz 動作時の処理性能で換算 .

表 14 再構成可能アーキテクチャによる動画像復号処理の実装例
Table 14 Implementation results of video decoder on reconfigurable architecture.

	実装方式	デバイス	処理方式	動作周波数 (MHz)	回路規模*	備考
(d)	粗粒度再構成可能デバイス [8]	RICA [8]	2D-DCT	448.8 MHz	0.911 mm ² (90 nm 換算)	2D-DCT を 3.01 μs で実行 . 面積はプログラムメモリは含めず , OpenRISC の 7.6 倍 , ARM7-TDMI-S の 5.9 倍 .
(e)	粗粒度再構成可能デバイス [9]	RICA [8]	H.264 デコーダ	N/A	N/A	NTSC D-1, 21 fps . スタンダードコアの面積は OpenRISC の 4 倍 .
(f)	粗粒度再構成可能デバイス [10]	DRP-1 [11]	MPEG-2 デコーダ	22.3	約 2200 万トランジスタ	SD, 30 fps . VLD, IQ, IDCT のパイプライン実装 . MC も含めた実装にはコンテキスト数やタイル数の拡張が必要
(g)	粗粒度再構成可能デバイス [12]	ADRES [13]	H.264/AVC デコーダ	100	4 mm ² (90 nm)	CIF, 25 fps . 27 MHz 動作で CIF MPEG-2 デコードも可能 .
(h)	粗粒度再構成可能デバイス [14]	XPP-III [15]	H.264/AVC デコーダ	400	75 mm ² (90 nm)	HD, 24 fps . XPP40.16.8 アレー (40 ALU-PAEs, 16 RAM-PAEs, 8 FUC-PAEs)
(i)	FPGA [16]	FLEX10K 200E [17]	MPEG-2 デコーダ	29.8	120 k gate (換算値)	SD, 30 fps . LE (logic) 60% , EAB (memory) 46%使用
(j)	FPGA [18]	Virtex-II Pro XC2VP30 [19]	H.264/AVC デコーダ	100	189k gate (換算値)	HD, 64 fps . ただし , エントロピーデコーダを除く .
(k)	FPGA [20]	Vertex-II XC2V3000 [21]	MPEG-4 エンコーダ	100	186 k gate (換算値)	パイプライン処理により 4CIF 30 fps を達成
(l)	FPGA [22]	Stratix EP1S40 [23]	MPEG-4 エンコーダ	70	243 k gate (換算値)	NIOS CPU を 4 個実装し , QIF 6 fps を達成
(m)	FPGA [24]	Stratix EP1S40 [23]	MPEG-4 エンコーダ	50	227 k gate (換算値)	NIOS CPU を 4 個実装し , QIF 13 fps を達成

*) FPGA の回路規模は , 使用された LUT 数や LE 数から換算 .

ることも妥当であるといえる .

再構成可能セルアレーの HDL コードと , 配置配線結果として得られる配置配線結果を用いて機能検証を行った . 論理シミュレーションの実行例として , 図 19 に 1D-IDCT の論理シミュレーション結果を示す . 論理シミュレーション結果から , 1 マクロブロック (6 ブロック) の処理を 400 サイクルで完了できていることが分かる .

4.2 他の実装方式との比較

まず , ASIC による実装結果と比較するため , MPEG-2 MP@ML デコーダの実装結果を表 13 に示す . デコードに必要な演算量はビットレートにも依存するため単純な比較は困難であるが , 単位時間当りに復号される画素数をデコード処理性能と定義する . ここで , 面積性能効率としては , 単位面積当りのデ

コード処理性能で比較するべきであるが , 製造プロセスのスケーリングによる動作周波数の向上も考慮する必要があるため , 更に動作周波数で正規化した . 外部インタフェースなどの周辺回路が異なり , 更に実装例 (b) では MPEG Audio デコーダも実装されていることも考慮する必要があるが , ARAM は ASIC の面積性能効率に対し 30%程度劣っていることが分かる . しかしながら , 詳細は次章で述べるが , ARAM ではアレー構造のスケーラビリティを利用した処理性能の拡張性を有し , これが専用回路による実装に対する優位性となっている .

