

OSCAR型シングルチップマルチプロセッサ上での JPEGエンコーディングプログラムの マルチグレイン並列処理

小高 剛 内田 貴之
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, uchida, kimura, kasahara}@oscar.elec.waseda.ac.jp

近年の JPEG, MPEG などを用いたマルチメディアコンテンツの増加に伴い, これらマルチメディア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. 特に, 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アーキテクチャは最も有望なアプローチとして注目され研究・開発がなされている. 本論文では, OSCAR 型メモリアーキテクチャシングルチップマルチプロセッサ上での JPEG エンコーディングプログラムのマルチグレイン並列処理手法を提案すると共に, 提案手法を適用した JPEG エンコーディングプログラムの OSCAR 型メモリアーキテクチャシングルチップマルチプロセッサ上で評価を行なった. その結果, シンプルな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では, 逐次実行に対して約 3.59 倍の性能向上が得られ, ほぼ同程度のトランジスタ数であると考えられる UltraSPARC-II 相当の 4 イシュースーパースカラプロセッサをコアとしたアーキテクチャに対しても約 2.87 倍の性能向上が得られた.

Multigrain Parallel Processing for JPEG Encoding Program on an OSCAR type Single Chip Multiprocessor

TAKESHI KODAKA, TAKAYUKI UCHIDA, 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, uchida, kimura, kasahara}@oscar.elec.waseda.ac.jp

With the recent increase of multimedia contents using JPEG and MPEG, low cost, low power consumption and high performance processors for multimedia have been expected. Particularly, single chip multiprocessor architecture having simple processor cores is attracting much attention to develop such processors. This paper describes multigrain parallel processing scheme for a JPEG encoding program for OSCAR type single chip multiprocessor and its performance. The evaluation shows an OSCAR type single chip multiprocessor having four single-issue simple processor cores gave us 3.59 times speed-up than sequential execution and 2.87 times speed-up than OSCAR type single chip multiprocessor that has a four-issue UltraSPARC-II type super-scaler processor core.

1 はじめに

最近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高性能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 Intel の Pentium III などに搭載されている SSE¹⁾ や, 日立の SH4²⁾, 富士通の FR500³⁾ などのような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.

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高速化を行う試みもなされている. NEC の MP98⁴⁾ では, 1 チップ上に 4CPU を搭載しマルチスレッド処理により高速化をはかっている. また, Stanford 大では, 2 次キャッシュ共有型のシングルチップマルチプロセッサ Hydra⁵⁾ 上でのメディ

アプリケーションの高速化が提案されている⁶⁾。

本論文では、OSCAR 型メモリアーキテクチャシングルチップマルチプロセッサ上での JPEG エンコーディングプログラムのマルチグレイン並列処理手法を提案し、提案手法を適用した JPEG エンコーディングプログラムの OSCAR 型シングルチップマルチプロセッサ上での評価結果を述べる。

以降、2. にてマルチグレイン並列処理、3. にて提案する JPEG エンコーディングプログラムのマルチグレイン並列処理手法、4. にて本論文の評価で用いる OSCAR 型シングルチップマルチプロセッサアーキテクチャ、5. にてマルチグレイン並列処理を適用した JPEG エンコーディングプログラムの 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それぞれ述べる。

2 マルチグレイン並列処理

ここでは、OSCAR 型シングルチップマルチプロセッサで扱うマルチグレイン並列処理技術について述べる。マルチグレイン並列処理とは、ループやサブルーチン等の粗粒度タスク間の並列処理を利用する粗粒度タスク並列処理(マクロデータフロー処理)⁷⁾、ループイタレーションレベルの並列処理である中粒度並列処理、基本ブロック内部のステートメントレベルの並列性を利用する近細粒度並列処理⁸⁾を階層的に組み合わせてプログラム全域に渡る並列処理を行なう手法である。

2.1 粗粒度タスク並列処理⁷⁾

(マクロデータフロー処理)

