

# 実世界情報システムプロジェクト

## ～ VR 研究グループ 南谷・中村研究室 ～

南谷崇 中村宏

情報理工学系研究科システム情報学専攻(先端科学技術研究センター)

### 1 はじめに

21 世紀においては、高速バックボーン、一般家庭までの高速ネットワークの整備と共に、多様なマルチメディア情報端末が大衆家電として各家庭に入り込むことによって、真の意味での高度情報化社会が達成されると期待されている。そこでは、人間がリアルタイムで現実の世界とのインタラクションをとることが可能な VR 空間を構築することが重要である。その実現のためには、社会インフラであるネットワークの整備、および、高品質なマルチメディア情報端末を廉価に提供することが重要となる。ここで後者に注目すると、マルチメディア情報端末に対する性能向上の要求は、デバイスの微細化による素子の速度向上を凌いでおり、従来のデジタルシステムの VLSI アーキテクチャを踏襲しては、その要求を満たすことができなくなる、という問題が指摘されている。この問題は具体的には以下のように説明される。大容量のデータを扱うマルチメディア処理においては、システム全体の処理能力は演算処理部の能力だけではなくメモリシステム側のデータ供給能力によっても決定される。半導体技術の進展により、素子の集積度と演算処理の速度は今後も向上すると期待されているが、DRAM,SRAM 等の記憶素子の速度は大きくは向上しないと予測され、また VLSI を実装するパッケージのピン数も物理的な制約によりそれほど増加しないため、プロセッサとメモリ間の転送スループットを向上させるのは困難な状況である。しかも、情報端末である以上携帯性が重要であり、消費電力を極力抑えてこの

問題を解決しないといけない。

そこで、本研究では素子集積度の向上を活用すべく、同一 VLSI 上にプロセッサコアと大容量のメモリを搭載する VLSI アーキテクチャを提案する。大容量メモリは、従来型のキャッシュだけでなく、新たにソフトウェアから制御可能なメモリとしても用いる。この可制御メモリは、アドレス指定可能としソフトウェアによるデータ配置と入れ替えを制御可能とする点が従来のキャッシュとは本質的に異なる。また、VLSI 上大容量メモリは、キャッシュと可制御メモリの比率を動的に再構成可能なものを提案する。我々はこのアーキテクチャを SCIMA(Software Controllable Integrated Memory Architecture)と呼んでいる。可制御メモリはキャッシュと違い、動作がソフトウェアによって陽に指定されるため、性能が予測可能でリアルタイム性の維持が容易である、という特徴がある。

本年度は、SCIMA の基本アーキテクチャの検討を、性能と消費電力の 2 つの観点から行った。

## 2 SCIMA のアーキテクチャ

### 2.1 概要

図 1 に、SCIMA の構成を示す。

SCIMA では、チップ上のメモリとして、キャッシュに加えオンチップメモリを搭載する。キャッシュはハードウェア制御によりデータ配置・置き換えが行われるのに対し、オンチップメモリは、ソフトウェアでデータ配置・置き換えの指定が可

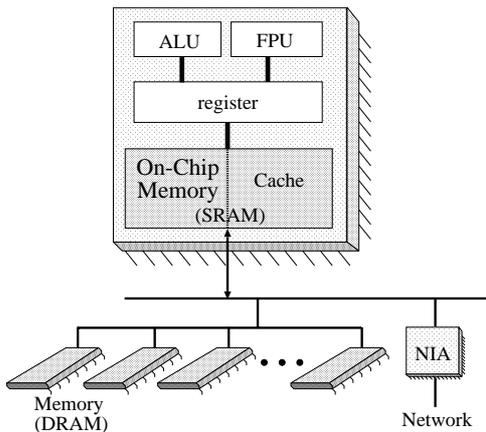


図 1: SCIMA の構成

能である .

SCIMA では、論理アドレス空間上にオンチップメモリ領域をマップする。オンチップメモリは大きな連続ブロック領域であるため、この管理を TLB ではなく専用レジスタで行ない、TLB ミスの頻発を防ぐ。導入されるレジスタは、オンチップメモリの開始アドレスを保持する ASR (On-Chip Address Start Register) とキャッシュとオンチップメモリの総容量を保持する AMR (On-Chip Address Mask Register) である。

## 2.2 拡張命令

SCIMA では、page-load/page-store と呼ぶオンチップメモリ-主記憶間の転送命令を ISA (命令セットアーキテクチャ) 上に備える。この命令により、オンチップメモリのデータ配置・置き換えをソフトウェアで行うことが可能となる。また、本命令は、ブロックストライド転送機能を備える。データ転送元の開始番地、データ転送先の開始番地、転送サイズ、ブロック幅、ストライド幅の 5 オペランドを用いたブロックストライド転送機能により、不連続なデータをオンチップメモリ上の連続領域に転送させることができるため、無駄なデータ転送を省き、チップ内の記憶領域を有効に利用可能である。

