

(19) 日本国特許庁 (JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2011-28343

(P2011-28343A)

(43) 公開日 平成23年2月10日 (2011.2.10)

(51) Int.Cl.	F I	テーマコード (参考)
G 0 6 F 12/08 (2006.01)	G 0 6 F 12/08 5 1 1 Z	5 B 0 0 5
G 0 6 F 12/00 (2006.01)	G 0 6 F 12/00 5 6 0 B	5 B 0 6 0
G 0 6 F 12/06 (2006.01)	G 0 6 F 12/08 5 1 3	
	G 0 6 F 12/08 5 0 3 F	
	G 0 6 F 12/08 5 1 9 Z	
審査請求 未請求 請求項の数 12 O L (全 27 頁) 最終頁に続く		

(21) 出願番号	特願2009-170679 (P2009-170679)	(71) 出願人	000005223
(22) 出願日	平成21年7月22日 (2009.7.22)		富士通株式会社
			神奈川県川崎市中原区上小田中4丁目1番1号
		(74) 代理人	100092152
			弁理士 服部 毅巖
		(72) 発明者	遠藤 久美子
			神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		(72) 発明者	石村 直也
			神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		Fターム (参考)	5B005 JJ12 KK14 MM01 NN01 NN25 5B060 CB01 CD13 KA02

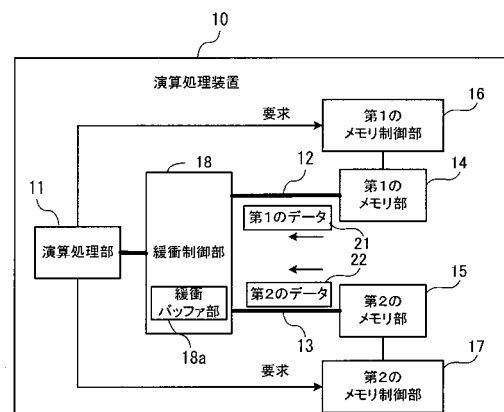
(54) 【発明の名称】 演算処理装置、およびデータ転送方法

(57) 【要約】

【課題】チップサイズの増加を抑制しながら、演算処理部へのデータ転送の並列化を可能とする。

【解決手段】演算処理部11からの要求に応じて、第1・第2のメモリ制御部16, 17により、第1・第2のデータ21, 22が第1・第2のバス12, 13を介して演算処理部11に転送される。第1・第2のバス12, 13を介して時間的に重複したデータ転送があった場合、緩衝制御部18により、一方のデータが演算処理部11に転送され、他方のデータが緩衝バッファ部18aに格納される。そして、緩衝制御部18により、一方のデータの転送終了後、他方のデータが緩衝バッファ部18aから演算処理部11に転送される。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

演算を行う演算処理部と、
前記演算処理部からの第 1 の要求を入力する第 1 のメモリ制御部と、
前記演算処理部からの第 2 の要求を入力する第 2 のメモリ制御部と、
前記第 1 のメモリ制御部に接続され、前記第 1 のメモリ制御部の制御により、前記第 1 の要求に応じた第 1 のデータを、第 1 のバスを介して出力する第 1 のメモリ部と、
前記第 2 のメモリ制御部に接続され、前記第 2 のメモリ制御部の制御により、前記第 2 の要求に応じた第 2 のデータを、第 2 のバスを介して出力する第 2 のメモリ部と、
緩衝バッファ部を有し、前記第 1 のバスを介して前記第 1 のメモリ部に接続されると共に、前記第 2 のバスを介して前記第 2 のメモリ部に接続され、前記第 1 のバスを介した前記第 1 のデータの出力と前記第 2 のバスを介した前記第 2 のデータの出力が同一期間に行われる場合、前記第 1 のデータを前記演算処理部に転送し、前記第 2 のデータを前記緩衝バッファ部に格納し、前記第 1 のデータの転送の終了後、前記第 2 のデータを前記緩衝バッファ部から前記演算処理部に転送する緩衝制御部と、
を有することを特徴とする演算処理装置。

10

【請求項 2】

前記演算処理装置において、
前記緩衝制御部は、前記第 1 および第 2 のメモリ部から同時にデータが出力された場合、前記第 1 および第 2 のメモリ部のうち、データ転送時間が大きい一のメモリ部からのデータを前記第 1 のデータとして前記演算処理部に転送し、他のメモリ部からのデータを前記第 2 のデータとして前記緩衝バッファ部に格納することを特徴とする請求項 1 記載の演算処理装置。

20

【請求項 3】

前記演算処理装置において、
前記緩衝制御部は、前記第 1 および第 2 のバスのうちのバスを介して出力されたデータを前記演算処理部に転送中に、他のバスからデータが出力された場合、前記他のバスを介して出力されたデータを前記バッファに格納することを特徴とする請求項 1 記載の演算処理装置。

【請求項 4】

前記演算処理装置は、前記演算処理部を複数有し、
前記緩衝制御部には、複数の前記演算処理部が接続され、
前記緩衝制御部は、前記第 1 のメモリ部が前記第 1 のバスを介して出力した第 1 のデータと前記第 2 のメモリ部が前記第 2 のバスを介して出力した第 2 のデータを、複数の前記演算処理部に振り分けることを特徴とする請求項 1 記載の演算処理装置。

30

【請求項 5】

前記演算処理装置において、
前記第 1 のメモリ制御部は、前記演算処理部からの前記第 1 の要求の処理開始から所定期間、前記演算処理部からの後続の要求の処理を停止することを特徴とする請求項 1 記載の演算処理装置。

40

【請求項 6】

前記演算処理装置において、
前記第 1 のメモリ制御部は、少なくとも前記緩衝制御部から前記演算処理部への前記第 1 のデータの転送に要する時間を、前記所定期間とすることを特徴とする請求項 5 記載の演算処理装置。

【請求項 7】

演算を行う演算処理部を有する演算処理装置のデータ転送方法において、
前記演算処理装置が有する第 1 のメモリ制御部が、前記演算処理部からの第 1 の要求を入力するステップと、
前記演算処理装置が有する第 2 のメモリ制御部が、前記演算処理部からの第 2 の要求を

50

入力するステップと、

前記第 1 のメモリ制御部に接続された第 1 のメモリ部が、前記第 1 のメモリ制御部の制御により、前記第 1 の要求に応じた第 1 のデータを、第 1 のバスを介して出力するステップと、

前記第 2 のメモリ制御部に接続された第 2 のメモリ部が、前記第 2 のメモリ制御部の制御により、前記第 2 の要求に応じた第 2 のデータを、第 2 のバスを介して出力するステップと、

前記演算処理装置が有する緩衝制御部が、前記第 1 のバスを介して前記第 1 のメモリ部に接続されると共に、前記第 2 のバスを介して前記第 2 のメモリ部に接続され、前記第 1 のバスを介した前記第 1 のデータの出力と前記第 2 のバスを介した前記第 2 のデータの出力が同一期間に行われる場合、前記第 1 のデータを前記演算処理部に転送し、前記第 2 のデータを前記緩衝バッファ部に格納し、前記第 1 のデータの転送の終了後、前記第 2 のデータを緩衝バッファ部から前記演算処理部に転送するステップと、

を有することを特徴とするデータ転送方法。

【請求項 8】

前記データ転送方法において、

前記緩衝制御部は、前記第 1 および第 2 のメモリ部から同時にデータが出力された場合、前記第 1 および第 2 のメモリ部のうち、データ転送時間が大きい一のメモリ部からのデータを前記第 1 のデータとして前記演算処理部に転送し、他のメモリ部からのデータを前記第 2 のデータとして前記緩衝バッファ部に格納することを特徴とする請求項 7 記載のデータ転送方法。

【請求項 9】

前記データ転送方法において、

前記緩衝制御部は、前記第 1 および第 2 のバスのうちのバスを介して出力されたデータを前記演算処理部に転送中に、他のバスからデータが出力された場合、前記他のバスを介して出力されたデータを前記バッファに格納することを特徴とする請求項 7 記載のデータ転送方法。

【請求項 10】

前記データ転送方法において、

前記演算処理装置は、前記演算処理部を複数有し、

前記緩衝制御部には、複数の前記演算処理部が接続され、

前記緩衝制御部は、前記第 1 のメモリ部が前記第 1 のバスを介して出力した第 1 のデータと前記第 2 のメモリ部が前記第 2 のバスを介して出力した第 2 のデータを、複数の前記演算処理部に振り分けることを特徴とする請求項 7 記載のデータ転送方法。

【請求項 11】

前記データ転送方法において、

前記第 1 のメモリ制御部は、前記演算処理部からの前記第 1 の要求の処理開始から所定期間、前記演算処理部からの後続の要求の処理を停止することを特徴とする請求項 7 記載のデータ転送方法。

【請求項 12】

前記データ転送方法において、

前記第 1 のメモリ制御部は、前記緩衝制御部から前記演算処理部への前記第 1 のデータの転送に要する時間を、前記所定期間とすることを特徴とする請求項 11 記載のデータ転送方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、演算処理装置、およびデータ転送方法に関する。

【背景技術】

【0002】

10

20

30

40

50

コンピュータに用いられるCPU (Central Processing Unit) やDSP (Digital Signal Processor) などの演算処理装置は、動作周波数を上げることで演算性能を向上させることができる。ところが近年、消費電力の増大などが原因となり、動作周波数を上げることによる演算処理装置の性能向上が限界に達しつつある。そこで、演算処理装置のマルチコア化によって、さらなる性能向上を目指す傾向にある。

【0003】

演算処理装置がマルチコア化されると、1つのLSI (Large Scale Integrated circuit) チップに複数の演算処理部としてのプロセッサコア (以下、単に「コア」と呼ぶ) が搭載される。マルチコア化された演算処理装置では、処理のスループットを確保するために、キャッシュメモリや主記憶装置を複数のバンクに分け、コアからキャッシュメモリや主記憶装置へのアクセスをバンク単位で行うなどの工夫がなされる。

10

【先行技術文献】

【特許文献】

【0004】

【特許文献1】特表2006-522385号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

ところで、演算処理装置内の各コアからバンクに分けられたキャッシュメモリにアクセスする場合、キャッシュメモリ制御回路によってアクセスが制御される。キャッシュメモリ制御回路が1つだけの場合、複数のコアから同時にデータ取得の要求が発生したとき、各要求をキャッシュメモリ制御回路が順番に処理することとなる。すると一部のコアにおいてキャッシュメモリからのデータ読み出しの待ち時間が長期化し、データアクセス性能の劣化を招いてしまう。

20

【0006】

そこで、データのアクセス性能向上のため、キャッシュメモリを複数のバンクに分けると共に、それぞれのバンクに対応してキャッシュメモリ制御回路を複数設け、キャッシュメモリへのアクセスをバンク単位で制御することが考えられる。キャッシュメモリへのアクセスをバンク単位で行うことにより、複数のコアからデータ取得の要求が同時に発生しても、要求先が異なるバンクであれば、個々のキャッシュメモリ制御回路がそれぞれの要求を同時に並列処理することができる。その結果、データ取得の効率を向上させることができる。