次に , 再構成可能アーキテクチャによる動画像復号処理の実装例を表 14 に示す . 実装例 (d) では , 命令セットベースで Instruction cell を合成し , プログラムをマッピングする RICA [8] を用い , 2D-DCT を小

面積で実現できている。しかし、RICA の面積はプログラムメモリ等のサイズを除いた数値となっている。実装例 (e) では、RICA による H.264/AVC デコーダの実装が報告されているが、スタンダードコアと呼ばれる回路のサイズ以外は明記されておらず、明確な比較が困難である。

実装例 (f) では、DRP-1 [11] による MPEG-2 デコーダの実装例が報告されている。ARAM より低い動作周波数で MPEG-2 デコーダを実現しているが、回路規模が大きくなっている。また、参考までに、H.264/AVC デコーダの実装例 (g), (h) と比較すると、デコードフレームサイズは大きく異なるが、回路面積はそれぞれ ARAM の 1.5 倍、28 倍となっている。ただし、H.264/AVC ではエントロピーデコーダも含めて複雑な制御処理が多くなるため、ARAM で実装する場合は、より高性能なホストプロセッサが必要になると考えられる。

実装例 (i) では、FPGA による MPEG-2 デコーダの実装例が報告されている。ARAM と比較して、動作周波数は低くなるが、回路規模が 1.6 倍となっている。FPGA による H.264/AVC デコーダの実装例 (j) では、HD サイズのデコードを ARAM の 2.5 倍の回路規模で実現するが、演算量の大きいエントロピーデコーダを除いた数値である。また、参考までに、動画像符号化処理である MPEG-4 エンコーダの実装例 (k), (l), (m) と比較すると、フレームサイズやフレームレートは異なるが、符号化処理は演算量が大きくなるため、回路規模はそれぞれ 2.5 倍、3.3 倍、3.1 倍となっている。

5. アプリケーション設計における拡張性評価

5.1 性能拡張

ARAM のアレー構造はヘテロジニアス構造でありながらタイルを基本単位としているため、動画像復号処理の画素並列性を生かした性能拡張が可能である。ただし、再構成可能セルアレーの性能拡張に合わせて、協調動作を行うプロセッサもより高い性能が求められる。

動画像復号処理の高性能化として、フレームサイズを拡大すると、単位時間当りに処理すべきマクロブロック数が増大する。また、ビットレートを向上させると、可変長符号のシンボル数が増大する。ここで、動画像復号処理における各処理過程の演算量は、VLD

では可変長符号のシンボル数に依存し、IQ, IDCT, MC では画素数 (マクロブロック数) に依存するという性質がある。また、IQ, IDCT, MC は画素並列処理が可能であるのに対して、VLD は並列処理が困難である。以上のことを考慮して、(1) 各コンテキストにおいて処理を並列化する手法と、(2) 各コンテキストを展開しパイプライン化する手法による ARAM の性能拡張について、以下に説明する。

(1) コンテキスト内処理並列化による性能拡張

ARAM は、CIF, 384 kbit/s の MPEG-4 リアルタイムデコードが可能である。ところが、最近では携帯電話による動画像撮影・再生機能として VGA (640 × 480 画素) に対応し、データ通信速度も下り 7.2 Mbit/s を達成する。そこで、MPEG-4 デコーダを VGA, 7.2 Mbit/s に対応させる性能拡張について述べる。

フレームサイズが VGA の場合、マクロブロック数から逆算して、1 マクロブロックの要求処理時間は $27.78 \mu\text{s}$ になる。ここで、シンボル数がビットレートに比例すると仮定すると、CIF, 384 kbit/s から VGA, 7.2 Mbit/s になれば、1 マクロブロック当りのシンボル数は 6.18 倍になると考えられる。表 15 (a) に示すように、CIF, 384 kbit/s デコード時の VLD の処理時間は、ホストプロセッサへの割込みや動的再構成にかかるオーバーヘッド ($1.53 \mu\text{s}$) を含めて $2.58 \mu\text{s}$ になることから、VGA, 7.2 Mbit/s デコード時の 1 マクロブロックの処理時間は、 $((2.58 - 1.53) \times 6.18 + 1.53) + 5.51 + 11.18 + 7.73 = 32.43 \mu\text{s}$ となり、要求処理時間を満たせない。

そこで、再構成可能セルアレーのサイズを図 20 (b) に示すように 2 倍に拡大し、IQ, IDCT, MC の処理時間を半減 (オーバーヘッド除く) させると、1 マクロブロックの処理時間は図 21 (b) に示すように $21.87 \mu\text{s}$

表 15 MPEG-4 デコーダの性能拡張
Table 15 Performance enhancement of MPEG-4 decoder.