オンチップメモリ領域は *page* と呼ぶ複数のブ

ロックに分割され、この *page* を単位として管理する。*page* のサイズは 2 のべき乗である。page-load/page-store 命令で転送できる最大データサイズはこの *page* のサイズであり、*page* を跨いでの転送はできない。

## 2.3 キャッシュ・オンチップメモリ統合機構

キャッシュとオンチップメモリに割り振られる容量をアプリケーションの性質に応じて変えるべく、総容量一定のもとオンチップメモリとキャッシュの容量比を実行時に再構成できる機構を提案する。

具体的には、N-way 連想キャッシュの連続する一部の way をオンチップメモリに割り当てる方式である。例えば、32KB、4-way セットアソシアティブキャッシュの way0 ~ way3 のうち、way0 と way1 をオンチップメモリに割り当てた場合、16KB のオンチップメモリと 16KB の 2-way セットアソシアティブキャッシュに分割して使用する。

キャッシュ・オンチップメモリ統合機構を実現するために、2.1 節で述べた ASR、AMR の他に、Way Lock Register (WLR) と呼ぶ特殊レジスタをハードウェア的に用意する。WLR はキャッシュの連想度と同ビット数のレジスタであり、各 way に対応するビットがセットされている場合、その way がオンチップメモリとしてロックされていることを示す。

アドレスのビットフィールドについて、図 2 のように Tag ビット部の下位  $w$  ビット ( $2^w$  がキャッシュの連想度となる) を WAY ビット、Tag ビット部のそれ以外のビットを Decision ビットと定義する。Decision ビットはアドレスを AMR でマスクすることで得られる。

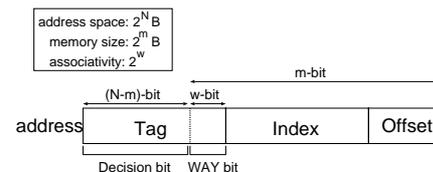


図 2: アドレスのビットフィールド

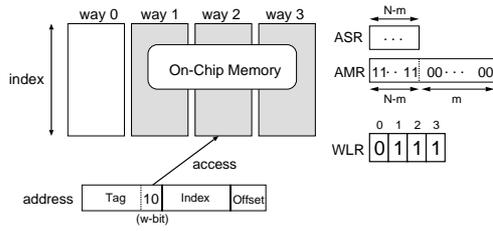


図 3: オンチップメモリの割り当てとアクセス

あるアドレスがアクセスされたとき，以下の 2 つの条件が成立した場合にオンチップメモリアクセスとなる．

- 当該アドレスの Decision ビットと ASR が一致する
- 当該アドレスの WAY ビットをデコードしたビット列において WRL の該当するビットがセットされている

この条件により，オンチップメモリアクセスであると判定された場合は，当該アドレスの WAY ビット部がアクセスすべき way を示しており，その way 内の Index ビットで決定されるカラムをアクセスする．

図 3 では，way 数が 4 のキャッシュにおいて，way1 ~ way3 がオンチップメモリとして使われ，WLR は “0111” となっている．ここで，WAY ビット部が “10” であるアドレスがアクセスされた場合，当該アドレスの Decision ビットが ASR と一致すれば，WAY ビット部が WLR で 1 がセットされている way2 を表すため，オンチップメモリアクセスと判定され，way2 がアクセスされる．

### 3 SCIMA の利点

#### 3.1 性能向上

プロセッサと主記憶の性能格差により，プロセッサは実行時間の多くを本来の計算処理ではなく，主記憶からのデータ待ち，すなわち無駄なストール時間として費やしている．この実行時間を解析するため，本論文ではプロセッサの実行

時間を CPU-busy time ( $T_b$ )，latency-stall ( $T_l$ )，throughput-stall ( $T_t$ ) の 3 つに分類する．CPU-busy time とはプロセッサが実際に計算処理を行っている時間を，latency-stall は主記憶のアクセスレーテンシがもたらすストール時間を，また throughput-stall はオフチップメモリのスループット不足に起因するストール時間をそれぞれ指す．ここで，プロセッサの総実行時間を  $T$ ，オフチップメモリスループットを無限大と仮定した場合の実行時間を  $T_\infty$ ，オフチップメモリスループットが無限大かつオフチップメモリレーテンシが 0 であると仮定した場合の実行時間を  $T_p$  とする．この  $T$ ， $T_\infty$ ， $T_p$  を用い，本論文では  $T_b$ ， $T_l$ ， $T_t$  を以下のように定義する．

$$\begin{aligned} T_b &= T_p \\ T_l &= T_\infty - T_p \\ T_t &= T - T_\infty \end{aligned}$$

