社団法人 情報処理学会 研究報告
IPSJ SIG Technical Reports

2008－ARC－178（9）
2008／5／14

モバイルマルチメディア処理向けのスケーラブルなマルチプロセッサ

薄井 弘之† 野村 周央† 山根 史之† 宮本 幸昌†
カムトーンキッティクン チャイヤスィット† 田辺 淳† 内山 真郷†
宮森 高† 坪井 芳郎†
†株式会社 東芝 〒212-8520 川崎市幸区堀川町 580-1
E-mail: †hiroyuki1.usui@toshiba.co.jp

あらまし モバイルマルチメディア処理向けのマルチプロセッサを開発した．各コアは32ビット RISC プロセッサに加え，SIMD 命令を実行可能な VLIW コプロセッサを持ち，広帯域の L2 キャッシュを共有している．対称かつ粗粒度な並列化方法の採用により，コア数に対する性能のスケーラビリティを実現できる．65nm CMOS プロセスで試作したチップでは8個のコアと 512KB の L2 キャッシュを持ち，8 コアでの実行時に H.264 720p 60fps のデコードが可能である．これは1コアでの実行時と比較して 7.5 倍の性能である．
キーワード マルチプロセッサ，マルチメディア，VLIW，H.264

A Scalable Multi-core Processor for Mobile Multimedia Applications

Hiroyuki USUI† Shuou NOMURA† Fumiyuki YAMANE† Yukimasa MIYAMOTO†
Chaiyasit KUMTORNKITTIKUL† Jun TANABE† Masato UCHIYAMA†
Takashi MIYAMORI† Yoshiro TSUBOI†
† Toshiba Corporation 580-1, Horikawa-Cho, Saiwai-ku, Kawasaki, 212-8520 Japan
E-mail: †hiroyuki1.usui@toshiba.co.jp

Abstract We implemented multi-core processor for mobile multimedia applications. Each core consists of 32bit RISC processor and the SIMD VLIW coprocessor. All cores share L2 cache memory with wide bandwidth. By symmetrical coarse-grained parallelization, scalability of the performance with respect to the number of cores is realized. In the chip fabricated by 65nm process, there are 8 cores and 512KB L2 cache, and H.264 720p 60fps decoding is achieved with 8 cores. This is 7.5 times performance enhancement from single core.
Keyword Mutli-processor, Multimedia, VLIW, H.264

1. 背景

今日のモバイルマルチメディア機器には，様々な動画像・音声コーデックへの対応が求められており，それぞれをハードウェアで実装することは面積の点で問題がある．また，処理を行う画像サイズの拡大に伴い，要求性能が製品によって大きく異なるため，要求仕様に応じて柔軟に構成を変えられることが望ましい．これらの要求に応えるためには，コア数を増やすことで，性能をスケーラブルに向上させることができるマルチプロセッサが効果的である．

また，多種多様なコーデックへの対応は，開発期間の長期化を招くため，ハードウェア・ソフトウェア共に互換性についても重要となる．

そこで，我々はモバイルマルチメディア処理向けのキャッシュベースのマルチプロセッサを開発した．本プロセッサはキャッシュのサイズや，コア数を変化させることで，様々な種類・解像度のコーデック処理を容易に実現する．さらに，キャッシュベースにすることで，システム構成が変わった場合でもバイナリコードの互換性を保ちやすく，ソフトウェアの再利用性を高めることが出来る．

2. 構成

2.1. 全体構成

本プロセッサの全体構成を図 1 に示す．Media Processing Engine(MPE)はコンフィギュラブルプロセッサ MeP コア[1]に加え，SIMD 命令を実行可能な VLIW コプロセッサを持ち，面積効率と電力効率を重視した設計になっている．また，MPE はそれぞれ専用の L1 キャッシュを持つと共に，L2 キャッシュを共有している．各 MPE 内の Cache Coherency Check(CCC)