マクロデータフロー処理では、ソースとなるプログラムを疑似代入文ブロック (BPA)、繰り返しブロック (RB)、サブルーチンブロック (SB) の三種類の粗粒度タスク (マクロタスク (MT)) に分割する。MT 生成後、コンパイラは BPA, RB, SB, 等の MT 間のコントロールフローとデータ依存を解析しそれらを表したマクロフローグラフ (MFG) を生成する。さらに MFG から MT 間の並列性を最早実行可能条件解析により引きだし、その結果をマクロタスクグラフ (MTG) として表現する。その後、コンパイラは MTG 上の MT をプロセッサあるいはプロセッサクラスタ (PC) に割り当てる。

2.2 中粒度並列処理 (ループ並列処理)

PG に割り当てられた MT が Doall 可能な RB である場合、この RB は PG 内のプロセッシングエレメント (PE) に対して、イタレーションレベルで割り当てられ並列実行される。

2.3 近細粒度並列処理⁸⁾

PG に割り当てられた MT が、BPA や中粒度並列処理を適用できない RB である場合、それらはステートメントレベルのタスクに分割され、PG 内の PE により並列処理される。

近細粒度並列処理においては、BPA 内のステートメント、もしくは複数ステートメントから構成される疑似代入文を一つの近細粒度タスクとして定義する。近細粒度タスクは、スタティックスケジューリングされた後、PG 内の各 PE へ割り当てられる。

3 JPEG エンコーディングアルゴリズムのマルチグレイン並列処理手法

ここでは、JPEG エンコーディングアルゴリズムのマルチグレイン並列処理手法について述べる。本論文で扱う JPEG エンコーディングアルゴリズムは、MediaBench⁹⁾ に収録されている JPEG ベンチマークプログラムである “jpeg-v6a” のものをベースとする。まず、JPEG エンコーディングにおける各処理について簡単に述べた後、これらの間のマルチグレイン並列性を利用する手法を述べる。

3.1 JPEG エンコーディングアルゴリズム^{10),11)}

JPEG エンコーディングは以下の 6 つの処理からなる。

1. 8x8-pixel ブロック分割
入力画像データを JPEG 基本処理単位である 8x8-pixel ブロックに分割
2. YCbCr 変換 (YCbCr)
分割された 8x8-pixel ブロックのデータを JPEG のデータフォーマットである YCbCr フォーマットへ変換
3. 2次元離散コサイン変換処理 (DCT)
YCbCr フォーマットへ変換された 8x8 ブロックデータに対し 2次元離散コサイン変換を行

なう。なお、変換後のデータは最左上の1データは直流 (DC) 係数, その他の 63 データは交流 (AC) 係数と呼ばれる

4. 一様量子化 (Quantize)
量子化テーブルを用い DCT 係数に対し 1 次元一様量子化を行なう。
5. 量子化 DC 係数の 1 次元予測 (1-D DC Prediction)
量子化された DC 係数と 1 つ前の 8x8 ブロックの量子化 DC 係数を減算し 1 次元 DC 予測値を計算する
6. エントロピー符合化
1 次元 DC 予測値および量子化 AC 係数を可変長符合を用いてエントロピー符合化を行なう。