	(a) 拡張前	(b) 拡張後	
フレームサイズ	CIF (352 × 288)	VGA (640 × 480)	
ビットレート	384 kbit/s	7.2 Mbit/s	
フレームレート	30 fps	30 fps	
マクロブロック数	11,880	36,000	
要求処理時間 [μs]	84.18	27.78	
処理時間 (オーバーヘッド) [μs]	VLD	2.58 (1.53)	8.02 (1.53)
	IQ	5.51 (0.59)	3.05 (0.59)
	IDCT	11.18 (1.18)	6.18 (1.18)
	MC	7.73 (1.53)	4.63 (1.53)
	TOTAL	26.99 (4.81)	21.87 (4.81)

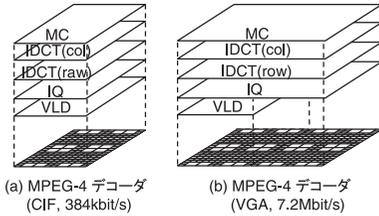


図 20 MPEG-4 デコーダのフレームサイズ拡張に対するマッピング

Fig. 20 Mapping scalability for the enhancement of frame size on MPEG-4 decoder.

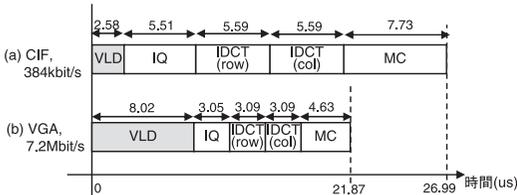


図 21 MPEG-4 デコーダにおける 1 マクロブロックの処理時間とそのスケジューリング

Fig. 21 Process scheduling for the enhancement of frame size on MPEG-4 decoder.

となり、要求処理時間を満たすことができる。

以上のように、IQ, IDCT, MC の画素並列性を利用することで、動画像復号処理の性能拡張が可能となる。一方で、並列処理が困難な VLD がボトルネックとなることが課題である。

(2) コンテキストパイプライン化による性能拡張

より大幅な高性能化が求められる場合、前述のような単純なアレー規模拡大による時分割実行では対応できない場合がある。ARAM は、SD, 6 Mbit/s の MPEG-2 リアルタイムデコードが可能であるが、既に処理能力の上限に近い。例えば、これを BS ハイビジョン放送で要求される HD(1920 × 1088), 24 Mbit/s に対応させる性能拡張について考える。

まず、フレームサイズが HD の場合、1 マクロブロックの要求処理時間は 4.08 μs になる。ここで、1 マクロブロック当りのシンボル数は、ビットレートとマクロブロック数から計算して 0.66 倍になると考えられるが、アレーサイズを拡大して IDCT, MC の処理 (表 16(a)) を高速化しても、要求処理時間を満たすことは困難である。

そこで、図 22(b) に示すように、IDCT と MC を 2 倍の数のセルで処理することで処理時間を削減し、更に時分割実行していた各コンテキストを空間展開し、パイプライン化することで大幅な性能拡張を行う。こ

表 16 MPEG-2 デコーダの性能拡張

Table 16 Performance enhancement of MPEG-2 decoder.

	(a) 拡張前	(b) 拡張後	
フレームサイズ	SD (720 × 480)	HD (1920 × 1088)	
ビットレート	6 Mbit/s	24 Mbit/s	
フレームレート	30 fps	30 fps	
マクロブロック数	40,500	244,800	
要求処理時間 [μs]	24.69	4.08	
処理時間 (オーバーヘッド) [μs]	VLD/IQ	5.03 (1.53)	3.44 (1.13)
	IDCT	11.18 (1.18)	5.38 (0.38)
	MC	7.96 (1.53)	3.78 (1.13)
	TOTAL	24.16 (4.23)	3.78 (—)

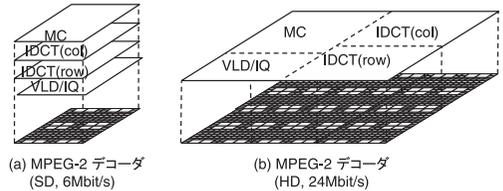


図 22 MPEG-2 デコーダのフレームサイズ拡張に対するマッピング

Fig. 22 Mapping scalability for the enhancement of frame size on MPEG-2 decoder.