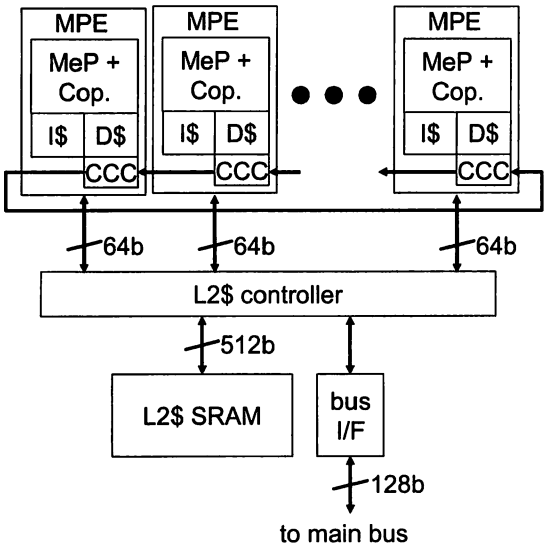

図 1　全体構成図

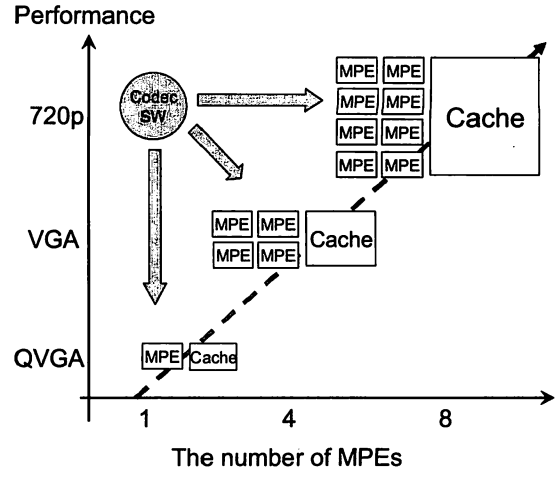

図 2　性能スケーラビリティ

Unit は，MPE-L2 キャッシュ間のバスとは別に，MPE 間でリングバスを形成している．

MPE 単体で SIMD 命令と VLIW 実行をサポートすることで，データ並列性と命令並列性を利用し MPE 単体での性能を向上する．さらに，マルチコア構成とすることでスレッドレベルの並列性を利用し，図 2 に示すように，画像サイズやフレームレートといった要求スペックに応じてコア数やキャッシュサイズを変更することで最適な構成を実現する．

さらに，キャッシュベースなマルチコアにすることで，メモリ操作が必要なスクラッチパッドメモリベースのものと比較してバイナリコードの互換性を保ちやすい．これにより，要求スペックに関わらず同じソフトウェアを実行できるようにし，開発期間の短縮を狙う．

2.2. MPE

MPE のブロック図を図 3 に示す．MeP コアは 32bit RISC プロセッサで，5 段パイプラインである．VLIW コプロセッサは画像認識システム向けの VLIW コプロセッサ[2]に動画像，音声 CODEC 向けの命令セットを追加したものである．コプロセッサ内には 64bit コプロセッサレジスタが 32 本と，256bit アキュムレータが 2 本実装されており，それを利用した SIMD 演算が実行できる．

MPE には実行モードとして，通常のモードと VLIW モードの 2 種類が存在し，通常のモードでは MeP コア命令(16 または 32bit 長)もしくは SIMD 命令(32bit 長)を含むコプロセッサ命令を 1 サイクルに 1 命令実行することが出来る．一方，VLIW モードでは MeP コア命令を 1 つと，コプロセッサ命令を 2 命令の最大 3 命令を 1 つの VLIW 命令(64bit 長)にパックし，1 サイクルで実行することが出来る．実行モードの切り替えは専用のサブルーチン分岐命令を用いて行われる．

消費電力低減を狙い，L1 命令キャッシュにはプリフェッチ時とループ処理時に命令を保持するバッファを，L1 データキャッシュには小規模のタグキャッシュを用意することで，それぞれの電力消費の主要因である L1 キャッシュへのアクセス回数を減らしている [3]．

