# シングルチップマルチプロセッサ上での マルチメディアアプリケーションの 近細粒度並列処理

# 小高 剛 宮下直久

### 木村啓二 笠原博徳

#### 早稲田大学理工学部電気電子情報工学科

〒 169-8555 東京都新宿区大久保 3-4-1 TEL:03-5286-3371 E-mail: {kodaka,naohisa,kimura,kasahara}@oscar.elec.waseda.ac.jp

近年のマルチメディアコンテンツの増加に伴い, JPEG, MPEG などのメディア系アプリケーションを効率良く処 理できる,低コストかつ低消費電力のプロセッサの開発が望まれている.これらの要求を満たすプロセッサとして,簡 素なプロセッサコアを複数搭載したシングルチップマルチプロセッサが注目を集めている.本稿では,シングルチップ マルチプロセッサのメディア系アプリケーションでの有用性を確かめるため,まず,第一段階として画像圧縮処理の JPEG エンコーディングプログラムを用い,その処理単位が最終的に8×8 画素のプロックになることに注目し,その 8×8 画素プロックの処理に近細粒度並列処理を施し OSCAR 型シングルチップマルチプロセッサ上で性能評価を行っ た.その結果,シンプルなシングルイシュープロセッサコアを4基搭載した OSCAR 型シングルチップマルチプロセッ サシステムは4イシュースーパースカラプロセッサの UltraSPARC-II 相当のプロセッサコアを1基搭載するシステム に対し約2.32 倍の速度向上が得られた.

# Near Fine Grain Parallel Processing

# on Multimedia Application for Single Chip Multiprocessor

Takeshi Kodaka , Naohisa Miyashita , Keiji Kimura and Hironori Kasahara

Dept. of Electrical, Electronics and Computer Engineering, Waseda University

3-4-1 Ohkubo Shinjuku-ku, Tokyo 169-8555, Japan Tel: +81-3-5286-3371 E-mail: {kodaka,naohisa,kimura,kasahara}@oscar.elec.waseda.ac.jp

With the recent increase of multimedia contents, such as JPEG and MPEG data, low cost and low power consumption processors that can process these multimedia contents efficiently are expected. In such microprocessors, single chip multiprocessor architecture having simple processor cores is attracting much attention. Considering the above facts, this paper evaluate a JPEG encoding program on OSCAR type single chip multiprocessor architecture using near fine grain parallel processing for  $8 \times 8$  pixel block that is a fundamental part of JPEG algorithm. The evaluation shows an OSCAR type single chip multiprocessor having four single-issue simple processor cores gives 2.32 times speedup than four-issue UltraSPARC-II type super-scaler processor.

# 1 はじめに

近年のマルチメディアコンテンツの増加に従い, JPEG, MPEGなどのメディア系アプリケーション を効率良く処理できる,低コストかつ低消費電力の プロセッサの開発が望まれている.このようなニー ズに対応するために CPU ベンダーからは命令セッ トアーキテクチャにメディア用命令セットを追加し メディア用処理能力を向上させる試みが行われてい る<sup>1)</sup>.

また,上記の要求を満たす別のアプローチとして, 簡素なプロセッサコアを複数搭載したシングルチッ プマルチプロセッサが注目を集めており,シングル チップマルチプロセッサを用いてメディア系アプリ ケーションの処理の高速化を行う試みもなされてい る.<sup>2)~5)</sup>

本稿ではマルチメディアアプリケーションのシン グルチップマルチプロセッサ上での性能を評価するた めに,まず,第一段階として画像圧縮処理のJPEG エンコーディングを用い,その処理単位が最終的 に8×8 画素のブロックになることに注目し,その 8×8 画素ブロックの処理に近細粒度並列処理を施 し OSCAR 型シングルチップマルチプロセッサ<sup>7)</sup> 上で性能を評価した. 以降,2節にて近細粒度並列処理,3節にて評価 プログラムの JPEG エンコーディングアルゴリズ ム,4節にて評価対象アーキテクチャついて説明し た後,5節にて OSCAR 型シングルチップマルチプ ロセッサ上で行った性能評価について述べる.