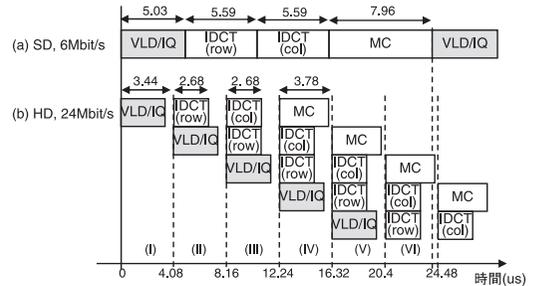


図 23 MPEG-2 デコーダにおける 1 マクロブロックの処理時間とそのスケジューリング

Fig. 23 Process scheduling for the enhancement of frame size on MPEG-2 decoder.

れにより、1 マクロブロックの処理時間は、図 23(b) に示すように 4.08 μs 以下になり、このときの再構成可能セルアレーのサイズは、拡張前の 6 倍になる。また、各コンテキストを空間展開したことで動的再構成のオーバーヘッドがなくなるため、表 16(b) に示すように、性能拡張後のオーバーヘッドが小さくなっている。

ここで、図 23(b) に示すように、各コンテキストを時分割実行せず空間展開すると、協調動作するホストプロセッサにはより高い性能が求められる。SD, 6 Mbit/s デコードの場合、図 23(a) から、ホストプロセッサによる 1 マクロブロック分のヘッダ解析と MC 動きベクトル計算は 19.14 μs 以内に完了すればよい。

ところが、HD, 24 Mbit/s デコードの場合、図 23 (b) に示すように、 $4.08 \mu\text{s}$ 以内に 1 マクロブロック分のヘッダ解析と MC 動きベクトルの計算を行うだけの性能が求められる。

5.2 機能拡張

ARAM 上に、図 24 に示すようなポリフェーズ構成の等分割フィルタバンクを実現する。このようなフィルタバンクは MPEG Audio でも使用されており、オーディオ信号を 32 のサブバンドに分割し、周波数分割には 512 タップのポリフェーズフィルタバンクを適用している [25]。

図 24 に示す 1 サンプル 16 ビットの分割フィルタを実現する。1 チャンネル 16 タップの FIR フィルタは、16 ビット乗算、32 ビット加算、16 タップ分のフィルタ係数テーブル、遅延バッファ、カウンタ等の制御回路で構成される。このうち制御回路を除いて、基本セル 9 個、乗算セル 4 個、レジスタセル 4 個にマッピングされ、16 サイクルで処理を行う。この FIR フィルタを 32 個の各サブバンドに対して繰り返し使用することで、32 サンプルを 512 サイクルで処理できる。

ARAM の再構成可能アレーは、乗算セル数が 16 個であるため、4 チャンネル分の並列処理が可能である。マッピング結果を表 17 に示す。これにより、サンプリング周波数が 44.1 kHz である MPEG Audio の分割フィルタとして、4 ストリーム分の並列処理が可能

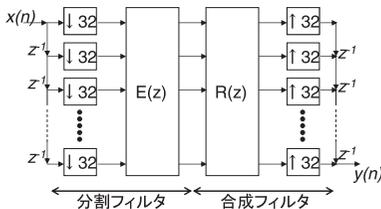


図 24 ポリフェーズ構成の 32 チャンネル等分割フィルタバンク

Fig. 24 General structure of 32-channel uniform polyphase filter bank.

表 17 フィルタバンクのマッピング結果

Table 17 Implementation result of filter bank.