キャッシュベースのマルチプロセッサの場合，キャッシュコヒーレンシの維持が問題となる．ハードウェアキャッシュスヌープを用いると，コア数が増えた場合に，スヌープで生じるバストランザクションにより性能が低下するため，スケーラビリティの面で問題となる．本プロセッサでは，MPE 間のキャッシュコヒーレンシをソフトウェアで維持することで，バストランザクションと消費電力の増加を抑える．ただし，デバッグ目的のために各 MPE にある Cache Coherency Check(CCC)ユニットでコヒーレンシ違反の検出を行う．

各 CCC ユニットは隣接する MPE の CCC ユニットと接続されており，全体でリングバスを形成する．L1 データキャッシュへのリフィルアクセス，L1 データキャッシュラインへの最初のライトアクセスならびに，L2 キャッシュへの直接アクセスが発生する場合に，そのアクセス情報をリングバスに流す．各 CCC ユニットは

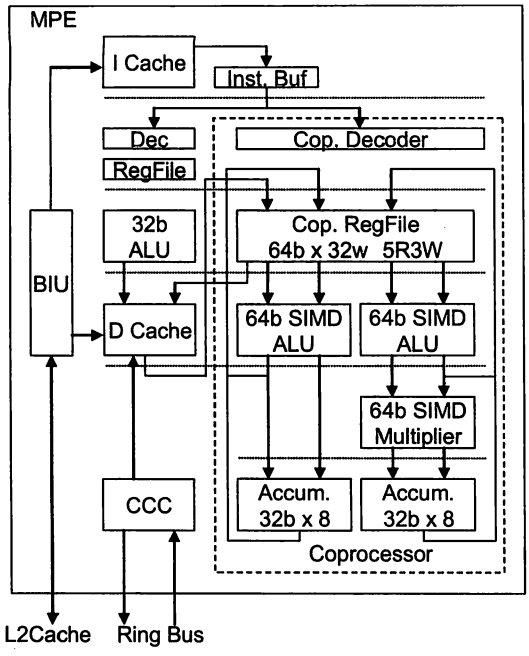

図 3 MPE の構成図

そのアクセス情報を受けると，データキャッシュをチェックし，コヒーレンシ違反を検出する．違反が検出された場合は割り込みを発生する．

このように，CCC ユニットを使ったチェック機構は通常のバスとは独立に設けられているので，このチェック機能を ON にしてもバストラフィックに影響を与えることはない．また，デバッグ時以外はチェック機能を OFF にすることで，消費電力の増加も最小限に抑えることができる．CCC ユニット自体のハードウェアコストもハードウェアキャッシュスヌープを行う場合と比較して非常に小さく済む．

2.3. L2 キャッシュ

L2 キャッシュは複数のコアで共有されるので，MPE 数を増やした場合のスケーラビリティを維持するために，競合に強いアーキテクチャをとる必要がある．ブロック図を図 4 に示す．L2 キャッシュ内ではデータ SRAM のみコントローラの 1/2 の周波数で動作し，コントローラは MPE と同じ周波数で動作する．

キャッシュユニット内には，MPE ごとに専用の 512bit のバッファがあり，バッファと L2 データ SRAM 間を 256bit 幅のデータパスで接続している．トランザクションはパイプライン処理されるため，512bit/2cycle の転送が可能である．さらに，ユニット内に 8 エントリのキューを用意することで，最大 7 つのキャッシュミス中にもキャッシュにアクセスするこ