#### 2 近細粒度並列処理<sup>6)</sup>

近細粒度並列処理とは基本ブロック内のステート メント間の並列性を利用する並列処理である.ステー トメントをプロセッサエレメント (PE) に割り当て る際には,スケジューリング手法としてデータ転送 オーバヘッドを考慮し実行時間を最小化するヒュー リスティックアルゴリズムである CP/DT/MISF法, CP/ETF/MISF法,ETF/CP法,あるいはDT/CP 法<sup>8)</sup>の4手法を適用し最良のスケジュールを選ぶ.

# 3 JPEGエンコーディングアルゴ リズム

本稿では, JPEG エンコーディングアルゴリズム として整数演算ベースラインシステムについて考 える.

ベースラインシステムの JPEG エンコードアル ゴリズムは,入力信号を輝度と色差成分信号に変換 する YCbCr 変換処理,画像信号を空間周波数成分 に変換する離散コサイン変換,ベクトル量子化を行 う量子化処理,エントロピー符号化を行う可変長符 号化処理の4つの処理から構成される.また,これ らの処理は,8×8 画素からなるブロックを単位と して行われる.

本節では,評価に用いたベースラインシステムの JPEG エンコードアルゴリズムについて説明する.

#### 3.1 YCbCr 変換

YCbCr 変換は,入力信号を JPEG 処理系の輝度 と色差の成分に変換する処理である.入力信号に より変換式は変化するが本稿では一般的な R(赤), G(緑), B(青)の三原色からの変換について述べる.

RGB から YCbCr 変換は

 $Y_{ij} = R_{ij} * 0.2990 + G_{ij} * 0.5870 + B_{ij} * 0.1140$  $Cb_{ij} = -R_{ij} * 0.1684 - G_{ij} * 0.3316 + B_{ij} * 0.5000$  $Cr_{ij} = R_{ij} * 0.5000 - G_{ij} * 0.4187 - B_{ij} * 0.0813$ で表される.

この式より,変換式には各演算間にデータの依存 が存在せずこれらの演算は並列に処理できる.

#### 3.2 離散コサイン変換

離散コサイン変換 (DCT) は,データを周波数成 分に変換する直交変換の1つである.現在 DCT は, 画像符号化の基本技術となっており多くの符号化標 準規格で採用されている.

なお,ここでは1次元 DCT のアルゴリズムにつ いて説明するが,2次元への拡張は,1次元 DCT を 行方向に対して1回処理を行った後,列方向へ1回 処理を行うことにより容易に実現できる.

3.2.1 基本式

DCT の基本式は , 入力データ列 f(u):(u = 0..N)に対する DCT 出力を F(u) とするとき

$$F(u) = \sqrt{\frac{2}{N}}C(u) \sum_{x=0}^{N-1} f(x) \cos \frac{(2x+1)u\pi}{2N}$$
$$C(u) = \begin{cases} 1/\sqrt{2} & (u=0)\\ 1 & (u\neq 0) \end{cases}$$

と表される.この場合,計算量は,加算・乗算とも に N<sup>2</sup> となり,計算量が大きく実用的ではないため 計算量をおさえるために次節に述べるようなバタフ ライ演算法が広く利用されている.

#### 3.2.2 バタフライ演算法

計算量をおさえるため,FFT(高速フーリエ変換) のようなバタフライ演算による高速演算法が広く利 用されている.代表的なアルゴリズムに Chen のア ルゴリズム<sup>10)</sup>があり次式で表される.

$$\begin{array}{rcl} A_N &=& P_N \left[ \begin{array}{cc} A_{N/2} & 0 \\ 0 & R_{N/2} \end{array} \right] \left[ \begin{array}{cc} I_{N/2} & \tilde{I}_{N/2} \\ \tilde{I}_{N/2} & -I_{N/2} \end{array} \right] \\ P_N & : 巡回行列 \end{array}$$

$$R_N$$
  $ilde{R}_N$ の要素  $ilde{r}_N(i,k)$ を  
行と列について逆に並べた行列

この方法により,計算量が,加算: $(3N/2)(\log_2 N - 1) + 2$ ,乗算:  $N \log_2 N - 3N/2 + 4$  に削減される.