使用セル数	基本セル	44 個
	乗算セル	16 個
	レジスタセル	16 個
	メモリセル	0 個
最大動作周波数	88.9 MHz	
同時処理チャンネル数	4 ch	
処理性能	32samples / 512cycles	

となる。

6. む す び

本論文では、ARAM が複数の動画復号処理を選択的に実現できることを示すため、MPEG-2 デコーダ、MPEG-4 デコーダ、H.263 デコーダを設計対象として、動画復号処理のマッピングについて述べた。各処理過程のマッピング方針を定義することで、マクロセルを用いて複数の動画復号処理がマッピングできることを示した。更に、再構成可能セルアレーのスケラビリティと、動画復号処理の画素並列性を生かした性能拡張について述べ、高性能化要求に対して柔軟に対応できることを示した。また、オーディオ処理にも用いられるフィルタバンクのマッピングについて述べ、ARAM が動画復号処理以外に対しても有効であることを示した。

本論文は、ARAM の有効性評価に主眼を置くため、現時点では大半の処理を手動で行っている。今後は、制御データフローグラフの生成にビヘイビア合成技術を用いることや、再構成可能セルアレー上へのマッピングにおけるマクロセルを整備していくことで、アプリケーション設計フローの自動化を進めていく。また、ARAM の有効性に関して、実装面積だけでなく消費電力についても評価を行っていく。

謝辞 本論文の執筆にあたり、貴重な御意見を頂きました関西学院大学石浦葉岐佐教授、シャープ(株)山田晃久氏に感謝致します。また、動画復号処理のマッピングを検討するにあたり、議論に参加して頂いた大阪大学橋本亮司氏に感謝致します。

文 献

- [1] R. Hartenstein, "A decade of reconfigurable computing: A visionary retrospective," Proc. Design, Automation and Test in Europe 2001 (DATE2001), pp.642-649, March 2001.
- [2] Y. Mitsuyama, K. Takahashi, R. Imai, M. Hashimoto, T. Onoye, and I. Shirakawa, "Area-efficient reconfigurable architecture for media processing," IEICE Trans. Fundamentals, vol.E91-A, no.12, pp.3651-3662, Dec. 2008.
- [3] P. Mulholland, K. Ide, and T. Sato, "C-based hardware design platform for a dynamically reconfigurable processor," Proc. High Performance Embedded Computing Workshop, Sept. 2005.
- [4] W.H. Chen, C.H. Smith, and S.C. Fralick, "A fast computational algorithm for the discrete cosine transform," IEEE Trans. Commun., vol.COM-25, no.9, pp.1004-1009, 1977.