上記 6 つの処理を入力画像データが終るまで繰り返して行なわれる。図 1 に JPEG エンコーディング実行ブロック図を示す。

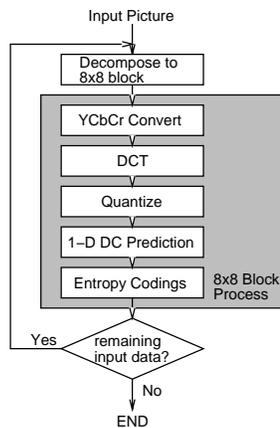


図 1: JPEG エンコーディング実行ブロック図

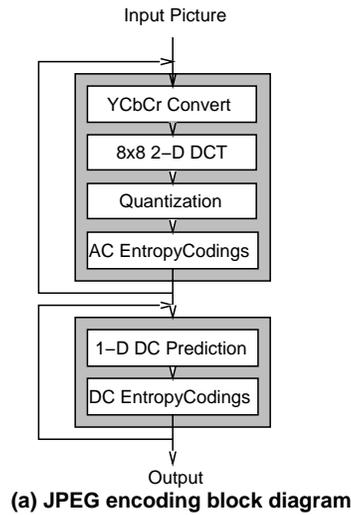
3.2 マルチグレイイン並列化 JPEG エンコーディング

3.1 で述べた基本的な JPEG エンコーディングアルゴリズムでは, 量子化 DC 係数の 1 次元予測 (1-D DC Prediction) で 1 つ前の 8x8 ブロックの量子化 DC 係数を用いるため, 8x8 ブロック間でのデータ依存が存在する。しかし, その他の YCbCr 変換, DCT およびエントロピー符合化処理では 8x8 ブロック内のデータのみ扱うため 8x8 ブロック間の

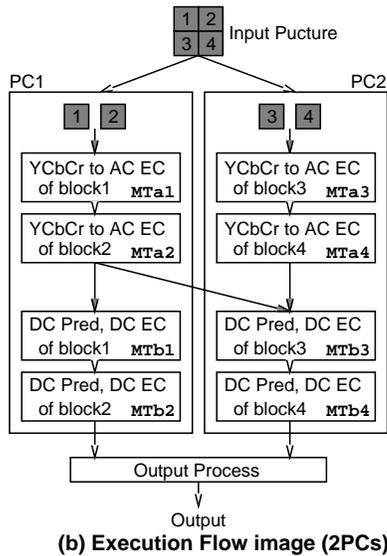
データ依存は存在しない。これらのデータ依存を考慮し, 並列処理を効果的に行なうために 1-D DC Prediction 処理および, 1-D DC Prediction の処理結果の 1 次元 DC 予測値を用いる 1 次元 DC 予測値のエントロピー符合化処理 (DC EC) を図 2(a) のように YCbCr, DCT, Quantize および量子化 AC 係数のエントロピー符合化処理 (AC EC) の後ろに移動する。そして, 8x8 ブロック処理を JPEG エンコーディング基本処理単位として, YCbCr, DCT, Quantize および AC EC を 1 つのマクロタスク (図中 MT_a) として定義し, 同様に 1-D DC Prediction および DC EC も 1 つのマクロタスク (図中 MT_b) として定義する。JPEG エンコーディング処理の実行では, 入力データを 8x8 ブロックに分割後, すべての 8x8 ブロックについて MT_a を実行し, その後 MT_b の実行を行なう。このような処理手順を行なうことにより MT_b を処理開始前にはすべての量子化 DC 係数が計算されているため, 他の PC で割り当てられた MT_b が必要とする量子化 DC 係数を転送することにより MT_b が並列実行可能となる。 MT_a, MT_b の各 PC への割り当ては, 各 PC で処理される MT_a および MT_b の数が等しくなるように割り当てる。その際, データ転送の最小化を考慮し同じ 8x8 ブロックを同一 PC 内で処理できるように MT_a, MT_b のスケジューリングを行ないスタティックに各 MT を割り当てる。

図 2(b) に入力データが 16x16pixel で PC 数 2 の時の実行イメージを示す。まず, 入力データを block1 から block4 の 4 つの 8x8 ブロックに分割をにする。PC1 では block1 と block2 を処理する MT_{a1} および MT_{a2} が実行され, 同様に PC2 では block3 と block4 を処理する MT_{a3} および MT_{a4} が実行される。このとき, 各 MT_a 間では依存がないため各 PC 間で並列に処理される。すべての MT_a が実行された後, MT_b 実行前に, PC2 で処理されている block3 の 1-D DC Prediction で必要とされる block2 の量子化 DC 係数を PC1 から PC2 へ転送する。データ転送後, 各 PC 間で MT_b が並列実行される。