30

【0007】

しかし、コアからキャッシュメモリへのアクセスの並列化は、演算処理装置を実装したLSIのチップサイズ増大の要因になる。すなわちキャッシュメモリアクセスを並列で実行可能とするには、キャッシュメモリからコアへのデータバスを複数設けて、さらにコアの受け口には、複数のデータバス用の配線とそれを受けるレジスタやセレクタを設けることとなる。これにより、演算処理装置を実装したLSIのチップサイズが増大する。チップサイズが肥大化すると、1枚のシリコンウエハから切り出せるチップ数が減少し、演算処理装置の製造コストも上昇してしまう。

40

【0008】

しかも、1つのチップ内の演算処理部であるコア数は増加傾向にある。そのため、個々のコアの占有面積の増加がチップサイズの肥大化に与える影響は、今後より顕著になるものと考えられる。

【0009】

本発明はこのような点に鑑みてなされたものであり、チップサイズの増加を抑制しながら、演算処理部へのデータ転送の並列化を可能とした演算処理装置、およびデータ転送方法を提供することを目的とする。

【課題を解決するための手段】

【0010】

50

上記課題を解決するために、以下のような演算処理装置が提供される。

演算処理装置は、演算処理部、第1のメモリ制御部、第2のメモリ制御部、第1のメモリ部、第2のメモリ部、および緩衝制御部を有する。演算処理部は、演算を行う。第1のメモリ制御部は、演算処理部からの第1の要求を入力する。第2のメモリ制御部は、演算処理部からの第2の要求を入力する。第1のメモリ部は、第1のメモリ制御部に接続され、第1のメモリ制御部の制御により、第1の要求に応じた第1のデータを、第1のバスを介して出力する。第2のメモリ部は、第2のメモリ制御部に接続され、第2のメモリ制御部の制御により、第2の要求に応じた第2のデータを、第2のバスを介して出力する。緩衝制御部は、緩衝バッファ部を有し、第1のバスを介して第1のメモリ部に接続されると共に、第2のバスを介して第2のメモリ部に接続され、第1のバスを介した第1のデータの出力と第2のバスを介した第2のデータの出力が同一期間に行われる場合、第1のデータを演算処理部に転送し、第2のデータを緩衝バッファ部に格納し、第1のデータの転送の終了後、第2のデータを緩衝バッファ部から演算処理部に転送する。

10

【0011】

また、上記課題を解決するために、以下のようなデータ転送方法が提供される。

このデータ転送方法では、演算を行う演算処理部を有する演算処理装置が有する第1のメモリ制御部が、演算処理部からの第1の要求を入力する。演算処理装置が有する第2のメモリ制御部が、演算処理部からの第2の要求を入力する。第1のメモリ制御部に接続された第1のメモリ部が、第1のメモリ制御部の制御により、第1の要求に応じた第1のデータを、第1のバスを介して出力する。第2のメモリ制御部に接続された第2のメモリ部が、第2のメモリ制御部の制御により、第2の要求に応じた第2のデータを、第2のバスを介して出力する。演算処理装置が有する緩衝制御部が、第1のバスを介して第1のメモリ部に接続されると共に、第2のバスを介して第2のメモリ部に接続され、第1のバスを介した第1のデータの出力と第2のバスを介した第2のデータの出力が同一期間に行われる場合、第1のデータを演算処理部に転送し、第2のデータを緩衝バッファ部に格納し、第1のデータの転送の終了後、第2のデータを緩衝バッファ部から演算処理部に転送する。

20

【発明の効果】

【0012】

上記演算処理装置およびデータ転送方法によれば、チップサイズの増加を抑制しながら、演算処理部へのデータ転送の並列化が可能となる。

30

【図面の簡単な説明】

【0013】

【図1】第1の実施の形態に係る演算処理装置の構成を示す図である。

【図2】第2の実施の形態に係る演算処理装置の回路構成例を示す図である。

【図3】キャッシュメモリ制御回路の内部構成を示す図である。

【図4】緩衝制御回路の内部構成を示す図である。

【図5】データ入出力回路の内部構成を示す図である。

【図6】待ち合わせ制御回路のデータ転送指示手順を示すフローチャートである。

【図7】「BANK0&1」と「BANK2&3」のデータ入力が行われた場合のデータ入出力を示すシーケンス図である。

40

【図8】「BANK0&1」のデータ入力後に続けて「BANK2&3」のデータ入力が行われた場合のデータ入出力を示すシーケンス図である。

【図9】「BANK0&1」と「BANK2&3」とから「CORE0」と「CORE1」へのデータ転送が連続して行われた場合のデータ入出力を示すシーケンス図である。

【図10】複数のキャッシュメモリ制御回路から同一コアへデータを転送する場合の処理を示すシーケンス図である。

【図11】マルチコアのLSIチップを実装したシステムモジュールの例を示す図である。

。

【発明を実施するための形態】

50

【 0 0 1 4 】

以下、本実施の形態について図面を参照して説明する。

[第 1 の実施の形態]

図 1 は、第 1 の実施の形態に係る演算処理装置の構成を示す図である。演算処理装置 10 は、演算処理部 11、第 1 のバス 12、第 2 のバス 13、第 1 のメモリ部 14、第 2 のメモリ部 15、第 1 のメモリ制御部 16、第 2 のメモリ制御部 17、および緩衝制御部 18 を有している。

【 0 0 1 5 】

演算処理部 11 は、演算器等を有するプロセッサコアであり、データ処理を行う。演算処理部 11 は、データ処理に必要なデータを、第 1 のメモリ制御部 16 および第 2 のメモリ制御部 17 に要求する。

10

【 0 0 1 6 】

第 1 のバス 12 および第 2 のバス 13 は、演算処理部 11 へのデータ転送に用いられる。

第 1 のメモリ部 14 は、第 1 のバス 12 に接続されている。第 2 のメモリ部 15 は、第 2 のバス 13 に接続されている。なお、第 1 のメモリ部 14 と第 1 のバス 12 との接続関係、および第 2 のメモリ部 15 と第 2 のバス 13 との接続関係は、それぞれ他の回路を経由した間接的な接続関係であってもよい。

【 0 0 1 7 】

第 1 のメモリ制御部 16 は、第 1 のバス 12 に対応付けて設けられ、演算処理部 11 から要求された第 1 のメモリ部 14 内の第 1 のデータ 21 を、対応する第 1 のバス 12 を介して演算処理部 11 に転送する。第 2 のメモリ制御部 17 は、第 2 のバス 13 に対応付けて設けられ、演算処理部 11 から要求された第 2 のメモリ部 15 内の第 2 のデータ 22 を、対応する第 2 のバス 13 を介して演算処理部 11 に転送する。

20

【 0 0 1 8 】

緩衝制御部 18 は、第 1 のバス 12 および第 2 のバス 13 それぞれと演算処理部 11 との間に接続されている。緩衝制御部 18 は、第 1 のバス 12 および第 2 のバス 13 を介して時間的に重複したデータ転送があった場合、すなわち同一期間内に複数のデータ転送があった場合、一方のデータを演算処理部 11 に転送し、他方のデータを緩衝バッファ部 18a に格納する。例えば第 1 のデータ 21 の方が先に緩衝制御部 18 に入力された場合、第 1 のデータ 21 が演算処理部 11 に転送され、第 2 のデータ 22 が緩衝バッファ部 18a に格納される。そして、緩衝制御部 18 は、一方のデータの転送終了後、他方のデータを緩衝バッファ部 18a から演算処理部 11 に転送する。

30

【 0 0 1 9 】

このような演算処理装置 10 によれば、演算処理部 11 からの要求に応じて、メモリ制御部 16 により、第 1 のデータ 21 が第 1 のバス 12 を介して演算処理部 11 に転送される。また演算処理部 11 からの要求に応じて、メモリ制御部 17 により、第 2 のデータ 22 が第 2 のバス 13 を介して演算処理部 11 に転送される。第 1 のバス 12 および第 2 のバス 13 を介して時間的に重複したデータ転送があった場合、緩衝制御部 18 により、一方のデータが演算処理部 11 に転送され、他方のデータが緩衝バッファ部 18a に格納される。そして、緩衝制御部 18 により、一方のデータの転送終了後、他方のデータが緩衝バッファ部 18a から演算処理部 11 に転送される。

40

【 0 0 2 0 】

これにより、演算処理装置 10 のチップサイズの増加を抑制しながら、演算処理部 11 からのメモリアクセスの並列化が可能となる。すなわち、演算処理部 11 へデータを転送するバスを複数設けているにも拘わらず、演算処理部 11 におけるバスからのデータ入力が 1 系統で済む。そのため、演算処理部 11 の専有面積が大きくなることがない。しかも、演算処理部 11 と第 1 のバス 12 および第 2 のバス 13 との間に設けた緩衝制御部 18 は、演算処理部 11 の内部で緩衝制御を行う場合に比べて、小さな回路で済む。そのため、演算処理装置 10 全体のチップサイズの増加も最小限に抑えることができる。

50

【 0 0 2 1 】

[第 2 の実施の形態]

第 2 の実施の形態は、マルチコアの L S I において、キャッシュメモリを介したデータアクセスの並列化を、L S I のチップサイズの増加を最小限に抑制して実現するものである。

【 0 0 2 2 】

図 2 は、第 2 の実施の形態に係る演算処理装置の回路構成例を示す図である。L S I 1 0 0 は、複数のコア 1 1 1 ~ 1 1 8 を有している。L S I 1 0 0 は、例えばマルチコア構成の C P U や D S P である。図 2 の例では、コア 1 1 1 ~ 1 1 8 の数は総計 8 個である。各コア 1 1 1 ~ 1 1 8 には「CORE0」~「CORE7」のコア識別番号が付与されている。コア 1 1 1 ~ 1 1 8 は、個別にデータ処理を実行することができる。

10

【 0 0 2 3 】

L S I 1 0 0 のキャッシュメモリは、複数のデータメモリ 1 2 1 ~ 1 2 4 を有している。データメモリ 1 2 1 ~ 1 2 4 それぞれが、キャッシュメモリの 1 つのバンクである。すなわちキャッシュメモリが、データメモリ 1 2 1 ~ 1 2 4 により 4 バンクに分けられている。各バンクには「BANK0」~「BANK3」のバンク番号が付与されている。コア 1 1 1 ~ 1 1 8 から要求されるメモリアドレス（主記憶装置内におけるデータのアドレス）では、所定の 2 ビットの値がバンク番号を示している。所定の 2 ビットを参照することで、どのバンクのデータに対する要求なのかが判別できる。第 2 の実施の形態では、要求されるメモリアドレスの下位 2 ビットによりバンクの切り分けを行う。

20

【 0 0 2 4 】

各データメモリ 1 2 1 ~ 1 2 4 は、コア 1 1 1 ~ 1 1 8 で使用される命令やデータを一次的に記憶する記憶領域である。例えばデータメモリ 1 2 1 ~ 1 2 4 には、1 ライン当たり 1 2 8 バイトのデータが格納される。データメモリ 1 2 1 ~ 1 2 4 の 1 ライン当たりのデータ量は、キャッシュブロックサイズと呼ばれる。バス 1 0 1 ~ 1 0 4 を介したデータ転送は、1 キャッシュブロックサイズ単位で行われる。