- [5] V. Betz and J. Rose, "VPR: A new packing placement and routing tool for FPGA research," Proc. Field Programmable Logic and Applications (FPL 1997), pp.213-222, Sept. 1997.
- [6] T. Demura, T. Oto, K. Kitagaki, S. Ishiwata, G. Otomo, S. Michinaka, S. Suzuki, N. Goto, M. Matsui, H. Hara, T. Nagamatsu, K. Seta, T. Shimazawa, K. Maeguchi, T. Odaka, Y. Uetani, T. Oku, T. Yamakage, and T. Sakurai, "A single-chip MPEG2 video decoder LSI," ISSCC Digest of Technical Papers, pp.72-73, Feb. 1994.
- [7] Y. Okada, T. Nakamoto, H. Gunji, M. Hase, M. Oku, Y. Tsuboi, H. Mizosoe, K. Imazawa, T. Saito, P.D. Vecchio, S. Cismas, K. Monsen, G. Haber, and K. Chan, "An 80mm² MPEG2 audio/video decode LSI," ISSCC Digest of Technical Papers, pp.264-265, Feb. 1997.
- [8] S. Khawam, I. Nousias, M. Milward, Y. Yi, M. Muir, and T. Arslan, "The reconfigurable instruction cell array," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol.16, no.1, pp.75-85, Jan. 2008.
- [9] A. Major, Y. Yi, I. Nousias, M. Milward, S. Khawam, and T. Arslan, "H.264 decoder implementation on a dynamically reconfigurable instruction cell based architecture," Proc. System On Chip Conference, pp.49-52, Sept. 2006.
- [10] 中村拓郎, 天野英晴, 長谷川揚平, 遠山 治, "動的リコンフィギュラブルプロセッサを用いた動画アプリケーションの機能分割実装," 信学技報, RECONF2006-8, May 2006.
- [11] M. Motomura, "A dynamically reconfigurable processor architecture," Proc. Microprocessor Forum, Oct. 2002.
- [12] B. Mei, F.J. Veredas, and B. Masschelein, "Mapping an H.264/AVC decoder onto the ADRES reconfigurable architecture," Proc. Field Programmable Logic and Applications (FPL 2005), pp.622-625, Aug. 2005.
- [13] B. Mei, S. Vernalde, D. Verkest, H.D. Man, and R. Lauwereins, "ADRES: An architecture with tightly coupled VLIW processor and coarse-grained reconfigurable matrix," Proc. Field Programmable Logic and Applications (FPL 2003), pp.61-70, Sept. 2003.
- [14] M.K.A. Ganesan, S. Singh, F. May, and J. Becker, "H.264 decoder at HD resolution on a coarse grain dynamically reconfigurable architecture," Proc. Field Programmable Logic and Applications (FPL 2005), pp.467-471, Aug. 2005.
- [15] PACT XPP Technologies, "XPP-III processor overview white paper," July 2006.
- [16] S.E.-D. Habib, M.B. Abdelhalim, A.E. Salama, A. Darwish, A.M. Galal, and E.A. Talkhan, "FPGA-based MPEG2 decoder," Proc. International Conference on Microelectronics, pp.194-197, Dec. 2002.
- [17] Altera Corporation, "FLEX 10KE embedded programmable logic devices data sheet," ver. 2.5, Jan. 2003.
- [18] L.V. Agostini, A.P.A. Filho, V.S. Rosa, E.A. Berriel, T.G.S. Santos, S. Bampi, and A.A. Susin, "FPGA design of a H.264/AVC main profile decoder for HDTV," Proc. Field Programmable Logic and Applications (FPL 2006), pp.501-506, Aug. 2006.
- [19] Xilinx Inc., "Virtex-II pro and virtex-II pro X platform FPGAs: Complete data sheet," Nov. 2007.
- [20] K. Denolf, A.C. Rus, R. Turney, P. Schumacher, and K. Vissers, "Memory efficient design of an MPEG-4 video encoder for FPGAs," Proc. Field Programmable Logic and Applications (FPL 2005), pp.391-396, Aug. 2005.
- [21] Xilinx Inc., "Virtex-II platform FPGAs: Complete data sheet," Nov. 2007.
- [22] O. Lehtoranta, E. Salminen, A. Kulmala, M. Hännikäinen, and T.D. Hämäläinen, "A parallel MPEG-4 encoder for FPGA based multiprocessor SoC," Proc. Field Programmable Logic and Applications (FPL 2005), pp.380-385, Aug. 2005.
- [23] Altera Corporation, Stratix Device Handbook, 2005.
- [24] A. Kulmala, T.D. Hämäläinen, and M. Hännikäinen, "Reliable GALS implementation of MPEG-4 encoder with mixed clock FIFO on standard FPGA," Proc. Field Programmable Logic and Applications (FPL 2006),
- [25] M. Kahrs and K. Brandenburg, Applications of Digital Signal Processing to Audio and Acoustics, Kluwer Academic Publishers, 1998.

付 録

1. 再構成可能アーキテクチャの詳細

1.1 配線部の詳細

メモリスルの配線部を図 A・1 に示す。

1.2 機能部の詳細

乗算セルの機能部の構成と構成情報を、それぞれ図 A・2 と表 A・1 に示す。機能部に 8 ビット乗算器をもち、出力が 16 ビットとなるため、ワード出力として 2 本のワード線を選択する。

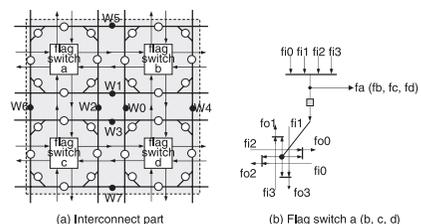


図 A・1 メモリスルの配線部

Fig. A.1 Interconnect part of memory cell.

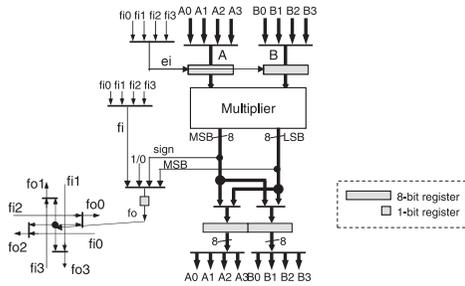


図 A-2 乗算セルの機能部の構成
Fig. A-2 Organization of function part of multiplication cell.