MT_a, MT_b 内の処理においては, 近細粒度並列処理により効果的な並列処理ができることが確認されている¹²⁾。このとき近細粒度タスクは, PC 内の PE に割り当てられ処理が行なわれる。近細粒度並列処理では, 演算コストが数 10 クロックの近細粒度タスクをスタティックに PE へ割り当てるため, 各



(a) JPEG encoding block diagram



(b) Execution Flow image (2PCs)

図 2: 並列化 JPEG エンコーディング

タスクのコスト推定が性能に大きな影響を与える。しかし、エントロピー符合化処理では演算コストが入力データに依存する (DCT 係数の低周波数成分ほど演算コストが大きい傾向がある)。そのため、エントロピー符合化処理のコスト推定にはプロファイル結果を基に各タスクの演算コストを見積もった。スケジューリングについては、ヒューリスティックスケジューリングアルゴリズムである CP/DT/MISF 法, CP/ETF/MISF 法, ETF/CP 法および DT/CP 法の 4 手法のうち最良のスケジュール結果を適用し、その後スタティックに近細粒度タスクを割り当てた。

本論文では、 8×8 ブロック間の並列性を用いる粗粒度並列処理と 8×8 ブロック内の並列性を用いる近

細粒度並列処理を用いた階層的なマルチグレイン並列処理を利用する。

4 OSCAR 型シングルチップマルチプロセッサアーキテクチャ⁸⁾

ここでは、OSCAR 型シングルチップマルチプロセッサ (SCM) アーキテクチャおよびそのプロセッサコアアーキテクチャについて述べる。

4.1 メモリアーキテクチャ

OSCAR 型 SCM のネットワークおよびメモリアーキテクチャは、図 3 のように 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), および分散共有メモリ (DSM) を持つプロセッシングエレメント (PE) を相互接続網 (バス結合, クロスバ結合など) で接続し 1 チップ上に搭載したアーキテクチャである。今回の評価では、データ転送を CPU の処理とオーバーラップして行なえる DTU についてはオーバーラップデータ転送スケジューリングアルゴリズムの開発が終わっていないため利用していない。また、PE 間相互結合網は 3 本バスを利用している。

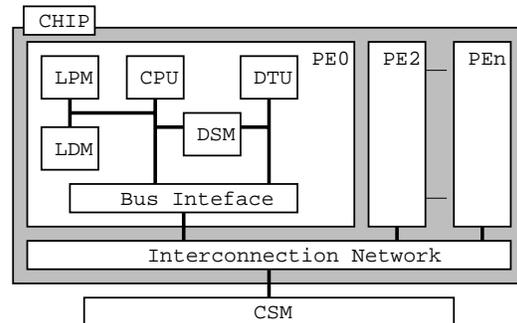


図 3: OSCAR 型 SCM アーキテクチャ

LPM は各々の CPU で実行するプログラムを格納し、1 クロックでアクセスできるものとする。同様に、LDM は PE 固有のデータを保持するために使用し、その容量は 1 チップあたりの総容量を 1M バイトとする。例えば、1 チップに 2PE 搭載する SCM では、各 PE あたり 512k バイトとして評価を行なっている。また、LDM へのアクセスレイテンシは 1 クロックとする。DSM は、自 PE と他 PE の双方から同時にアクセス可能なマルチポートメモリであ

り、近細粒度タスク間のデータ転送や、マクロデータフロー処理におけるダイナミックスケジューリング時のスケジューリング情報の通知等に使用する。DSM の容量は 1PE あたり 16k バイトとし、自 PE からのアクセスレイテンシは 1 クロック、他 PE からのアクセスレイテンシは 4 クロックとする。さらに、チップ外部には集中共有メモリ (CSM) が接続され、各 PE で共有されるデータを格納する。この CSM のアクセスレイテンシは 20 クロックとする。