【 0 0 2 5 】

また、データメモリ 1 2 1 ~ 1 2 4 としては、主記憶装置よりも高速にリード/ライトが可能な記憶装置が使用される。例えばデータメモリ 1 2 1 ~ 1 2 4 として、スタティック R A M (Static Random Access Memory) が使用される。

30

【 0 0 2 6 】

データメモリ 1 2 1 ~ 1 2 4 は、同じバンク識別番号が付与されたバンクごとに設けられた外部の主記憶装置（図示せず）に接続されている。L S I 1 0 0 から主記憶装置へのアクセスは、各主記憶装置に対応付けられた主記憶制御回路 1 3 1 ~ 1 3 4 を介して行われる。

【 0 0 2 7 】

また、L S I 1 0 0 は、4 つのバンクに分けられたデータメモリ 1 2 1 ~ 1 2 4 へのデータ入出力を制御するキャッシュメモリ制御回路 1 4 0 , 1 5 0 を有している。2 つのキャッシュメモリ制御回路 1 4 0 , 1 5 0 は、それぞれ 2 つずつのデータメモリを制御する。図 2 の例では、キャッシュメモリ制御回路 1 4 0 がデータメモリ 1 2 1 , 1 2 2 を制御し、キャッシュメモリ制御回路 1 5 0 がデータメモリ 1 2 3 , 1 2 4 を制御する。

40

【 0 0 2 8 】

ここで、データメモリ 1 2 1 ~ 1 2 4 とコア 1 1 1 ~ 1 1 8 との間は、データ転送用のバス 1 0 1 ~ 1 0 4 で接続されている。バス 1 0 1 は、データメモリ 1 2 1 およびデータメモリ 1 2 2 からコア 1 1 1 , 1 1 2 , 1 1 5 , 1 1 6 へのデータの転送に使用される。バス 1 0 2 は、データメモリ 1 2 3 およびデータメモリ 1 2 4 からコア 1 1 1 , 1 1 2 , 1 1 5 , 1 1 6 へのデータの転送に使用される。バス 1 0 3 は、データメモリ 1 2 1 およびデータメモリ 1 2 2 からコア 1 1 3 , 1 1 4 , 1 1 7 , 1 1 8 へのデータの転送に使用される。バス 1 0 4 は、データメモリ 1 2 3 およびデータメモリ 1 2 4 からコア 1 1 3 , 1 1 4 , 1 1 7 , 1 1 8 へのデータの転送に使用される。

50

【 0 0 2 9 】

バス 1 0 1 ~ 1 0 4 のデータ幅は、例えば 3 2 バイトである。この場合、コアからの要求に対する応答で転送されるデータ量が 1 2 8 バイトであれば、データメモリ 1 2 1 ~ 1 2 4 からコア 1 1 1 ~ 1 1 8 へ、バスの動作クロックで 4 サイクルかけてデータ転送が行われる。

【 0 0 3 0 】

キャッシュメモリ制御回路 1 4 0 は、キャッシュメモリの「BANK0」と「BANK1」とに相当するデータメモリ 1 2 1 , 1 2 2 に接続されている。キャッシュメモリ制御回路 1 4 0 は、コア 1 1 1 ~ 1 1 8 からのデータ要求を受け取ると、要求されたデータがデータメモリ 1 2 1 , 1 2 2 内に存在するか否かを判断する。要求されたデータがデータメモリ 1 2 1 , 1 2 2 にあれば、キャッシュメモリ制御回路 1 4 0 はデータメモリ 1 2 1 , 1 2 2 から要求されたデータを読み出し、バス 1 0 1 , 1 0 3 を介してデータ要求を出力したコアに転送する。また、要求されたデータがデータメモリ 1 2 1 , 1 2 2 になければ、キャッシュメモリ制御回路 1 4 0 は主記憶制御回路 1 3 1 , 1 3 2 を介して、要求されたデータを主記憶装置から読み込む。キャッシュメモリ制御回路 1 4 0 は、主記憶装置から読み込んだデータを、キャッシュデータとしてデータメモリに格納すると共に、バス 1 0 1 , 1 0 3 を介してデータを要求したコアに転送する。

【 0 0 3 1 】

キャッシュメモリ制御回路 1 5 0 は、キャッシュメモリの「BANK2」と「BANK3」とに相当するデータメモリ 1 2 3 , 1 2 4 に接続されている。キャッシュメモリ制御回路 1 5 0 は、コア 1 1 1 ~ 1 1 8 からのデータ要求を受け取ると、要求されたデータがデータメモリ 1 2 3 , 1 2 4 に存在するか否かを判断する。要求されたデータがデータメモリ 1 2 3 , 1 2 4 にあれば、キャッシュメモリ制御回路 1 5 0 はデータメモリ 1 2 3 , 1 2 4 から要求されたデータを読み出し、バス 1 0 2 , 1 0 4 を介してデータ要求を出力したコアに転送する。また、要求されたデータがデータメモリ 1 2 3 , 1 2 4 になければ、キャッシュメモリ制御回路 1 5 0 は主記憶制御回路 1 3 3 , 1 3 4 を介して要求されたデータを主記憶装置から読み込む。キャッシュメモリ制御回路 1 5 0 は、主記憶装置から読み込んだデータを、キャッシュデータとしてデータメモリに格納すると共に、バス 1 0 2 , 1 0 4 を介してデータ要求を出力したコアに転送する。

【 0 0 3 2 】

キャッシュメモリ制御回路 1 4 0 は、パイプラインにより 2 バンクのデータメモリ 1 2 1 , 1 2 2 と主記憶制御回路 1 3 1 , 1 3 2 とを制御する。同様にキャッシュメモリ制御回路 1 5 0 は、パイプラインにより 2 バンクのデータメモリ 1 2 3 , 1 2 4 と主記憶制御回路 1 3 3 , 1 3 4 とを制御する。このバンクは、要求するメモリアドレスのある 2 ビットを用いて分けられ、コアからの要求がアドレスに応じて分散して処理されることを示す。コアからの要求がキャッシュミスした場合、主記憶制御回路 1 3 1 ~ 1 3 4 からのデータは同じバンクのデータメモリに登録される。

【 0 0 3 3 】

ここで、2 つのキャッシュメモリ制御回路 1 4 0 , 1 5 0 は、相互に通信せず全く独立に動作する。これは相互の依存関係によるハングなどを誘発させないためである。このような独立した構成は、システムコントローラ L S I をバンクごとに別 L S I にすると似ている。これに伴い、キャッシュメモリ (データメモリ 1 2 1 ~ 1 2 4) からコア 1 1 1 ~ 1 1 8 へのデータバスは、「BANK0」および「BANK1」のデータを転送するバス 1 0 1 , 1 0 3 と、「BANK2」および「BANK3」のデータを転送するバス 1 0 2 , 1 0 4 が設けられる。これらのバス 1 0 1 ~ 1 0 4 は、互いに独立に配線される。

【 0 0 3 4 】

緩衝制御回路 1 6 0 , 1 7 0 , 1 8 0 , 1 9 0 は、コア 1 1 1 ~ 1 1 8 へ転送するデータの入力タイミングを制御する回路である。緩衝制御回路 1 6 0 , 1 7 0 , 1 8 0 , 1 9 0 は、それぞれ 2 つのコアに接続されており、各コアへのデータの振り分けを行う。具体的には、緩衝制御回路 1 6 0 はコア 1 1 1 , 1 1 2 に接続されている。緩衝制御回路 1 7

10

20

30

40

50

0はコア113, 114に接続されている。緩衝制御回路180はコア115, 116に接続されている。緩衝制御回路190はコア117, 118に接続されている。

【0035】

各緩衝制御回路160, 170, 180, 190は、緩衝バッファを有している。緩衝制御回路160, 170, 180, 190は、「BANK0&1」と「BANK2&3」とのデータ転送が競合した場合は、緩衝バッファを用いて待ち合わせ制御を行う。すなわち、緩衝制御回路160, 170, 180, 190は、2つのバスを介して入力されたデータが競合すると、一方のデータを緩衝バッファに格納する。入力されたデータが競合する場合とは、例えばバス101, 102を介して2つのキャッシュメモリ制御回路140, 150からほぼ同時に同一のコア111宛のデータが入力された場合である。このような場合、コア111に接続された緩衝制御回路160は、一方のデータは緩衝バッファを介さずにコアに転送し、他方のデータは緩衝バッファに格納する。そして緩衝制御回路160は、コアに先に転送したデータの転送終了後に、緩衝バッファに格納したデータをコアに転送する。

【0036】

このようなマルチコアのLSI100では、キャッシュメモリ制御回路140, 150によって各コア111~118へのデータ転送が制御される。以下、キャッシュメモリ制御回路140, 150の内部構造について説明する。

【0037】

図3は、キャッシュメモリ制御回路の内部構造を示す図である。図3には、コア111から要求されたデータをキャッシュメモリ制御回路140, 150がコア111に送信する回路構成が示されている。

【0038】

キャッシュメモリ制御回路140は、ムーブインポート(MI PORT)141、パイプライン制御回路142、タグメモリ(TAG)143、ムーブインバッファ(MIB)144, 145、およびムーブインデータキュー(MIDQ)146, 147を有している。ムーブインポート141は、制御バスでコア111に接続されている。ムーブインポート141は、コア111からの要求を受け付ける回路である。ムーブインポート141からの出力はパイプライン制御回路142に入力されている。パイプライン制御回路142は、リクエストの受信や選択、タグ検索などのパイプライン制御を行う回路である。パイプライン制御回路142には、タグメモリ143が接続されている。タグメモリ143は、データメモリ121, 122に格納されたデータのタグを記憶する回路である。

【0039】

タグメモリ143からの出力は、データメモリ121, 122へのアドレス入力、あるいはムーブインバッファ144, 145への入力となる。ムーブインバッファ144, 145は、タグメモリ143を検索した結果、要求されたデータがデータメモリ121, 122に存在しないときに、要求されたデータのメモリアドレスなどを記憶する。またムーブインバッファ144, 145は、主記憶装置からのデータをキャッシュメモリに登録するまでの制御を行う。ムーブインバッファ144, 145に格納されたメモリアドレスは、主記憶装置から読み込むデータのメモリアドレスとして、主記憶制御回路131, 132に対して出力される。

【0040】

また主記憶制御回路131, 132から出力されたデータは、ムーブインバッファ144, 145とムーブインデータキュー146, 147に入力される。ムーブインデータキュー146, 147は、主記憶装置から読み込んだデータを一時的に格納する回路である。なお、ムーブインバッファ144, 145はパイプライン制御回路142に接続されており、主記憶制御回路131, 132からデータが入力されると、格納していたメモリアドレスをパイプライン制御回路142に出力する。ムーブインデータキュー146, 147は、主記憶制御回路131, 132から入力されたデータを格納すると、パイプライン制御回路142からの指示により、そのデータをデータメモリ121, 122へ書き込むと共にバス101とに対して出力する。