表 1: 実行時間に対する影響

オンチップメモリの特徴	$T_b$	$T_l$	$T_t$
ソフトウェア制御	-	-	↓
p-load/p-store (大粒度転送)	↑	↓	-
p-load/p-store (ストライド転送)	↑	↓	↓
p-load/p-store の命令スケジューリング	-	↓	-
大容量オンチップメモリ	-	↓	↓
キャッシュにおけるレーテンシ隠蔽技術	$T_b$	$T_l$	$T_t$
大きなキャッシュラインの採用	-	↓	↑
non-blocking キャッシュ	-	↓	↑
キャッシュプリフェッチ	↑	↓	↑
大容量キャッシュ	-	↓	↓

実行時間に対する影響 表 1 は，SCIMA のオンチップメモリの特徴が，上記で分類した各実行時間に与える影響を示したものである．また，キャッシュアーキテクチャにおける代表的なレーテンシ隠蔽技術についても同様に示している．表中の「p-load/p-store (大粒度転送)」，及び「p-load/p-store (ストライド転送)」はそれぞれ 2.2 節で述べた page-load/page-store 命令の大粒度転送，及びストライド転送の機能を表す．

まず，ソフトウェア制御によりデータの再利用性を最大限に活用することで，オフチップメモリトラフィックを最小限に抑えられる．これは，

throughput-stall の短縮につながる。また，page-load/page-store 命令の大粒度転送により，メモリアクセス回数を削減することで，latency-stall を短縮することができる。さらに，ストライド転送機能は，無駄なトラフィック及びアクセス回数を削減でき，latency-stall，throughput-stall の短縮に有効である。

キャッシュにおけるレーテンシ隠蔽技術は，latency-stall を短縮させることができるが，「大容量キャッシュ」を除き throughput-stall を増大させてしまう。これは，プロセッサ・主記憶間のデータトラフィックを増大させてしまうためであり，これらレーテンシ隠蔽技術がプロセッサの総実行時間短縮に常に有効であるとは限らないことを示している。

一方，オンチップメモリを用いることで latency-stall 及び throughput-stall の両者の短縮が可能となる。今後オフチップメモリのレーテンシ増大とスループット不足がより深刻化すると予測されるため，将来的に SCIMA の有効性はさらに増すと考えられる。

### 3.2 低消費電力

従来の  $n$  ウェイ連想キャッシュではキャッシュアクセスが発生すると，アクセスされたアドレスの Index 部をデコードすることで，該当するキャッシュ内のセットを決定する。次に，該当セットのタグ，およびデータメモリの全てのウェイを並列にアクセスし，それぞれのデータを読み出す。そして，読み出されたタグとアクセスされたアドレスの Tag 部分を比較し，一致するものがあればキャッシュヒットとなり，該当ウェイのデータが選択される (data select)。このように，選択すべきデータがタグメモリの内容に依存するため，高速化のためには全てのウェイを並列に読み出す必要があり無駄な電力を消費している。

それに対し，SCIMA のキャッシュ・オンチップメモリ統合機構における，オンチップメモリのアクセス動作を図 4 に示す。従来のキャッシュに加え，SCIMA ではアクセスされたアドレスがオンチップ

メモリ領域に対するものかどうかの判定 (OCM-test: On-Chip Memory test) と，その際にどのウェイのデータをアクセスすべきかの選択 (way-select) を行なう回路が追加される。

本機構において，OCM-test 回路でキャッシュアクセスと判定された場合は，従来のキャッシュアクセスと同様の動作となる。一方，オンチップメモリアクセスと判定された場合は，way-select 回路により決定されるウェイ内のデータが選択される。この時，アクセスすべきセットは，OCM-test の判定結果によらず，アドレスの Index 部のみで決定される。

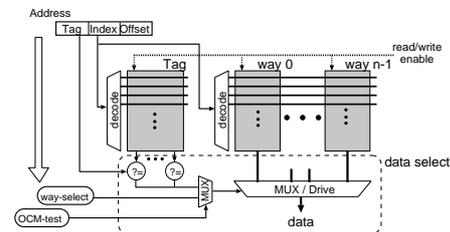


図 4: SCIMA におけるオンチップメモリアクセス

従って，キャッシュアクセス時には全てのウェイを並列にアクセスする必要があり，無駄に消費してしまう電力が多い一方で，SCIMA においてオンチップメモリアクセスの際には，アクセスすべきウェイはアドレスから一意に決まる。従って，選択すべきデータのウェイをキャッシュアクセス時より早い段階で決定すれば，オンチップメモリアクセスの際に，1つのウェイのみを選択的にアクセスすることができ，消費電力の削減につながる。

## 4 まとめ

次世代マルチメディア情報端末を指向した，同一 VLSI 上にプロセッサコアと大容量のメモリを搭載する VLSI アーキテクチャを提案し，その有効性を性能と消費電力の観点から検討した。来年度以降は，アーキテクチャの詳細化，有効性の具体的検討を引き続き実施する予定である。