OSCAR 型 SCM では、これら 4 種類のメモリに対し最適なデータ配置を行なうことにより効率の良い並列処理を行なうことができる。

4.2 プロセッサコアアーキテクチャ

各 PE が持つ CPU は、SPARC V9 規格に準拠したスーパースカラプロセッサである Sun Microsystems 社の UltraSPARC II⁽¹³⁾ のパイプライン構成をベースとし、バリア同期機構等用の特殊レジスタや特殊レジスタを操作するための命令を付加したプロセッサである。

今回の評価で用いる OSCAR 型 SCM のプロセッサコアは、整数演算ユニット (IEU) を 1 本、ロードストアユニット (LSU) を 1 本、浮動小数点ユニット (FPU) を 1 本持つシングルイシューのシンプルな構成とした。

また、スーパースカラとの比較のため UltraSPARC-II 相当のプロセッサコア (US-II) を用いたアーキテクチャについても評価を行なう。UltraSPARC-II 相当のプロセッサコアは、IEU を 2 本、LSU を 1 本、FPU を 2 本持つ 4 イシュー in-order 発行構成で、動的スケジューリング用命令バッファエントリ数は 12 である。なお、シングルイシュープロセッサコアを 4 基持った OSCAR 型 SCM と UltraSPARC-II 相当のプロセッサコア 1 基を持った OSCAR 型 SCM はトランジスタ数はほぼ同等であると考えられる。

5 性能評価

ここでは、JPEG エンコーディングプログラムにマルチグレイン並列処理を適用し OSCAR 型メモリアーキテクチャシングルチップマルチプロセッサ (SCM) 上にて評価した結果について述べる。

OSCAR 型 SCM 上での評価結果を逐次実行時間に対する速度向上率として図 4 に示す。図中横軸

表 1: プロセッサコア仕様

	Simple	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	

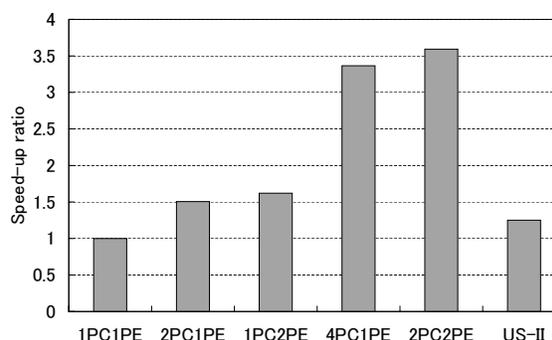


図 4: JPEG エンコーディング評価結果

の $mPCnPE$ は SCM は m 個のプロセッサクラスタ (PC) を持ち、各 PC が n 個のプロセッサエレメント (PE) を持つことを示す。よって、SCM 中の総 PE 数は $m \times n$ である。このとき、PC 間では粗粒度タスク並列処理、PC 内 PE 間では近細粒度並列処理を行なうことを前提としている。次に図中の US-II は、UltraSPARC-II 相当の 4 イシュースーパースカラプロセッサコアによる速度向上率である。

まず、UltraSPARC-II 相当のプロセッサコアを用いた場合、シングルイシュープロセッサコアの逐次実行時間に対して約 1.25 の速度向上率を得た。ほぼ同等のトランジスタ数であるシングルイシュープロセッサコアを 4 基使用したアーキテクチャでの 2PC2PE 実行時と比較するとシンプルなプロセッサコアを 4 基使用したアーキテクチャでは、UltraSPARC-II 相当プロセッサコアを 1 基使用したアーキテクチャに対して約 2.87 倍の性能向上が得られることが分かる。これは、JPEG エンコーディング処理では積和演算を多用するので乗算などのマルチサイクル命令によるパイプラインストールが発生するため UltraSPARC-II 相当のスーパースカラプロセッサ

では十分な性能が出せないのに対し、シングルイシュープロセッサコアを複数搭載した SCM アーキテクチャではプロセッサ間負荷分散により効果的な並列処理が行なえたためである。