10

20

30

40

50

【 0 0 4 1 】

キャッシュメモリ制御回路 1 5 0 は、ムーブインポート (M I P O R T) 1 5 1、パイプライン制御回路 1 5 2、タグメモリ (T A G) 1 5 3, 1 5 4、ムーブインバッファ (M I B) 1 5 4, 1 5 5、およびムーブインデータキュー (M I D Q) 1 5 6, 1 5 7 を有している。キャッシュメモリ制御回路 1 5 0 内の各要素の機能は、キャッシュメモリ制御回路 1 4 0 内の同名の要素と同じである。

【 0 0 4 2 】

次に、以上のような構成のキャッシュメモリ制御回路 1 4 0, 1 5 0 に対して、コア 1 1 1 からデータリードの要求が出力された場合の処理を説明する。コア 1 1 1 から出力された要求には、読み込むべきデータのメモリアドレスが含まれる。以下の例では、要求において「BANK0」のメモリアドレスが指定されているものとする。

10

【 0 0 4 3 】

キャッシュメモリ制御回路 1 4 0, 1 5 0 のそれぞれのムーブインポート 1 4 1, 1 5 1 は、コア 1 1 1 からの要求を受信する。なおムーブインポート 1 4 1 は、「BANK0」、「BANK1」に相当するメモリアドレスを指定した要求のみを受け付け、ムーブインポート 1 5 1 は、「BANK2」、「BANK3」に相当するメモリアドレスを指定した要求のみを受け付ける。この例では「BANK0」のメモリアドレスを指定した要求であるため、ムーブインポート 1 4 1 が要求を受け付ける。

【 0 0 4 4 】

ムーブインポート 1 4 1 は、コア 1 1 1 からの要求に応答して、キャッシュメモリ制御回路 1 4 0 におけるリード処理を起動する。リード処理は、タグメモリ 1 4 3 からメモリアドレスを検索する処理である。リード処理の指示は、パイプライン制御回路 1 4 2 に供給される。パイプライン制御回路 1 4 2 に供給される指示には、リード対象のデータのメモリアドレスが含まれる。

20

【 0 0 4 5 】

タグメモリ 1 4 3 には、データメモリ 1 2 1, 1 2 2 に格納されたデータのタグが格納されている。タグには、データのメモリアドレスとデータの状態とが含まれる。データの状態は、データが有効か否かを示すフラグビットや、格納されたデータが主記憶装置から読み込まれたままか書き換えられたかを示すフラグビットにより示される。

【 0 0 4 6 】

30

リード指示に示されたメモリアドレスが、タグメモリ 1 4 3 に含まれており、データメモリ内の対応するデータが有効である場合、要求されたデータがキャッシュに存在する (キャッシュヒット) と判定される。キャッシュヒットなら、そのデータのメモリアドレスを含むタグに対応するデータメモリ 1 2 1 内のアドレス (キャッシュアドレス) が、タグメモリ 1 4 3 から出力される。データメモリ 1 2 1 から出力されたデータは、バス 1 0 1 を介して緩衝制御回路 1 6 0 に入力される。

【 0 0 4 7 】

要求されたデータのメモリアドレスが、タグメモリ 1 4 3 に含まれていない場合、要求されたデータがキャッシュに存在しない (キャッシュミスヒット) と判定される。また、リード指示に示されたメモリアドレスが、タグメモリ 1 4 3 に含まれているが、データメモリ内の対応するデータが無効の場合にも、キャッシュミスヒットと判定される。キャッシュミスヒットなら、ムーブインバッファ 1 4 5 を経由して主記憶制御回路 1 3 1 にデータが要求される。

40

【 0 0 4 8 】

ムーブインバッファ 1 4 5 は、主記憶装置から読み出すデータのメモリアドレスを記憶する。ムーブインバッファ 1 4 5 に格納されたメモリアドレスは、主記憶制御回路 1 3 1 に渡される。すると、主記憶制御回路 1 3 1 によって、主記憶装置からメモリアドレスに対応するデータが読み出される。読み出されたデータは、主記憶制御回路 1 3 1 によってムーブインデータキュー 1 4 6 に格納される。データが格納されたことは、ムーブインバッファ 1 4 5 に伝えられる。すると、ムーブインバッファ 1 4 5 からパイプライン制御回

50

路 1 4 2 へ、キャッシュを登録するムーブイン要求 (M V I N) が発行される。このムーブイン要求 (M V I N) により、タグメモリ 1 4 3 に要求されたメモリアドレスが登録される。また、主記憶装置から読み出したデータがムーブインデータキュー 1 4 6 から出力され、データメモリ 1 2 1 に書き込まれると共に、バス 1 0 1 を介して緩衝制御回路 1 6 0 に入力される。ムーブイン要求 (M V I N) がパイプライン制御回路 1 4 2 を流れて要求に応じた処理が完了すると、ムーブインバッファ 1 4 5 が解放される。キャッシュメモリ制御回路 1 4 0 から緩衝制御回路 1 6 0 に入力されたデータは、緩衝制御回路 1 6 0 によってコア 1 1 1 に出力される。

【 0 0 4 9 】

なお、図 3 の例では、コア 1 1 1 からの要求に応じてデータをコア 1 1 1 に送信する構成を示しているが、他のコア 1 1 2 ~ 1 1 8 に対してもバス 1 0 1 ~ 1 0 4 を介してデータを転送できる。その場合、キャッシュメモリ制御回路 1 4 0、1 5 0 は、バス 1 0 1 ~ 1 0 4 を介して送信したデータがどのコア宛のデータなのかを、図示していない制御信号線を介して緩衝制御回路 1 6 0、1 7 0、1 8 0、1 9 0 に対して通知する。これにより、各緩衝制御回路 1 6 0、1 7 0、1 8 0、1 9 0 は、バス 1 0 1、1 0 2 を介して入力されたデータがどのコアへの転送データなのかを認識できる。

【 0 0 5 0 】

次に、緩衝制御回路 1 6 0 の機能を詳細に説明する。

図 4 は、緩衝制御回路の内部構成を示す図である。緩衝制御回路 1 6 0 は、データ入出力回路 1 6 1 と待ち合わせ制御回路 1 6 2 とを有する。

【 0 0 5 1 】

データ入出力回路 1 6 1 は、待ち合わせ制御回路 1 6 2 からの指示に従って、バス 1 0 1、1 0 2 から入力されたデータをコア 1 1 1、1 1 2 に転送する。なおデータ入出力回路 1 6 1 は、待ち合わせ制御回路 1 6 2 から入力されたデータを緩衝バッファに格納すべき旨の指示を受けると、データを内部の緩衝バッファに格納後、所定のタイミングでそのデータをコア 1 1 1、1 1 2 に転送する。

【 0 0 5 2 】

待ち合わせ制御回路 1 6 2 は、バス 1 0 1、1 0 2 を介したデータが入力された場合に、そのデータをバッファリングするか、バッファリングをせずにコア 1 1 1、1 1 2 に転送するかを判定する。待ち合わせ制御回路 1 6 2 は、判定結果に応じて、データ入出力回路 1 6 1 にデータの入出力指示を与える。

【 0 0 5 3 】

次に、データ入出力回路 1 6 1 の内部構成について詳細に説明する。

図 5 は、データ入出力回路の内部構成を示す図である。データ入出力回路 1 6 1 は、バス 1 0 1、1 0 2 からの入力データをコア 1 1 1、1 1 2 へ転送もしくは緩衝バッファ 3 0 へ書き込み、および緩衝バッファ 3 0 内に格納されたデータを読み出しコア 1 1 1、1 1 2 へ転送を行う回路構成となっている。

【 0 0 5 4 】

バス 1 0 1 から入力される「BANK0&1」の入力データは、3 つの A N D (論理積) 回路 4 1、5 1、6 1 に入力される。また、A N D 回路 4 1 へは、O R 回路 4 2 の出力信号が入力されている。O R 回路 4 2 には、待ち合わせ制御回路 1 6 2 からの 2 つの制御信号が入力されている。一方の制御信号は、「BANK0&1」のコア 1 1 1 宛のデータの緩衝バッファ 3 0 への書き込みを指示する信号である。他方の制御信号は、「BANK0&1」のコア 1 1 2 宛のデータの緩衝バッファ 3 0 への書き込みを指示する信号である。

【 0 0 5 5 】

バス 1 0 2 から入力される「BANK2&3」の入力データは、3 つの A N D (論理積) 回路 4 3、5 3、6 3 に入力される。また、A N D 回路 4 3 へは、O R 回路 4 4 の出力信号が入力されている。O R 回路 4 4 には、待ち合わせ制御回路 1 6 2 からの 2 つの制御信号が入力されている。一方の制御信号は、「BANK2&3」のコア 1 1 1 宛のデータの緩衝バッファ 3 0 への書き込みを指示する信号である。他方の制御信号は、「BANK2&3」のコア 1 1

2宛のデータの緩衝バッファ30への書き込みを指示する信号である。

【0056】

2つのAND回路41, 43の出力信号は、OR回路45に入力されている。OR回路45の出力信号は、緩衝バッファ30に入力されている。緩衝バッファ30は、例えば32バイトのレジスタを4段に並べた128バイトの記憶領域である。緩衝バッファ30の出力は、AND回路52, 62に入力される。

【0057】

AND回路51には、バス101からのデータ入力に加え、待ち合わせ制御回路162からの制御信号が入力されている。この制御信号は、「BANK0&1」からデータのコア111への転送を指示する信号である。AND回路52には、緩衝バッファ30からの出力データに加え、待ち合わせ制御回路162からの制御信号が入力されている。この制御信号は、緩衝バッファ30内のデータのコア111への転送を指示する信号である。AND回路53には、バス102からのデータ入力に加え、待ち合わせ制御回路162からの制御信号が入力されている。この制御信号は、「BANK2&3」からデータのコア111への転送を指示する信号である。

【0058】

3つのAND回路51~53の出力信号は、OR回路54に入力されている。OR回路54の出力信号は、コア111への入力となる。

AND回路61には、バス101からのデータ入力に加え、待ち合わせ制御回路162からの制御信号が入力されている。この制御信号は、「BANK0&1」からデータのコア112への転送を指示する信号である。AND回路62には、緩衝バッファ30からの出力データに加え、待ち合わせ制御回路162からの制御信号が入力されている。この制御信号は、緩衝バッファ30内のデータのコア112への転送を指示する信号である。AND回路63には、バス102からのデータ入力に加え、待ち合わせ制御回路162からの制御信号が入力されている。この制御信号は、「BANK2&3」からデータのコア112への転送を指示する信号である。

【0059】

3つのAND回路61~63の出力信号は、OR回路64に入力されている。OR回路64の出力信号は、コア112への入力となる。

このような回路構成であれば、「BANK0&1」と「BANK2&3」とのどちらか一方のデータをコアへ転送中でも、転送先のコアが異なれば緩衝バッファ30を介さずにもう一方のデータをコアへ転送可能である。同様に緩衝バッファ30からの読み出しデータをコアへ転送中でも、入力されたデータの転送先となるコアが異なれば、入力されたデータを直ぐに転送可能である。