本稿の評価では,離散コサイン変換処理にバタフ ライ演算法を用いた.バタフライ演算のタスクグラ フを図1に示す.図中(a)は,1次元DCT演算の近 細粒度タスクグラフを表している.また(b)は,2 次元DCT処理を行うときの各行・列処理のデータ 依存を示すグラフであり各ノードは(a)のタスクグ ラフを表している.

図 1(a) に示したようにバタフライ演算法では,1 次元演算において並列処理できる部分が存在する.



また,2次元処理では図1(b)のように1次元処理を 各行で行方向に一回,各列で列方向に一回処理を行 うことで実現させるため,行方向処理中では各行処 理の間でのデータ依存が無く並列に実行できる.同 様に列方向についても同様にデータ依存がないため 並列に実行できる.

#### 3.3 量子化

量子化処理は, DCT 処理後の値を予め定められ た量子化ステップサイズに対する比率を整数値で求 ることであり次式で表される.

$Sq_{ij}$	=	$round\left(\frac{S_{ij}}{q_{ij}}\right)$				
Q	=	$ \left(\begin{array}{c} q_{11}\\ q_{21} \end{array}\right) $	$q_{12} \\ q_{22}$	· · · ·	$\begin{array}{c} q_{1j} \\ q_{2j} \end{array}$	
		$\begin{pmatrix} & \cdots & \\ & q_{i1} \end{pmatrix}$	$q_{i2}$		$q_{ij}$	J

ここで Sq<sub>ij</sub> は量子化値, S<sub>ij</sub> は入力となる DCT係数, q<sub>ij</sub> は量子化ステップサイズを示し, round() 演算は, 四捨五入を行う演算子である.

本稿の評価で用いた量子化テーブルは MediaBench<sup>9)</sup> に含まれている JPEG エンコーディングプログラ ムにおけるデフォルト値を用いる.

量子化演算では,式中の各要素が一意に決まるた

め各演算間にデータ依存が存在せず,これらの演算 は並列に処理できる.

#### 3.4 可変長符号化

量子化後,エントロピー符号化のため可変長符号 化処理が行われる.ベースラインシステムでの可変 長符号化は,8×8の量子化値に対して DC 成分と AC 成分に別けて行われる.

#### 3.4.1 DC 成分の可変長符号化

DC 成分とは,入力となる 8 × 8 の量子化値中の 最左上の値,すなわち座標(0,0)の値を示す.DC 成 分の可変長符号化には予測が用いられ,現在の 8 × 8 の量子化値の DC 成分に対し隣接する直前の 8 × 8 の量子化値の DC 成分との差分をとりその差分値が 符号化される.

DC 成分の可変長符号化を行うには,上記 DC 差 分値の絶対値において MSB 側から見て始めにビッ ト"1"が立つ場所が LSB 側から数えて何ビット目 であるかという数値 SSSS を求め,SSSS に対応 する可変長符号を求める.次に,DC 差分値が正の ときは DC 差分値の LSB から SSSS ビット目まで を,負のときは (DC 差分値-1)の LSB から SSSS ビットまでを付加ビットとして,(可変長符号+付 加ビット)の形で可変長符号化を行う.

#### 3.4.2 AC 成分の可変長符号化

AC 成分とは,8×8の量子化値のDC 成分以外 の63個の値を示す.AC 成分は,ジグザグスキャン 順にスキャニングされ符号化される.

AC 成分の可変長符号化は,ジグザグスキャン順 に0係数の連続出現回数(RUN)と非0係数の値によ リ行われる.まず,スキャン順に非0係数が出るま でRUN 値をカウントする.ただしRUN 値が16に なった場合は,ZRL符号(0が16個続いたことを示 す符号)を出力しRUN 値を0にリセットしRUN 値 のカウントを再開する.非0係数が現れたらその非 0係数に対してDC 成分の符号化と同様にSSSS値 を求め,非0係数が現れるまでのRUN 値とSSSS 値を用いて2次元可変長符号テーブルから対応する 可変長符号を求める.そして,非0係数が正のとき は非0係数のLSBからSSSSビットを,負のとき は(非0係数-1)のLSBからSSSSビットを付加 ビットとし,(可変長符号+付加ビット)の形で可変 長符号化を行う. 本稿では,可変長符号テーブルは MediaBench に 含まれている JPEG エンコーディングプログラム におけるデフォルト値を用いる.