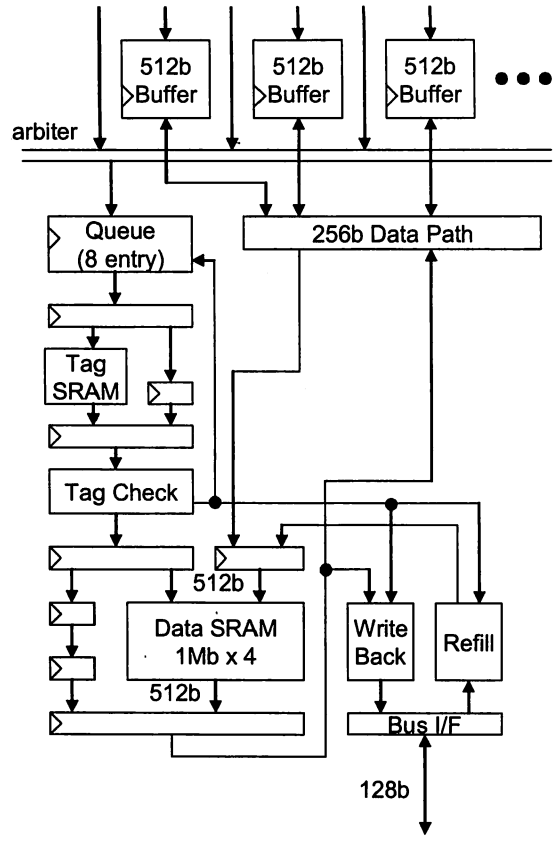

図 4 L2 キャッシュの構成図

とが可能である．このようなアーキテクチャにより，L2 キャッシュで競合が発生した場合のキャッシュアクセスレイテンシの増加を抑えている．

MPE ごとに用意された 512bit のバッファは L1 命令キャッシュ用のプリフェッチバッファとしても用いられる．命令キャッシュミスにより，L2 キャッシュから読み出したデータを 512bit のバッファに格納した時点で，次のキャッシュラインへのプリフェッチアクセスを行う．このプリフェッチアクセスは 512bit のバッファから MPE へのデータ出力と並列に行われる．プリフェッチアクセスが L2 キャッシュから読み出したデータは 512bit バッファに格納される(この時点で MPE へのデータ出力は終了している)．次の L2 キャッシュアクセスがプリフェッチしたアドレスに対する命令キャッシュミスだった場合，そのバッファ内のデータが直ちに出力され，さらに次のキャッシュラインへのプリフェッチを行う．アドレスが異なる場合や，命令キャッシュミスではない場合にはバッファ内のデータは破棄され，L2 キャッシュへアクセスする．

また，8 エントリのキューを用いたメインメモリか

ら L2 キャッシュへのプリフェッチも可能である．L2 キャッシュアクセスがキューに格納されると同時に，次のキャッシュラインに対するプリフェッチアクセスについてキューの別エントリに格納することが出来る．このプリフェッチアクセスはキュー内に実行できるプリフェッチではないアクセスが存在しない場合のみ実行され，L2 キャッシュに該当アドレスのデータがない場合，メインメモリからデータを読み出して L2 キャッシュに格納する．これを利用することで，L2 キャッシュの実行に余裕がある時に，性能向上を図る．

3. H.264 デコーダの並列処理

今回，本プロセッサ上で実装した H.264 デコーダの並列処理方法について説明する．本処理では，VLD をフレームレベルで処理し，それ以後の信号処理は図 5 のようにマクロブロックライン単位で並列に処理する．

H.264 デコードでは，左と上のマクロブロック(MB)間にデータ依存が存在する．左の MB とのデータ依存はライン単位で処理を行っているので，自動的に解消されるが，上の MB 間のデータ依存は，他の MPE で処理されている上のマクロブロックラインの処理状況をモニターし，処理が終わり次第デコード結果を取得することで解決する．上の MB の実画像データを参照するデブロッキングフィルタのような処理が H.264 デコードには含まれるため，必要とする他 MPE の処理結果のデータ量が大きくなるが，広帯域の共有 L2 キャッシュにより，他 MPE の処理結果を短時間で取得することが出来る．

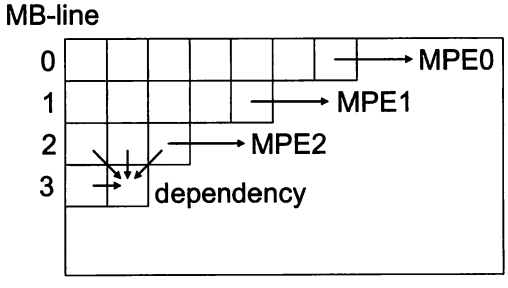

図 5　H.264 Decode の並列処理方法

また，共有 L2 キャッシュは動き補償予測時に，参照画像が置かれているメインメモリへのアクセス回数を削減する効果もある．図 6 に示すように，MPE は自分が参照した画像だけでなく，他の MPE が参照した画像についても再利用することが出来る．L2 キャッシュの 1 ラインのサイズは 256B であり，Luma のマクロブロック 1 個，もしくは Chroma のマクロブロック 2 個を 1 ラインに格納することが出来る．