【0060】

なお、図5に示した各種の指示を示す制御信号は、該当する指示が出されていない状態ではオフ(信号がローレベル)であり、該当する指示が出されたときにオン(信号がハイレベル)となる正論理による信号である。

【0061】

このような回路構成のデータ入出力回路161に対して、待ち合わせ制御回路162からデータ転送に関する指示が入力されることで、適切なデータ転送が実行される。データ転送指示は、バス101, 102を介して緩衝制御回路160にデータが入力されたときに実行される。

【0062】

図6は、待ち合わせ制御回路のデータ転送指示手順を示すフローチャートである。データ転送指示手順は、バス101, 102に対してデータ入力が行われた際に、入力された各データについて行われる。以下、データ転送指示の対象となるデータを「転送対象データ」とし、図6に示す処理をステップ番号に沿って説明する。

【0063】

[ステップS11] 待ち合わせ制御回路162は、転送対象データが入力されたバスと

は他のバスに対して先にデータ入力あり、そのデータが転送中か否かを判断する。他のバスからのデータが転送中でなければ、処理がステップ S 1 2 に進められる。他のバスからのデータが転送中であれば、処理がステップ S 1 5 に進められる。

【 0 0 6 4 】

[ステップ S 1 2] 待ち合わせ制御回路 1 6 2 は、他のバスからのデータが転送中ではない場合、転送対象データのデータ入力と他のバスからのデータ入力とが同時か否かを判断する。データ入力が同時でなければ、処理がステップ S 1 4 に進められる。データ入力が同時であれば、処理がステップ S 1 3 に進められる。

【 0 0 6 5 】

[ステップ S 1 3] 待ち合わせ制御回路 1 6 2 は、転送対象データのデータ入力と他のバスからのデータ入力とが同時の場合、同時に入力された他のデータよりも、転送対象データの方がコアに対する実装距離が遠いか否かを判断する。ここで実装距離とは、L S I 1 0 0 におけるコアとデータメモリとの間を伝搬する信号が通る伝送路の距離である。第 2 の実施の形態では、実装距離が遠い方のデータメモリからのデータ転送を優先する。そこで転送対象データの方が実装距離が遠ければ、処理がステップ S 1 4 に進められる。転送対象データの方が実装距離が近ければ、処理がステップ S 1 5 に進められる。

10

【 0 0 6 6 】

[ステップ S 1 4] 待ち合わせ制御回路 1 6 2 は、転送対象データが入力されたバスから送信先のコアへの、転送対象データの転送指示をデータ入出力回路 1 6 1 に対して出力する。その後、処理が終了する。

20

【 0 0 6 7 】

[ステップ S 1 5] 待ち合わせ制御回路 1 6 2 は、転送対象データの緩衝バッファ 3 0 への書き込み指示をデータ入出力回路 1 6 1 に対して出力する。このとき待ち合わせ制御回路 1 6 2 は、緩衝バッファ 3 0 に書き込んだデータの転送先を示す情報を、内部のレジスタやラッチ回路などで保持しておく。

【 0 0 6 8 】

[ステップ S 1 6] 待ち合わせ制御回路 1 6 2 は、先行する他のバスのデータ、または同時に他のバスに入力されたデータ（衝突データ）の転送が終了したか否かを判断する。転送が終了していなければ、ステップ S 1 6 の処理が繰り返される。転送が終了したら、処理がステップ S 1 7 に進められる。

30

【 0 0 6 9 】

[ステップ S 1 7] 待ち合わせ制御回路 1 6 2 は、緩衝バッファ 3 0 内のデータのコアへの転送指示をデータ入出力回路 1 6 1 に対して出力する。なお、データの転送先は、ステップ S 1 5 で保持した情報により判別される。

【 0 0 7 0 】

このように、データ入出力回路 1 6 1 に対して待ち合わせ制御回路 1 6 2 が指示を与えることで 2 つのキャッシュメモリ制御回路 1 4 0 , 1 5 0 から 1 つのコアへ同時期にデータが送信されても、それらのデータを順番にコアに入力できる。

【 0 0 7 1 】

図 7 は、「BANK0&1」と「BANK2&3」のデータ入力が行われた場合のデータ入出力を示すシーケンス図である。この例では、2 つの入力データの送信先は、共にコア識別番号「CORE0」のコア 1 1 1 であるものとする。なお図 7 における縦線の間隔は、バス 1 0 1 , 1 0 2 の動作クロックの 1 サイクルの時間に相当する。

40

【 0 0 7 2 】

時刻 t 1 に、「BANK0&1」側のデータがバス 1 0 1 を介して緩衝制御回路 1 6 0 に入力されている。同時に、「BANK2&3」側のデータがバス 1 0 2 を介して緩衝制御回路 1 6 0 に入力されている。図 2 に示すように、コア 1 1 1 への実装距離は、「BANK2&3」のデータメモリ 1 2 3 , 1 2 4 の方が遠い。

【 0 0 7 3 】

そこで、待ち合わせ制御回路 1 6 2 は、時刻 t 1 において、データ入出力回路 1 6 1 に

50

対して「BANK2&3」からのデータのコア 1 1 1 への転送指示を出力する。すなわち、待ち合わせ制御回路 1 6 2 は、AND 回路 5 3 に入力する制御信号をオンにする。これにより、AND 回路 5 3 と OR 回路 5 4 とを経由して、「BANK2&3」側のデータが優先的（待ち合わせなしで）にコア 1 1 1 に転送される。

【 0 0 7 4 】

また、待ち合わせ制御回路 1 6 2 は、時刻 t_1 において、「BANK2&3」側のデータ転送が実行中（ビジー状態）になったことにより、データ入出力回路 1 6 1 に対して「BANK0&1」のデータの緩衝バッファ 3 0 への書き込みを指示する。すなわち、待ち合わせ制御回路 1 6 2 は、OR 回路 4 2 に入力する一方の制御信号をオンにする。これにより、「BANK0&1」側のデータが、OR 回路 4 2、AND 回路 4 1、OR 回路 4 5 を経由して、緩衝バッファ 3 0 に書き込まれる。

10

【 0 0 7 5 】

時刻 t_1 から 4 サイクルの時間経過した時刻 t_2 に、「BANK2&3」側のデータのコア 1 1 1 への転送が終了する（「BANK2&3」のビジー状態解消）。すると待ち合わせ制御回路 1 6 2 は、データ入出力回路 1 6 1 に対して、緩衝バッファ 3 0 内のデータのコア 1 1 1 への転送を指示する。すなわち、待ち合わせ制御回路 1 6 2 は、AND 回路 5 2 に入力する制御信号をオンにする。これにより、緩衝バッファ 3 0 に格納されていた「BANK0&1」側のデータが、AND 回路 5 2、OR 回路 5 4 を経由してコア 1 1 1 に転送される。そして、時刻 t_2 から 4 サイクルの時間経過した時刻 t_3 に、「BANK0&1」側のデータのコア 1 1 1 への転送が終了する。

20

【 0 0 7 6 】

このように「BANK0&1」と「BANK2&3」のデータが同時に転送されるケースでは、コア識別番号が「CORE0」のコア 1 1 1 から見て距離が遠い「BANK2&3」のデータは待ち合わせなしでそのまま転送される。また、コア識別番号が「CORE0」のコア 1 1 1 から距離が近い「BANK0&1」からのデータは、緩衝バッファ 3 0 へ順番に書き込まれる。これにより、バス 1 0 1、1 0 2 から同時に出力された転送データが、コア 1 1 1 に対して「BANK2&3」側のデータ、「BANK0&1」側のデータの順で入力される。

【 0 0 7 7 】

図 8 は、「BANK0&1」のデータ入力後に続けて「BANK2&3」のデータ入力が行われた場合のデータ入出力を示すシーケンス図である。この例では、2つの入力データの送信先は、共にコア識別番号「CORE0」のコア 1 1 1 であるものとする。なお図 8 における縦線の間隔は、バス 1 0 1、1 0 2 の動作クロックの 1 サイクルの時間に相当する。

30

【 0 0 7 8 】

時刻 t_{11} に、「BANK0&1」側のデータがバス 1 0 1 を介して緩衝制御回路 1 6 0 に入力されている。そこで、待ち合わせ制御回路 1 6 2 は、時刻 t_{11} において、データ入出力回路 1 6 1 に対して「BANK0&1」からのデータのコア 1 1 1 への転送指示を出力する。すなわち、待ち合わせ制御回路 1 6 2 は、AND 回路 5 1 に入力する制御信号をオンにする。これにより、AND 回路 5 1 と OR 回路 5 4 とを経由して、「BANK0&1」側のデータのコア 1 1 1 への転送が待ち合わせなしで開始される。このデータ転送には、動作クロックで 4 サイクルの時間がかかる。

40

【 0 0 7 9 】

時刻 t_{11} から 2 サイクルの時間経過後の時刻 t_{12} に、「BANK2&3」側のデータがバス 1 0 2 を介して緩衝制御回路 1 6 0 に入力されている。この時点では、「BANK0&1」側のデータ転送が実行中（ビジー状態）である。そこで、待ち合わせ制御回路 1 6 2 は、時刻 t_{12} において、データ入出力回路 1 6 1 に対して「BANK2&3」側のデータの緩衝バッファ 3 0 への書き込みを指示する。すなわち、待ち合わせ制御回路 1 6 2 は、OR 回路 4 4 に入力する一方の制御信号をオンにする。これにより、「BANK2&3」側のデータが、OR 回路 4 4、AND 回路 4 3、OR 回路 4 5 を経由して、緩衝バッファ 3 0 に書き込まれる。

【 0 0 8 0 】

50

時刻 t_{11} から 4 サイクルの時間経過した時刻 t_{13} に、「BANK0&1」側のデータのコア 1 1 1 への転送が終了する（「BANK0&1」のビジー状態解消）。すると待ち合わせ制御回路 1 6 2 は、データ入出力回路 1 6 1 に対して、緩衝バッファ 3 0 内のデータのコア 1 1 1 への転送を指示する。すなわち、待ち合わせ制御回路 1 6 2 は、AND 回路 5 2 に入力する制御信号をオンにする。これにより、緩衝バッファ 3 0 に格納されていた「BANK2&3」側のデータが、AND 回路 5 2、OR 回路 5 4 を経由してコア 1 1 1 に転送される。そして、時刻 t_{13} から 4 サイクルの時間経過した時刻 t_{14} に、「BANK2&3」側のデータのコア 1 1 1 への転送が終了する。

【0081】

このように「BANK0&1」のデータを 4 サイクルでコア 1 1 1 へ転送後に、緩衝バッファ 3 0 に保持されていた「BANK2&3」のデータが 4 サイクルで順番に読み出され、同時にコア 1 1 1 へ転送される。すなわち、どちらか片方のデータ転送中にもう一方のデータが衝突した場合には、遅れて到着したデータが緩衝バッファ 3 0 へ書き込まれる。

【0082】

図 7、図 8 に示したように、「BANK0&1」と「BANK2&3」とのデータが同時衝突・ずれ衝突ケースどちらも、コアに設けるインターフェースは 1 系統であっても、コアに対して 2 系統のデータ転送が可能となる。