表 A-1 乗算セルの構成情報

Table A-1 Configuration data for multiplication cell.

ビット数	ユニット	構成
5	入出力レジスタ	パイプライン、ライトイネーブル、定数、未使用（バイパス）
2	乗算器	ワード線入力（の符号）
4	フラグ出力	出力選択 ($f_{o0} \sim f_{o3}$)
4	フラグ入力	入力選択 (f_i, e_i)
4	ワード入力	ポート A, B への入力選択
6	ワード出力	出力線の選択または Hi-Z 出力

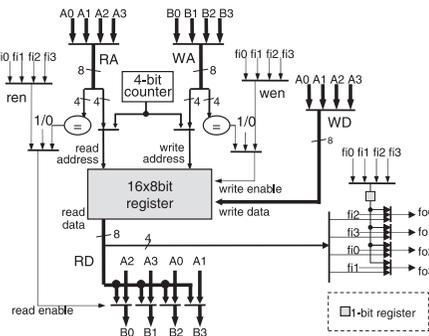


図 A-3 レジスタセルの機能部の構成
Fig. A-3 Organization of function part of register cell.

レジスタセルの機能部の構成と構成情報を、それぞれ図 A-3 と表 A-2 に示す。16 ワードの 2-port レジスタファイルをもち、書込みアドレス (WA) とデータ (WD)、読出しアドレス (RA) とデータ (RD) に計 4 本のワード線を選択する。4 ビットカウンタを用いて遅延器としても機能する。

メモリセルの機能部の構成と構成情報を、それぞれ図 A-4 と表 A-3 に示す。256 ワードの 1-port メモリを有し、アドレス (A)、書込みデータ (WD) は図 A-1 の W0-W7 から選択され、読出しデータ (RD) は W0-W3 から選択される。

表 A-2 レジスタセルの構成情報

Table A-2 Configuration data for register cell.

ビット数	ユニット	構成
3	ライトイネーブル	信号選択
3	リードイネーブル	信号選択
4	定数	イネーブルアドレス用定数
3	フラグ入力	入力選択
6	フラグ出力	出力選択 ($f_{o0} \sim f_{o3}$)
6	ワード入力	ポート WA, WD, RA への入力選択
3	ワード出力	出力線の選択または Hi-Z 出力

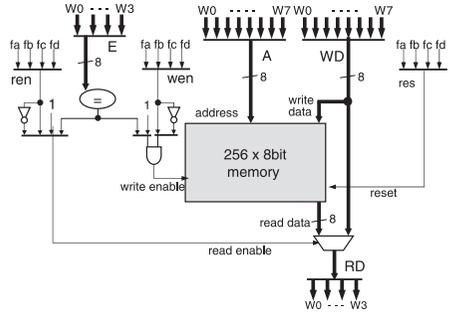


図 A-4 メモリセルの機能部の構成
Fig. A-4 Organization of function part of memory cell.

表 A-3 メモリセルの構成情報

Table A-3 Configuration data for memory cell.

ビット数	ユニット	構成
4	ライトイネーブル	信号選択
4	リードイネーブル	信号選択
8	定数	イネーブルアドレス用定数
8	フラグ入力	入力選択 (f_a, f_b, f_c, f_d)
8	フラグ出力	フラグスイッチ a~d (図 A-1) での出力選択 ($f_{o0} \sim f_{o3}$)
8	ワード入力	ポート A, WD, E への入力選択
3	ワード出力	出力線の選択または Hi-Z 出力

2. マクロセルの構成例

2.1 ビットストリームシフタ

ビットストリームシフタのマクロセルを図 A-5 に示す。図中の網掛けの部分が機能している回路である。基本セルを三つ連結させ、図 A-6 に示す操作によって +1~+8 ビットシフト機能を実現する。

2.2 飽和処理とミスマッチ制御

飽和処理とミスマッチ制御のマクロセルを、それぞれ図 A-7(a), (b) に示す。飽和処理は、 C_a1 による“大小比較”と C_a2 による“条件分岐”のマクロセルの組合せで実現する。ミスマッチ制御は、 C_b1-C_b2 による“累算”、 C_b3-C_b5 による“ビット結合”、セル C_b6 による“条件分岐”のマクロセルの組合せで実現する。

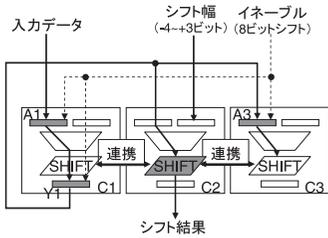


図 A-5 ビットストリームシフタのマクロセル
Fig. A-5 Mapping macro of bitstream shifter.