上記のように, DC 成分の符号化には 8×8 ブロッ ク間の依存が, AC 成分の符号化には非0係数の判 定および直前までの0係数の個数が分からなければ 符号化できないので,8×8 ブロック内で制御依存 およびデータ依存が存在し,そのため並列処理が難 しい.

本稿の評価では,並列性抽出のためAC成分の可 変長符号化処理をPE台数分に分割し処理分割に伴 い分割境界部分のRUN値が分離されてしまうため 符号化の最後に値を修正するための処理を追加している.

ここでは簡単のために分割数を2とし,スキャニ ング順で始めの2~32までを分割ブロック0,後の 33~64を分割ブロック1とし説明する(AC成分は DC成分を抜いた2~64までとなる).

分割ブロック0では,分割しない場合と同様の処 理が行われる.分割ブロック1では,非0係数を可 変長符号化するときに,その非0係数の可変長符号 化が分割ブロック内で初めてかどうか判定する.も し,その分割ブロック内で初めての可変長符号化を 行う非0係数であった場合,そのときのRUN 値と 非0係数の位置および非0係数のSSSS 値を保存 しておく.

分割ブロック0,1での可変長符号化が終了した ら,符号値を修正するために,分割ブロック0での 最後のRUN値と保存した分割ブロック1で初めに 出現した非0係数までのRUN値との和を取る.こ の和をとったRUN値と保存しておいた分割ブロッ ク1で初めに出現した非0係数のSSSS値をもと に可変長符号を求め可変長符号化を行い分割ブロッ ク1における初めの非0係数の位置に求めた可変長 符号を上書する.

以上が,分割数2での可変長符号化である.分割 数を増やすには同様に分割ブロック内での最初の非 0係数出現までのRUN値と非0係数の位置および 非0係数のSSSS値を保存し,各分割ブロックの 可変長符号化処理が終了したら前の分割ブロックの 最後のRUN値の和を取りその和を取ったRUN値 で可変長符号を求め直して符号を作り直す部分を増 やすことにより対応できる.

# 4 対象アーキテクチャ

本節では評価対象アーキテクチャである OSCAR 型シングルチップマルチプロセッサのアーキテクチャ について述べる.また,比較対象とする UltraSPARC-II 相当プロセッサを CPU コアとするアーキテクチャ についても述べる.

# 4.1 OSCAR 型シングルチップマルチプ ロセッサアーキテクチャ

OSCAR 型アーキテクチャ<sup>7)</sup> によるシングルチッ プマルチプロセッサの構成を図 2 に示す.OSCAR 型アーキテクチャでは,CPU,データ転送ユニット (DTU),ローカルプログラムメモリ(LPM),ロー カルデータメモリ(LDM),そしてデュアルポート メモリで構成された分散共有メモリ(DSM)をもつ プロセッシングエレメント(PE)と,集中共有メモ リ(CSM)が相互接続網を介して接続されている.



#### 図 2: OSCAR 型 SCM アーキテクチャ

本評価では各メモリへのアクセスレイテンシを, LPM に対して1クロック,LDM に対して1クロッ ク,DSM は,自 PE 上の DSM へのアクセスに対し て1クロック,他 PE 上の DSM へのアクセスに対 して4クロック,CSM に対して20クロックかかる とする.また、相互接続網は3本のバスを使用する.

プロセッサコアには,SPARC V9 規格に準拠した 表1に示す単一命令発行のシンプルなRISC プロセッ サを使用する.命令レイテンシは,UltraSPARC-II と同様とする.プロセッサ数は1~4 基とし,LDM の総容量は,1Mbyte,DSM は16Kbyteとする.