【0083】

なお、緩衝バッファ 3 0 の記憶容量は、1 キャッシュブロックサイズ（データ要求に対する応答で転送されるデータ量）分である。各バス 1 0 1、1 0 2 のデータ転送が連続して発生した場合でも、1 キャッシュブロックサイズ分の緩衝バッファ 3 0 で、適切に緩衝制御を行うことができる。

【0084】

図 9 は、「BANK0&1」と「BANK2&3」とから「CORE0」と「CORE1」へのデータ転送が連続して行われた場合のデータ入出力を示すシーケンス図である。なお図 9 における縦線の間隔は、バス 1 0 1、1 0 2 の動作クロックの 1 サイクルの時間に相当する。

【0085】

時刻 t_{21} に、「BANK0&1」側から「CORE0」へ送信するデータが、バス 1 0 1 を介して緩衝制御回路 1 6 0 に入力されている。同時に、「BANK2&3」側から「CORE0」へ送信するデータがバス 1 0 2 を介して緩衝制御回路 1 6 0 に入力されている。これらのデータの転送手順は、図 7 に示した通りである。

【0086】

図 9 の例では、時刻 t_{21} から 4 サイクルの時間経過した時刻 t_{22} に、「BANK0&1」側から「CORE1」へ送信するデータがバス 1 0 1 を介して緩衝制御回路 1 6 0 に入力されている。同時に、「BANK2&3」側から「CORE1」へ送信するデータがバス 1 0 2 を介して緩衝制御回路 1 6 0 に入力されている。

【0087】

そこで、待ち合わせ制御回路 1 6 2 は、時刻 t_{22} において、データ入出力回路 1 6 1 に対して「BANK2&3」からのデータのコア 1 1 2 への転送指示を出力する。すなわち、待ち合わせ制御回路 1 6 2 は、AND 回路 6 3 に入力する制御信号をオンにする。これにより、AND 回路 6 3 と OR 回路 6 4 とを経由して、「BANK2&3」側のデータが優先的（待ち合わせなしで）にコア 1 1 2 に転送される。同時に、「BANK0&1」側から「CORE0」へ送信するデータが、緩衝バッファ 3 0 からコア 1 1 1 へ転送される。緩衝バッファ 3 0 からコア 1 1 1 へのデータ転送は、AND 回路 5 3、OR 回路 5 4 を経由して行われ、「BANK2&3」からのデータのコア 1 1 2 への転送経路とは重複しない。そのため、2 つのコア 1 1 1、1 1 2 へのデータ転送を並行して行うことができる。

【0088】

また、待ち合わせ制御回路 1 6 2 は、時刻 t_{22} において、「BANK2&3」側のデータ転送が実行中（ビジー状態）になったことにより、データ入出力回路 1 6 1 に対して「BANK0&1」のデータの緩衝バッファ 3 0 への書き込みを指示する。すなわち、待ち合わせ制御

10

20

30

40

50

回路 1 6 2 は、O R 回路 4 2 に入力する一方の制御信号のオン状態を維持する。これにより、「BANK0&1」側のデータが、O R 回路 4 2、A N D 回路 4 1、O R 回路 4 5 を経由して、緩衝バッファ 3 0 に書き込まれる。

【 0 0 8 9 】

ここで、図 9 の例では、緩衝バッファ 3 0 の記憶容量が 1 2 8 バイトであり、キャッシュメモリからコアへのデータ要求（ムーブイン要求）に対する応答で転送されるデータ量が 1 2 8 バイトである。また、動作クロックで 1 サイクル当たりのデータ転送量は 3 2 バイトである。すると、時刻 t_{22} からの 1 サイクルで緩衝バッファ 3 0 から 3 2 バイトのデータがコア 1 1 1 に転送され、緩衝バッファ 3 0 には 3 2 バイト分の空き領域が生じる。その空き領域に、時刻 t_{22} からの 1 サイクルで「BANK0&1」側のデータの先頭の 3 2 バイトが格納される。その後、1 サイクルごとに、緩衝バッファ 3 0 から 3 2 バイト分の出力と、緩衝バッファ 3 0 への 3 2 バイト分の入力とが繰り返される。このように、緩衝バッファ 3 0 が 1 回当たりの転送データ量と同じサイズの記憶容量を有していれば、図 9 に示したような連続のデータ転送が発生した場合であっても、緩衝バッファ 3 0 への書き込み待ちを発生させずにすむ。

【 0 0 9 0 】

時刻 t_{22} から 4 サイクルの時間経過した時刻 t_{23} に、「BANK2&3」側のデータのコア 1 1 2 への転送が終了する（「BANK2&3」のビジー状態解消）。すると待ち合わせ制御回路 1 6 2 は、データ入出力回路 1 6 1 に対して、緩衝バッファ 3 0 内のデータのコア 1 1 2 への転送を指示する。すなわち、待ち合わせ制御回路 1 6 2 は、A N D 回路 6 2 に入力する制御信号をオンにする。これにより、緩衝バッファ 3 0 に格納されていた「BANK0&1」側のデータが、A N D 回路 6 2、O R 回路 6 4 を経由してコア 1 1 2 に転送される。そして、時刻 t_{23} から 4 サイクルの時間経過した時刻 t_{24} に、「BANK0&1」側のデータのコア 1 1 2 への転送が終了する。

【 0 0 9 1 】

このようにして、バス 1 0 1、1 0 2 から同時に出力された転送データが、コア 1 1 1、1 1 2 のそれぞれに対して、「BANK2&3」側のデータ、「BANK0&1」側のデータの順で順番に入力される。

【 0 0 9 2 】

図 9 に示したように、「BANK0&1」から「CORE0」のコア 1 1 1 と「CORE1」のコア 1 1 2 へ、「BANK2&3」から「CORE0」のコア 1 1 1 と「CORE1」のコア 1 1 2 へデータが連続して転送されるケースでは、データ転送の待ち合わせが行われる。待ち合わせにより緩衝バッファ 3 0 に書き込まれたデータは、送信先のコアへの他のデータ転送が終了すると、送信先のコアに転送される。このように 2 つのコア 1 1 1、1 1 2 で緩衝制御回路 1 6 0 を共用しても、緩衝バッファ 3 0 の記憶容量は、1 キャッシュブロックサイズで足りる。しかも、緩衝制御回路 1 6 0 を 1 つのコア 1 1 1、1 1 2 で共用したからといって、待ち合わせが多く発生することもない。

【 0 0 9 3 】

また、緩衝バッファ 3 0 への書き込みは、図 7 のようにデータが同時に衝突したケース・もしくは図 8 のようにデータがずれて衝突したケースを契機に、データ転送にかかる 4 サイクルの時間だけ有効となる。緩衝バッファ 3 0 からの読み出しは、同時衝突では優先される遠い方のデータ、ずれ衝突では先行データの転送完了の次のタイミングから 4 サイクルの時間だけ有効となり、緩衝バッファ 3 0 に書き込まれた順番に読み出され、コアへ転送される。すなわち、緩衝バッファ 3 0 からは先入れ先出しで読み出される。

【 0 0 9 4 】

しかも、図 5 に示すように、緩衝バッファ 3 0 の書き込みと読み出しの回路が分かれているので、緩衝バッファ 3 0 へ 1 2 8 バイトの全データが揃うのを待つ必要がない。また書き込み中と読み出しを同時に行っても、データが混じったり入れ替わったりすることもない。

【 0 0 9 5 】

10

20

30

40

50

ところで、図 9 の例では、「BANK0&1」側と「BANK2&3」側共に、1 番目のデータの送信先がコア 1 1 1（「CORE0」）であり、2 番目のデータの送信先がコア 1 1 2（「CORE1」）である。そのため、緩衝バッファ 3 0 の記憶容量は、データ要求（ムーブイン要求）に対する応答で転送されるデータ量と同じだけあればよい。もし、図 9 に示す 4 つのデータの転送先がすべてコア 1 1 1 であるような状況が発生し得るのであれば、緩衝バッファ 3 0 の記憶容量を、要求 1 回当たりの転送データ量の 2 倍以上にしておくこともできる。ただし、緩衝バッファ 3 0 の記憶容量を大きくすることは、LSI 1 0 0 の面積の増大につながる。そこで、第 2 の実施の形態では、キャッシュメモリ制御回路 1 4 0, 1 5 0 において、同一コアからの連続する要求の受付を制限する。これにより、緩衝バッファ 3 0 の記憶容量を少なく抑えておくことが可能となる。受付を制限する時間は、例えば、少なくともキャッシュメモリ制御部からコアへのデータ転送に要する時間（バスを占有する時間）とする。本実施の形態では、キャッシュメモリ制御部からコアへのデータ転送に要する時間に対して、緩衝制御回路に接続されたバスの数を乗算した時間を、受付を制限する時間とする。具体的には、4 サイクルでデータが転送され、緩衝制御回路に 2 つのバスが接続されているため、制限する時間は 8 サイクルとなる。8 サイクルの制限時間を設けることで、バスに出力されたデータについて、コアに直接転送することも緩衝バッファ 3 0 に格納することもできないような事態を回避できる。

10

【0096】

また、図 2 に示したように、バス 1 0 1, 1 0 2 は、コア 1 1 1, 1 1 2, 1 1 5, 1 1 6 へのデータ転送で共用される。そのため、バス 1 0 1, 1 0 2 を共用するコア 1 1 1, 1 1 2, 1 1 5, 1 1 6 を 1 つのグループにまとめる。同様に、バス 1 0 3, 1 0 4 は、コア 1 1 3, 1 1 4, 1 1 7, 1 1 8 へのデータ転送で共用される。そのため、バス 1 0 3, 1 0 4 を共用するコア 1 1 3, 1 1 4, 1 1 7, 1 1 8 を 1 つのグループにまとめる。そして、キャッシュメモリ制御回路 1 4 0, 1 5 0 は、同一グループへのデータの送出を、データ転送に要する時間（バスを占有する時間）内に 1 回に制限する。本実施の形態では、4 サイクルに 1 回に制限される。これにより、バス 1 0 1 ~ 1 0 4 上でのデータ転送の競合が防止される。

20

【0097】

図 1 0 は、複数のキャッシュメモリ制御回路から同一コアへデータを転送する場合の処理を示すシーケンス図である。図 1 0 には、キャッシュメモリ制御回路 1 4 0, 1 5 0 によるパイプライン処理と、緩衝制御回路との動作例が示されている。キャッシュメモリ制御回路 1 4 0, 1 5 0 は、同一コア宛のデータ送出は 8 サイクルに 1 回に制限している。またキャッシュメモリ制御回路 1 4 0, 1 5 0 は、コア 1 1 1, 1 1 2, 1 1 5, 1 1 6 を 1 つのグループ、コア 1 1 3, 1 1 4, 1 1 7, 1 1 8 を 1 つのグループとし、同一グループ内のコア宛のデータ送出は 4 サイクルに 1 回に制限している。