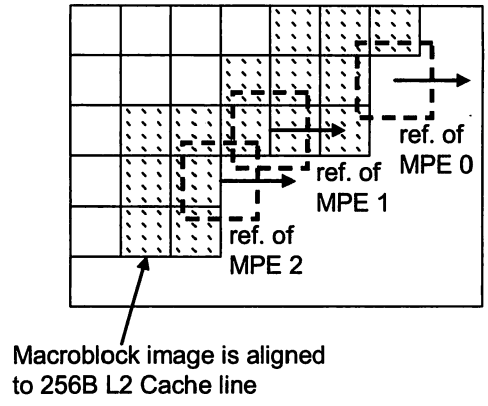

図 6　参照画像の再利用の様子

4. 評価

今回我々が試作したチップ画像を図 7 に，仕様を表 1 に示す．本チップは MPE を 8 個，L2 キャッシュを 512KB とした場合の実装である．また，このチップでは消費電力を抑えるための回路技術が実装されており，特に各 MPE が未使用のときには Power-gating を行うことで，消費電力を抑えることが出来る[4].

今回は，このチップ上で動作する MPE 数及び L2 キャッシュ容量を変えて性能を測定し，コア数とキャッシュ容量を可変にした場合のスケーラビリティについて評価した．

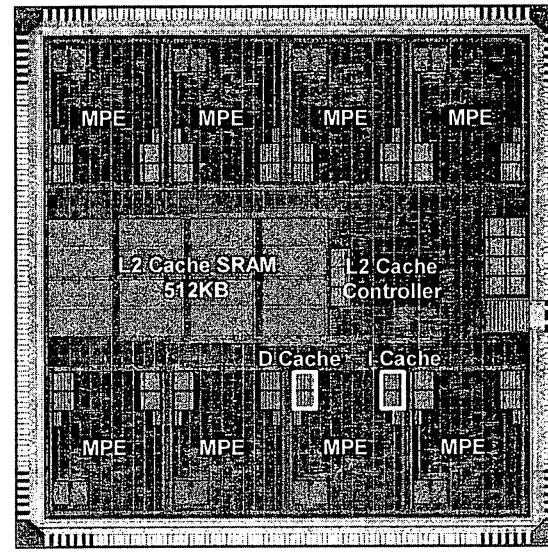

図 7 チップ画像

表 1 チップ仕様

Technology	65nm CMOS, 8-layer-metal
Die Size	5.06mm x 5.06mm
Supply Voltage	2.5V(IO), 1.2V(Core)
Clock Frequency	333MHz(MPE, L2Logic) 166MHz(L2 SRAM, Bus I/F)
L1 Cache	8KB(Instruction) / 8KB(Data) 2way, 64bytes/line
L2 Cache	512/256KB(Unified) 4-way, 256bytes/line

本チップ上で MPE 数を変えて 720p H.264 デコードを実行した時のフレームレートを図 8 に示す．実行 MPE 数を 8 個にした場合，L2 キャッシュサイズが 512KB ならば 64fps と，MPE 数が 1 個の場合の 8.5fps と比較して約 7.5 倍と性能面でスケーラビリティを達成している．ただし，MPE が 5 個以上の場合に性能面でのスケーラビリティを維持するためには，L2 キャッシュが 512KB 必要となる．

次に，720p 60Frame 分の H.264 デコードを行った場合の L2 キャッシュとメインメモリ間の転送量を図 9 に示す．キャッシュ容量が 512KB で MPE 数が 8 個の場合の転送量は 310MB/s である．キャッシュ容量が 256KB の場合はコア数が増えた場合の転送量の増加量が大きい．本評価では，メインバスアクセス時のレイテンシが 80 サイクルで，メインバスが一度に複数のトランザクションを処理できないため，L2 キャッシュミスが増加するとスケーラビリティの維持が難しくなり，L2 キャッシュ 256KB 時で MPE が 5 個以上の場合の性能低下につながっている．