#### 4.2 比較対象プロセッサコア

比較対象プロセッサコアとして表1に示す in-order 4 命令発行アーキテクチャの UltraSPARC-II 相当の 演算器構成を持つプロセッサを用いる.本評価では, データサイズが小さく使用するデータが全てキャッ シュにのるため,このプロセッサコアを持つシステ ムでは,メモリアクセスが必ず1クロックで可能な 理想的なメモリシステムを仮定する.また,命令の レイテンシは4.1節で述べた OSCAR 型アーキテク チャと同様とする.

表 1: プロセッサコア仕様

	SCM	US-II Type	
パイプライン段数	9		
同時命令発行数	1	4	
IEU	1	2	
FPU	1	2	
LSU	1	1	
命令発行タイプ	in-order		

## 5 性能評価

本節では, JPEG エンコーディングアルゴリズム に近細粒度並列処理を適用して性能評価を行った結 果を述べる.

対象アーキテクチャは,4節で述べたシングルイ シュープロセッサコアを持つ OSCAR 型シングル チップマルチプロセッサアーキテクチャ(プロセッ サ数:1~4基),および,比較対象とする4イシュー スーパースカラプロセッサコアの UltraSPARC-II 相当のプロセッサコアを持つアーキテクチャ(プロ セッサ数:1基)である.

評価は,YCbCr 変換,離散コサイン変換,量子 化,可変長符号化の個別評価と,YCbCr 変換から可 変長符号化までを通して行う全体評価により行い, 入力データは,8×8RGB データとし,輝度成分の エンコーディングのみを対象とした.なお,可変長 符号化では,3.4節で述べたように並列性抽出のた めPE 台数分分割し後処理を加えたプログラムを使 用する.

この結果を図3に示す.図3では,YCBCrは YCbCr 変換,DCT は離散コサイン変換,Quant は 量子化,VLC は可変長符号化をそれぞれ個別評価し た結果であり Total は YCbCr 変換から可変長符号 化までの全体評価を行った結果である.評価結果は, シングルチップマルチプロセッサアーキテクチャで のシングルイシュープロセッサ1基での実行結果に 対する速度向上率として表す.



#### 図 3: 評価結果

YCbCr 変換では,ステートメント間のデータ依 存が存在しない.そのため,PE数4台にて,PE数 1台に対し約3.50倍というスケーラブルな速度向上 を示した.また,UltraSPARC-II相当のプロセッサ コアに対しても約3.18倍の速度向上が得らた.

離散コサイン変換では, PE数4台にて, PE数 1台に対し約2.58倍, UltraSPARC-II相当のプロ セッサコアに対して約2.42倍の速度向上が得られた.アルゴリズム上, 行方向処理から列方向処理へ 移行するとき約半分のデータが自PEから他のPE へ転送されることの影響によりYCbCr変換と比較 して速度向上率が低くなってしまったと考えられる.

量子化では YCbCr 変換と同様に,ステートメン ト間のデータ依存が存在しない.そのため, PE 数 4 台にて, PE 数 1 台に対し約 3.59 倍というスケー ラブルな速度向上を示した.また, UltraSPARC-II 相当のプロセッサコアに対しても約 3.26 倍の速度 向上が得らた.

可変長符号化では,分割処理により並列実行可能 となり速度向上を示した.PE数4台にて,PE数1 台に対し約2.79倍,UltraSPARC-II相当のプロセッ サコアに対して約1.72倍の速度向上が得られた.

全体を通して行った性能評価では, PE数4台に て, PE数1台に対し約2.90倍, UltraSPARC-II相 当のプロセッサコアに対し約2.32倍の速度向上が 得らた.

UltraSPARC-II 相当のプロセッサコアを用いた

アーキテクチャの場合速度向上が少ないのは,積和 演算が多用されるため命令に乗算,除算のマルチサ イクル命令が瀕出するのでパイプラインがストール しIPCが向上しなかったためである.