30

【0098】

なお、2 つのキャッシュメモリ制御回路 1 4 0, 1 5 0 は独立に動作するため、「BANK 0&1」からのデータと「BANK2&3」からのデータが競合することがある。競合した場合にはどちらか一方のデータを緩衝バッファ 3 0 で待ち合わせればよい。また「BANK0&1」→「CORE0」よりも「BANK2&3」→「CORE0」の方が転送距離が長い。そのため、転送にかかるレイテンシが 3 サイクル多くなっている。

40

【0099】

図 1 0 の例では、時刻 t_{31} に、コア 1 1 1 から「BANK2」内のデータの読み出しの要求（READ）が出力されている。するとキャッシュメモリ制御回路 1 5 0 では、時刻 t_{31} から 8 サイクルの時間経過するまで、コア 1 1 1 からのデータ読み出し要求（READ）、およびキャッシュを登録するムーブイン要求（MVIN）のパイプライン制御回路 1 5 2 への供給が禁止される。また、キャッシュメモリ制御回路 1 5 0 では、時刻 t_{31} から 4 サイクルの時間経過するまで、コア 1 1 1 と同じグループの他のコア 1 1 2, 1 1 5, 1 1 6 からの要求（READ / MVIN）のパイプライン制御回路 1 5 2 への供給が禁止される。

50

【0100】

時刻 t_{32} に、コア 111 から「BANK0」内のデータの読み出しの要求 (READ) が出力されている。するとキャッシュメモリ制御回路 140 では、時刻 t_{31} から 8 サイクルの時間経過するまで、コア 111 からのデータ読み出し要求 (READ)、およびキャッシュを登録するムーブイン要求 (MVIN) のパイプライン制御回路 142 への供給が禁止される。また、キャッシュメモリ制御回路 140 では、時刻 t_{31} から 4 サイクルの時間経過するまで、コア 111 と同じグループの他のコア 112, 115, 116 からの要求 (READ / MVIN) のパイプライン制御回路 142 への供給が禁止される。

【0101】

キャッシュメモリ制御回路 150 では、時刻 t_{33} から 4 サイクルかけてデータメモリ 123 からデータが読み出される。読み出されたデータは、6 サイクルかけてバス 102 で転送され、時刻 t_{36} から 4 サイクルかけて緩衝制御回路 160 に入力される。入力されたデータは、遅延なしでコア 111 に入力される。

【0102】

キャッシュメモリ制御回路 140 では、時刻 t_{34} から 4 サイクルかけてデータメモリ 121 からデータが読み出される。読み出されたデータは、3 サイクルかけてバス 101 で転送され、時刻 t_{36} から 4 サイクルかけて緩衝制御回路 160 に入力される。

【0103】

ここで、キャッシュメモリ制御回路 140 から緩衝制御回路 160 へのデータ入力と、キャッシュメモリ制御回路 150 から緩衝制御回路 160 へのデータ入力とが、共に時刻 t_{36} から開始されており、同時である。緩衝制御回路 160 では、実装距離が長いキャッシュメモリ制御回路 150 からの入力データの転送を優先するため、キャッシュメモリ制御回路 140 から入力されたデータは、緩衝バッファ 30 に格納される。その後、緩衝バッファ 30 に格納された「BANK0」のデータは、時刻 t_{38} から 4 サイクルかけてコア 111 に転送される。

【0104】

さらに時刻 t_{36} には、コア 111 から「BANK2」内のデータの読み出しの要求 (READ) が出力され、その 1 サイクル後の時刻 t_{37} には、コア 111 から「BANK0」内のデータの読み出しの要求 (READ) が出力されている。「BANK0」→「CORE0」よりも「BANK2」→「CORE0」のレイテンシの方が 3 サイクル多い。「BANK0」から読み出されたデータの方が先の時刻 t_{40} に緩衝制御回路 160 に入力される。このデータは遅延なくコア 111 に入力される。時刻 t_{40} から 2 サイクル経過後の時刻 t_{41} に、「BANK2」から読み出されたデータが緩衝制御回路 160 に入力される。このデータは、緩衝バッファ 30 に格納された後、時刻 t_{42} から 4 サイクルかけてコア 111 に入力される。

【0105】

このように、同一コア宛のデータ送出が 8 サイクルに 1 回、また同一バスを使用するコア同士は 4 サイクルに 1 回に制限する。これによって、1 キャッシュブロックサイズの緩衝バッファを 2 コアに対して 1 つ実装するだけで、2 系統のデータ転送をコアの受け口は 1 系統で行うことが可能である。

【0106】

なお、マルチコアの LSI 100 は、コンピュータなどのシステムボードに実装できる。

図 11 は、マルチコアの LSI チップを実装したシステムモジュールの例を示す図である。システムモジュール 200 には、LSI 100 に加え、バンクに分けられたメモリ 211 ~ 214 が実装されている。各メモリ 211 ~ 214 は、システムモジュール 200 の主記憶装置である。LSI 100 で実行されるプログラムの命令や、実行に必要なデータなどがメモリ 211 ~ 214 に格納される。

【0107】

各メモリ 211 ~ 214 には、主記憶装置のバンクに対応付けられている。図 11 の例では、メモリ 211 が「BANK0」、メモリ 212 が「BANK1」、メモリ 213 が「BANK2」、

10

20

30

40

50

、メモリ 2 1 4 が「BANK3」に対応する。各メモリ 2 1 1 ~ 2 1 4 は、L S I 1 0 0 内のバンク番号が同じ主記憶制御回路 1 3 1 ~ 1 3 4 に接続されている。

【 0 1 0 8 】

L S I 1 0 0 は、コア 1 1 1 ~ 1 1 8 から要求されたデータが、キャッシュメモリとして使用する内部のデータメモリ 1 2 1 ~ 1 2 4 内に存在しない場合、該当するデータをメモリ 2 1 1 ~ 2 1 4 から取得する。

【 0 1 0 9 】

以上説明したように、第 2 の実施の形態に係る L S I 1 0 0 では、チップの肥大化を抑制しながら、複数のキャッシュメモリ制御回路 1 4 0 , 1 5 0 を用いた各コア 1 1 1 ~ 1 1 8 での効率的なデータ取得が可能となる。すなわち、キャッシュメモリ制御回路 1 4 0 , 1 5 0 を用いた効率的なキャッシュ処理を行いながらも、コアのデータ入力用ポートが 1 つで済み、L S I 1 0 0 全体の肥大化が抑制される。

【 0 1 1 0 】

ここで、コア内部でデータの緩衝制御を行う場合と第 2 の実施の形態のようにコアの外で緩衝制御を行う場合との違いについて説明する。

コア内部でデータの緩衝制御を行う場合、以下のような構成となる。

【 0 1 1 1 】

コアは、データ要求（ムーブイン要求）の際に M I B を確保し、データが応答されたらそれに対し書き込みを行う。M I B の数は、1 つのコア当たり 1 2 8 バイト × 1 0 エントリである。コア内部で緩衝制御するために、コアは M I B に対して 2 ポートからライトする。2 系統から転送されるデータを選択して M I B に書き込む論理回路およびその周辺回路は、エントリごとに設けられる。そのため、図 4 , 図 5 に示した第 2 の実施の形態に係る緩衝制御回路 1 6 0 よりも複雑な回路にならざるを得ない。このようなコア内部の回路増加が L S I チップのサイズの増大化に及ぼす影響は、コア数が増加するほど大きくなる。

【 0 1 1 2 】

しかも、データバスは、コアの M I B まで 2 系統配線しなければならない。配線によるノイズを考慮すると、狭小な領域で 2 系統の配線は難しい。そのため配線エリアの確保が必要になり、L S I チップの面積増加の原因となる。

【 0 1 1 3 】

第 2 の実施の形態では、図 4 , 図 5 に示したような比較的簡単な回路で緩衝制御を行うことができ、コア内部で緩衝制御を行う場合よりも、L S I チップの肥大化を抑制できる。しかも、複数のコアで 1 つの緩衝制御回路を共用することができるため、L S I チップに第 2 の実施の形態に係る緩衝制御回路を追加実装しても、L S I チップのサイズの肥大化に与える影響は極めて小さい。

【 0 1 1 4 】

さらに、緩衝制御回路とコアとの間の配線は、1 系統でよい。配線が 1 系統であれば、狭小な領域に 2 系統の配線を形成する場合のようなノイズの問題が発生せず、狭小な領域での配線が可能となる。

【 0 1 1 5 】

また、第 2 の実施の形態によれば、緩衝制御による副作用を最小限に抑えることができる。

まず、緩衝制御回路を持たない場合を考える。この場合、例えば、複数のキャッシュメモリ制御回路がコアへデータ転送を行うためにバスを使用するサイクルを、緩衝制御回路を不要とするだけの十分な長さに設定する。すなわち、コアにおいて衝突が発生しないだけの十分に長い間隔をあけて、データをコアに転送する。十分な間隔を開けることでデータの衝突は生じないが、衝突しないケースでも無駄に待ち合わせをすることとなり、レイテンシは低下する。レイテンシの低下は、L S I チップの性能低下になる。

【 0 1 1 6 】

第 2 の実施の形態では、緩衝制御回路を設けることによってレイテンシの低下が抑制さ

10

20

30

40

50

れている。すなわち、第2の実施の形態では、複数のキャッシュメモリ制御回路それぞれがコアにデータを転送する際に、転送されるデータのコアでの衝突の回避を目的とした待ち時間を設ける必要がない。ここでレイテンシは、要求を発行して応答が返されるまでの時間である。そのため、緩衝制御による一部のデータが緩衝バッファに格納された後に転送されたとしても、キャッシュメモリ制御回路から遅延無くデータが出力されている限り、レイテンシの低下とはみなさない。

【0117】

なお、第2の実施の形態では、キャッシュメモリ制御回路が緩衝制御回路のバッファリングの能力に合わせて、連続する要求の受け付けを禁止する。具体的には、同一コアからの要求のパイプライン制御回路への投入が、8サイクルに1回に制限される。スループットに関しては、この投入制限が性能低下の要因ともなり得る。しかし、第2の実施の形態では要求されたメモリアドレスの下位ビットによりバンク分けを行っており、万遍なくアクセスが発生すると考えられる。よって片方のデータメモリからのデータ転送に偏るケースは希である。そのため、同一コアからの要求の投入制限による性能低下は微少である。

【0118】

以上のことから、第2の実施の形態によれば、性能的な副作用をほとんど起こすことなく、かつチップサイズの増大を最小限に抑制して、複数のキャッシュメモリ制御回路を用いた複数のバンクへの並列のキャッシュによるデータアクセス効率の向上が可能となる。

【0119】

ところで上記の第2の実施の形態では、LSI100としてマルチコアLSIの例を示したが、コアが1つだけの場合であっても、コアへのデータ転送の並列化の必要があれば、同様に実現できる。

【0120】

また上記の第2の実施の形態では、バンク分けしたキャッシュメモリを有した例であるため、2つのキャッシュメモリ制御回路が異なるバスを介してコアにデータを転送している。しかし、キャッシュメモリ制御回路は、コアからの要求に応答してデータをコアに転送するメモリ制御回路の一例である。例えば、キャッシュメモリを有せずに主記憶装置から取得したデータをコアに転送する場合もある。この場合、主記憶制御回路が、各コアからの連続した要求を禁止する禁止期間の制御を行う。