次に、OSCAR 型 SCM アーキテクチャで粗粒度並列処理単体での性能と粗粒度並列処理と近細粒度並列処理を階層的に組み合わせたマルチグレイン並列処理での性能を比較すると、総 PE 数が 2 の時では、逐次実行時に対し 2PC1PE では約 1.50 倍、1PC2PE では約 1.62 倍の速度向上率が得られた。同様に総 PE 数が 4 の時、4PC1PE では約 3.36 倍、2PC2PE では約 3.59 倍の速度向上率が得られた。以上より使用プロセッサ数が同じ時には、階層的なマルチグレイン並列処理を使用した場合の方が高い実行効率を与えることが確かめられた。この理由は、粗粒度並列処理と近細粒度並列処理を用いた階層的なマルチグレイン並列処理を行なうことにより、各プロセッサ間の負荷バランスがより良くなりプロセッサアイドル時間が低減されるためと考えられる。

6 まとめ

本論文では、JPEG エンコーディング処理のマルチグレイン並列処理手法を提案し OSCAR 型メモリアーキテクチャシングルチップマルチプロセッサ (SCM) 上にて評価を行なった。その結果シングルイシュープロセッサコアを 4 基搭載した OSCAR 型 SCM 上にて逐次実行時間に対し約 3.59 倍の速度向上率が得られた。また、シングルイシュープロセッサコアを 4 基搭載した SCM ではほぼ同等のトランジスタ数を持つ UltraSPARC-II 相当プロセッサコアを 1 基搭載した SCM に対し約 2.87 倍の性能向上が得られることが確かめられた。以上よりシンプルなプロセッサコアを集積した OSCAR 型 SCM 上での JPEG エンコーディングのマルチグレイン並列処理は集積度向上に対しスケラブルな性能向上を可能とするということが確認できた。

今後の課題として、MPEG-2 や JPEG2000 などのマルチメディアアプリケーションに対するマルチグレイン並列化手法の検討及びより大規模な SCM 上での評価等が挙げられる。

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/NEDO 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 平田雅規氏、宮田操氏、東芝 浅野滋徳氏、安川英樹氏、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. Pentkovki and J.Keshava: Implementing Streaming SIMD Extensions on the Pentium III Processor, *IEEE MICRO*, Vol. 20, No. 4 (2000).
- [2] F. Arakawa, O. Nishi, K. Uchiyama and N. Nakagawa: SH4 RISC Multimedia Microprocessor, *IEEE MICRO*, Vol. 18, No. 2 (1998).
- [3] A. Suga and K. Matsunami: Introducing the FR500 Embedded Microprocessor, *IEEE MICRO*, Vol. 20, No. 4 (2000).
- [4] M. Edahiro, S. Matsushita, M. Yamashita and N. Nishi: A Single-Chip Multiprocessor for Smart Terminals, *IEEE MICRO*, Vol. 20, No. 4 (2000).
- [5] L. Hammond, B. Hubbert, M. Siu, M. K. Prabhu, M. Chen and K. Olukotun: The Stanford HYDRA CMP, *IEEE MICRO*, Vol. 19, No. 2 (1999).
- [6] E. Iwata and K. Olukotun: Exploiting Coarse-Grain Parallelism in the MPEG-2 Algorithm, CSL-TR-98-771, Stanford University Computer System Lab. (1998).
- [7] H. Kasahara, M. Obata and K. Ishizaka: Automatic Coarse Grain Task Parallel Processing on SMP using OpenMP, *Proc. 12th Workshop on Languages and Compilers for Parallel Computing* (2000).
- [8] 木村啓二, 加藤孝幸, 笠原博徳: 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 (2001).
- [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] E. Hamilton: *JPEG File Interchange Format Version 1.02* (1992).
- [11] Aldus Developers Desk: *TIFFTM Revision 6.0* (1992).
- [12] 小高剛, 宮下直久, 木村啓二, 笠原博徳: 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, ARC2001-140-11, 情報処理学会 (2001).
- [13] Sun Microelectronics: *UltraSPARCTM User's Manual* (1997).