720p 60fps H.264 デコード時の消費電力は 333MHz で 620mW であり，720p 30fps の場合は 166MHz で実行して 209mW である．

5. 結論

モバイルマルチメディア処理向けのスケーラブルなマルチプロセッサを開発した．各コアは L2 キャッシュを共有しており，コア数やキャッシュサイズを変更することで，様々な種類・解像度のコーデック処理をサポートする．コヒーレンシ維持のために生じるバストランザクションを抑えるために実装された CCC ユニットと，広帯域で競合に強い L2 キャッシュにより，8 コアで 720p 60fps H.264 デコードが可能である．これは 1 コアで処理を行った場合と比較して約 7.5 倍の性能となり，性能面でのスケーラビリティを達成した．

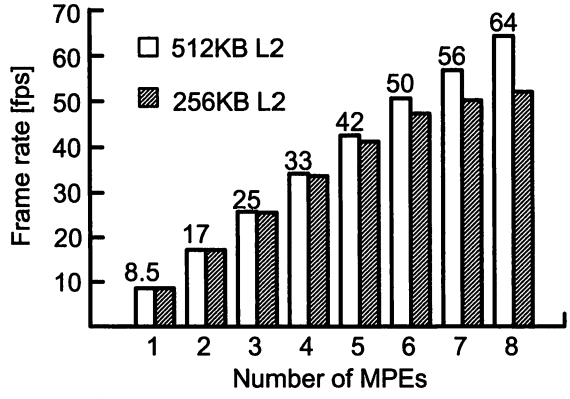

図 8 720p H.264 Decode の処理 Frame 数

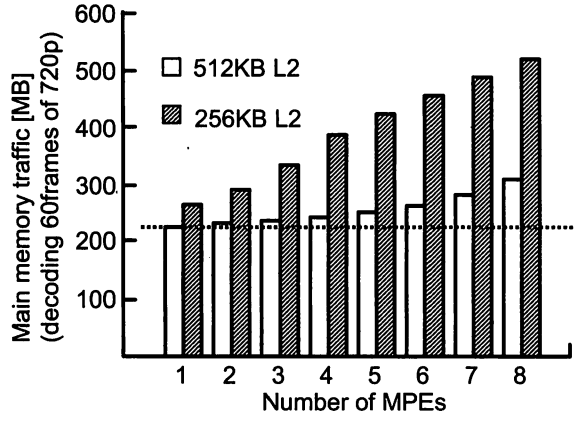

図 9 720p H.264 Decode 時のメインメモリ -L2 キャッシュ間データ転送量

文 献

[1] T. Miyamori, "A Configurable and extensible media processor, " Embedded Processor Forum, 2002.

[2] J. Tanabe, Y. Taniguchi, T. Miyamori, Y. Miyamoto, H. Takeda, M. Tarui, H. Nakayama, N. Takeda, K. Maeda, M. Matsui, "Visconti: multi-VLIW image recognition processor based on configurable processor," IEEE Custom-Integrated Circuits Conference, pp.185-188, 2003.

[3] S. Hosoda, M. Uchiyama, J. Tanabe, K. Yasufuku, T. Tamai, T. Matsumoto, T. Miyamori, M. Nakagawa, "A Low-power Mobile Multimedia Processor for Scalable Multi-core System," Cool Chips, 2008

[4] S. Nomura, F. Tachibana, T. Fujita, C. K. Teh, H. Usui, F. Yamane, Y. Miyamoto, C. Kumtornkittikul, H.Hara, T.Yamashita, et al., "A 9.7mW AAC-Decoding, 620mW H.264 720p 60fps Decoding, 8-Core Media Processor with Embedded Forward-Body-Biasing and Power-Gating Circuit in 65nm CMOS Technology," IEEE International Solid-State Circuit Conference, pp. 12-13, 2008