### 6 まとめ

本稿では、マルチメディアアプリケーションの画 像圧縮処理のひとつである JPEG エンコーディング について近細粒度並列処理を適用し OSCAR 型シン グルチップマルチプロセッサ上での性能評価を行っ た.その結果、シングルイシュープロセッサを4基 搭載した OSCAR 型シングルチップマルチプロセッ サシステムでは同システムにシングルイシュープロ セッサを1基搭載したシステムに対し約2.90倍、4 イシューの UltraSPARC-II 相当プロセッサコアを 1基搭載したシステムに対し約2.32倍の性能が得ら れた.

これにより, JPEG エンコーディングアルゴリズ ムにおいて近細粒度並列処理を適用することの有用 性が確認できた.

今後の課題として,8×8 画素ブロック間の並列 性と8×8 画素ブロック内の近細粒度並列性を階層 的に用いるシングルチップマルチプロセッサ上での マルチグレイン並列処理や MPEG-2 や JPEG2000 などに用いられるマルチメディアアルゴリズムでの 並列処理を行うことがあげられる.

#### 7 謝辞

本研究の一部は,STARC「自動並列化コンパイ ラ協調型シングルチップマルチプロセッサの研究」 及び経済産業省ミレニアムプロジェクト「アドバン スト並列化コンパイラ」により行われた.本論文作 成にあたり,有益なコメントをいただいたSTARC 研究員である,STARC小澤時典氏,平田雅規氏, 東芝浅野滋徳氏,富士通高橋宏政氏,ソニー倉田 隆弘氏,松下高山秀一氏に感謝致します.

# 参考文献

- [1] S.K.Raman, V.Pentkovski and J.Keshava,Intel Corporation, "Implementing Streaming SIMD Extensions on the Pentium III Processor", IEEE MICRO,July-August.2000
- [2] M.Edahiro, S.Matsushita and M.Yamashina, N.Nishi, "A Single-Chip Multiprocessor

for Smart Terminals", IEEE MICRO, July-August.2000

- [3] L.A.Barroso, K.Gharachorloo, R.McNamara, A.Nowatzyk, S.Qadeer, B.Sano,S.Smith, R.Stets and B.Verghese, "Piranha: A Scalable Architecture Based on Single-Chip Multiprocessing" In Proc. ISCA 00, Jun.2000
- [4] P.Sriram, S.Sudharsanan and Amit Gulati, "MPEG-2 Video Decompression on a Multiprocessing VLIW Microprocessor", Sun Microsystems
- [5] E.Iwata and K.Olukotun, "Exploiting Coarse-Grain Parallelism in the MPEG-2 Algorithm", Stanford University Computer System Lab. Technical Report CSL-TR-98-771, Sep.1998
- [6] H.Kasahara, M.Obata and K.Ishizaka, "Automatic Coarse Grain Task Parallel Processing on SMP using OpenMP", Proc. of 13th International Workshop on Languages and Compilers for Parallel Computing (LCPC'00), Aug.2000.
- [7] 木村、加藤、笠原、"近細粒度並列処理用シング ルチップマルチプロセッサにおける プロセッ サコアの評価"、情報処理学会論文誌、Vol. 42, No. 4, Apr.2001
- [8] 笠原, "並列処理技術", コロナ社, Jun.1991
- [9] C.Lee, M.Potkonjak and W.H.Mangione-Smith "MediaBench: A Tool for Evaluating and Synthesizing Multimedia and Communications Systems", 30th International Symposium on Microarchitecture (MICRO-30), 1997
- [10] W.H.Chen, C.H.Smith and S.C.Fralick, "A Fast Computational Algorithm for the Discrete Cosine Transform", IEEE Trans. Comm., Vol. COM-25, No.9, pp.1004-1009, Sep.1997
- [11] Sun Microelectronics, "UltraSPARC<sup>TM</sup> User's Manual", Jul.1997
- [12] 吉田, 渡辺, "ディジタル画像圧縮の基礎", 日経 BP 出版センター, Jun.1999
- [13] 小野, 鈴木, "JPEG/MPEG2 の技術", オーム 社, Jan.2001
- [14] 笠原,木村,"シングルチップマルチプロセッサ", 特許出願番号第 363702 号