1. 入力データを前後 8 ビットのデータも含めてレジスタ A1, Y1, A3 に格納.
2. シフト幅が +1~+3 ビットの場合, C1 と C2 による連携シフトの結果を C2 から出力.
3. シフト幅が +4~+8 ビットの場合, イネーブル信号と新たな 8 ビットデータでレジスタ A1, Y1, A3 のデータを更新し (+8 ビットシフトの実行), C2 と C3 による -4~0 ビットの連携シフト結果を +4~+8 ビットのシフト結果としてセル C2 から出力.

図 A-6 ビットストリームシフタの動作
Fig. A-6 Process of MC.

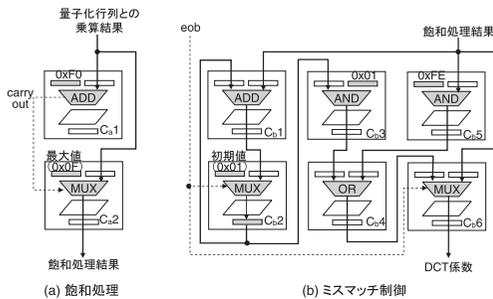


図 A-7 飽和処理とミスマッチ制御のマクロセル
Fig. A-7 Mapping macros of saturation and mismatch control.

(平成 21 年 7 月 23 日受付, 12 月 8 日再受付)



密山 幸男 (正員)

平 10 阪大・工・情報システム卒・平 12 同大学院博士前期課程了。平 14 同大学院博士後期課程退学。同年, 阪大助手, 現在同大助教。再構成可能アーキテクチャとその設計に興味をもつ。IEEE, 情報処理学会, 画像電子学会各会員。



高橋 一真

平 17 阪大・工・情報システム卒。平 19 同大学院博士前期課程了。現在 (株) メガチップスに勤務。



今井林太郎

平 16 阪大・工・情報システム卒。平 18 同大学院博士前期課程了。



橋本 昌宜 (正員)

平 9 京大・工・電子卒。平 13 同大学院博士課程 (通信情報システム専攻) 了。博士 (情報学)。同年, 京都大学情報学研究科助手, 平 16 大阪大学情報科学研究科助教, 現在同准教授。VLSI の設計技術, CAD の研究に従事。IEEE, 情報処理学会各会員。



尾上 孝雄 (正員)

平 3 阪大・工・電子卒。平 5 同大学院博士前期課程了。阪大助手, 講師, 京大助教, 阪大助教を経て, 平 15 同大教授。博 (工)。メディア処理向け組込みシステムのアーキテクチャ, 設計及び実装, 低消費電力化設計等に興味をもつ。IEEE, ACM, 情報処理学会, 映像情報メディア学会各会員。



白川 功 (名誉員: フェロー)

昭 38 阪大・工・電子卒。昭 43 同大学院博士課程了。工博。同年阪大助手, 昭 48 同助教, 昭 62 同教授。平 10 同大型計算機センター長, 平 11 同評議員, 同総長補佐役, 平 13 同大学院工学研究科長・工学部長, 平 15 同定年退官, 同名誉教授 (株) シンセシス取締役会長。平 16 兵庫県立大学大学院応用情報科学研究科教授, 現在に至る。その間, 昭 49~50 カルフォルニア大バークレイ校客員研究員。回路理論, グラフ理論, VLSI の設計自動化, システム VLSI の実装設計, ISO 準拠製品データベース構築などの教育研究に従事。現在, IEEE Life Fellow; ACM, 情報処理学会, 映像情報メディア学会, システム制御情報学会, 計測自動制御学会, エレクトロニクス実装学会各会員。著書: 「演習グラフ理論」, 「回路理論 I, II」(共著, コロナ社) など。