【0121】

以上、実施の形態を例示したが、実施の形態で示した各部の構成は同様の機能を有する他のものに置換することができる。また、他の任意の構成物や工程が付加されてもよい。さらに、前述した実施の形態のうちの任意の2以上の構成(特徴)を組み合わせたものであってもよい。

【0122】

以上説明した実施の形態の主な技術的特徴は、以下の付記の通りである。

(付記1) 演算を行う演算処理部と、

前記演算処理部からの第1の要求を入力する第1のメモリ制御部と、

前記演算処理部からの第2の要求を入力する第2のメモリ制御部と、

前記第1のメモリ制御部に接続され、前記第1のメモリ制御部の制御により、前記第1の要求に応じた第1のデータを、第1のバスを介して出力する第1のメモリ部と、

前記第2のメモリ制御部に接続され、前記第2のメモリ制御部の制御により、前記第2の要求に応じた第2のデータを、第2のバスを介して出力する第2のメモリ部と、

緩衝バッファ部を有し、前記第1のバスを介して前記第1のメモリ部に接続されると共に、前記第2のバスを介して前記第2のメモリ部に接続され、前記第1のバスを介した前記第1のデータの出力と前記第2のバスを介した前記第2のデータの出力が同一期間に行われる場合、前記第1のデータを前記演算処理部に転送し、前記第2のデータを前記緩衝バッファ部に格納し、前記第1のデータの転送の終了後、前記第2のデータを前記緩衝バッファ部から前記演算処理部に転送する緩衝制御部と、

を有することを特徴とする演算処理装置。

【 0 1 2 3 】

(付記 2) 前記演算処理装置において、

前記緩衝制御部は、前記第 1 および第 2 のメモリ部から同時にデータが出力された場合、前記第 1 および第 2 のメモリ部のうち、データ転送時間が大きい一のメモリ部からのデータを前記第 1 のデータとして前記演算処理部に転送し、他のメモリ部からのデータを前記第 2 のデータとして前記緩衝バッファ部に格納することを特徴とする付記項 1 記載の演算処理装置。

【 0 1 2 4 】

(付記 3) 前記演算処理装置において、

前記緩衝制御部は、前記第 1 および第 2 のバスのうちのバスを介して出力されたデータを前記演算処理部に転送中に、他のバスからデータが出力された場合、前記他のバスを介して出力されたデータを前記バッファに格納することを特徴とする付記 1 記載の演算処理装置。

10

【 0 1 2 5 】

(付記 4) 前記演算処理装置は、前記演算処理部を複数有し、

前記緩衝制御部には、複数の前記演算処理部が接続され、

前記緩衝制御部は、前記第 1 のメモリ部が前記第 1 のバスを介して出力した第 1 のデータと前記第 2 のメモリ部が前記第 2 のバスを介して出力した第 2 のデータを、複数の前記演算処理部に振り分けることを特徴とする付記 1 記載の演算処理装置。

【 0 1 2 6 】

20

(付記 5) 前記演算処理装置において、

前記第 1 のメモリ制御部は、前記演算処理部からの前記第 1 の要求の処理開始から所定期間、前記演算処理部からの後続の要求の処理を停止することを特徴とする付記 1 記載の演算処理装置。

【 0 1 2 7 】

(付記 6) 前記演算処理装置において、

前記第 1 のメモリ制御部は、少なくとも前記緩衝制御部から前記演算処理部への前記第 1 のデータの転送に要する時間を、前記所定期間とすることを特徴とする付記 5 記載の演算処理装置。

【 0 1 2 8 】

30

(付記 7) 前記演算処理装置において、

前記第 1 のメモリ制御部は、前記緩衝制御部から前記演算処理部への 1 回のデータ転送に要する時間にバス数を乗算した時間を、前記所定期間とすることを特徴とする付記 5 記載の演算処理装置。

【 0 1 2 9 】

(付記 8) 前記演算処理装置において、

前記緩衝制御部が有する前記バッファ部は、データ転送 1 回当たりのデータ量に相当する記憶容量であることを特徴とする付記 1 記載の演算処理装置。

【 0 1 3 0 】

40

(付記 9) 前記演算処理装置は、前記演算処理部を複数有し、

前記第 1 および第 2 のバスには、それぞれ複数の前記演算処理部が接続され、

前記第 1 および前記第 2 のメモリ制御部は、複数の前記演算処理部の 1 つから前記要求を受け付けると、前記要求を出力した前記演算処理部と同じバスに接続された前記演算処理部からの前記要求の受け付けを、所定期間だけ停止することを特徴とする付記 1 記載の演算処理装置。

【 0 1 3 1 】

(付記 10) 前記第 1 のメモリ制御部は、前記緩衝制御部から前記演算処理部への 1

回のデータ転送に要する時間を、前記所定期間とすることを特徴とする付記 9 記載の演算処理装置。

【 0 1 3 2 】

50

(付記 1 1) 演算を行う演算処理部を有する演算処理装置のデータ転送方法において

、
前記演算処理装置が有する第 1 のメモリ制御部が、前記演算処理部からの第 1 の要求を入力するステップと、

前記演算処理装置が有する第 2 のメモリ制御部が、前記演算処理部からの第 2 の要求を入力するステップと、

前記第 1 のメモリ制御部に接続された第 1 のメモリ部が、前記第 1 のメモリ制御部の制御により、前記第 1 の要求に応じた第 1 のデータを、第 1 のバスを介して出力するステップと、

前記第 2 のメモリ制御部に接続された第 2 のメモリ部が、前記第 2 のメモリ制御部の制御により、前記第 2 の要求に応じた第 2 のデータを、第 2 のバスを介して出力するステップと、

前記演算処理装置が有する緩衝制御部が、前記第 1 のバスを介して前記第 1 のメモリ部に接続されると共に、前記第 2 のバスを介して前記第 2 のメモリ部に接続され、前記第 1 のバスを介した前記第 1 のデータの出力と前記第 2 のバスを介した前記第 2 のデータの出力が同一期間に行われる場合、前記第 1 のデータを前記演算処理部に転送し、前記第 2 のデータを前記緩衝バッファ部に格納し、前記第 1 のデータの転送の終了後、前記第 2 のデータを緩衝バッファ部から前記演算処理部に転送するステップと、

を有することを特徴とするデータ転送方法。

【0133】

(付記 1 2) 前記データ転送方法において、

前記緩衝制御部は、前記第 1 および第 2 のメモリ部から同時にデータが出力された場合、前記第 1 および第 2 のメモリ部のうち、データ転送時間が大きい一のメモリ部からのデータを前記第 1 のデータとして前記演算処理部に転送し、他のメモリ部からのデータを前記第 2 のデータとして前記緩衝バッファ部に格納することを特徴とする付記 1 1 記載のデータ転送方法。

【0134】

(付記 1 3) 前記データ転送方法において、

前記緩衝制御部は、前記第 1 および第 2 のバスのうちのバスを介して出力されたデータを前記演算処理部に転送中に、他のバスからデータが出力された場合、前記他のバスを介して出力されたデータを前記バッファに格納することを特徴とする請求項 1 1 記載のデータ転送方法。

【0135】

(付記 1 4) 前記データ転送方法において、

前記演算処理装置は、前記演算処理部を複数有し、

前記緩衝制御部には、複数の前記演算処理部が接続され、

前記緩衝制御部は、前記第 1 のメモリ部が前記第 1 のバスを介して出力した第 1 のデータと前記第 2 のメモリ部が前記第 2 のバスを介して出力した第 2 のデータを、複数の前記演算処理部に振り分けることを特徴とする付記 1 1 記載のデータ転送方法。

【0136】

(付記 1 5) 前記データ転送方法において、

前記第 1 のメモリ制御部は、前記演算処理部からの前記第 1 の要求の処理開始から所定期間、前記演算処理部からの後続の要求の処理を停止することを特徴とする付記 1 1 記載のデータ転送方法。

【0137】

(付記 1 6) 前記データ転送方法において、

前記第 1 のメモリ制御部は、少なくとも前記緩衝制御部から前記演算処理部への前記第 1 のデータの転送に要する時間を、前記所定期間とすることを特徴とする請求項 1 5 記載のデータ転送方法。

【符号の説明】

10

20

30

40

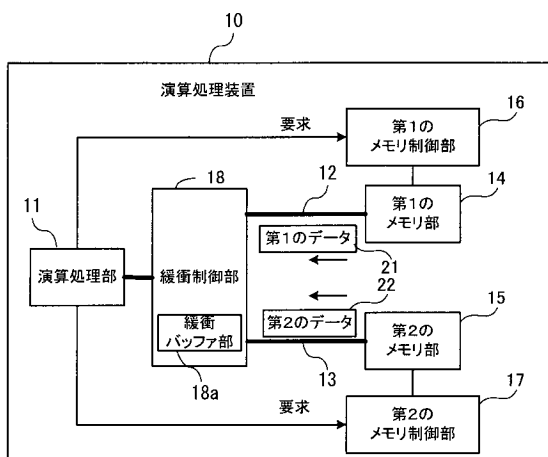
50

【 0 1 3 8 】

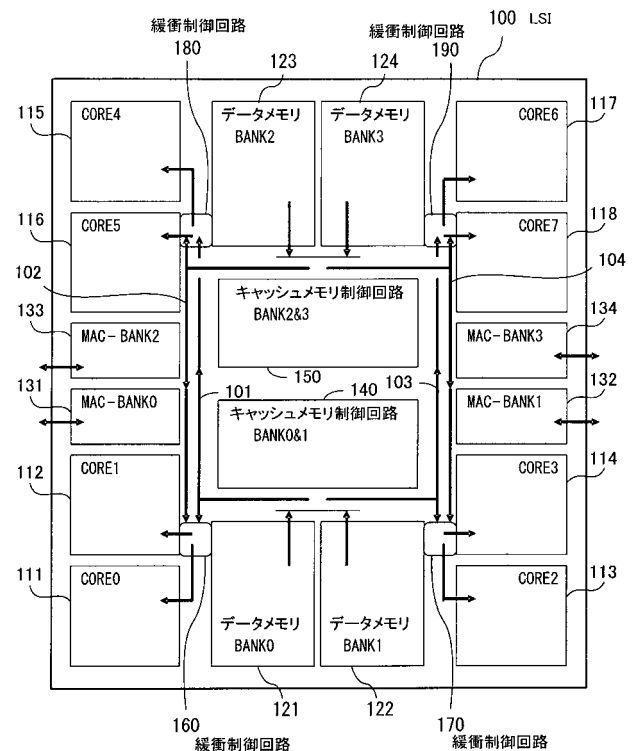
- 1 0 演算処理装置
- 1 1 演算処理部
- 1 2 第 1 のバス
- 1 3 第 2 のバス
- 1 4 第 1 のメモリ部
- 1 5 第 2 のメモリ部
- 1 6 第 1 のメモリ制御部
- 1 7 第 2 のメモリ制御部
- 1 8 緩衝制御部
- 1 8 a 緩衝バッファ部
- 2 1 第 1 のデータ
- 2 2 第 2 のデータ

10

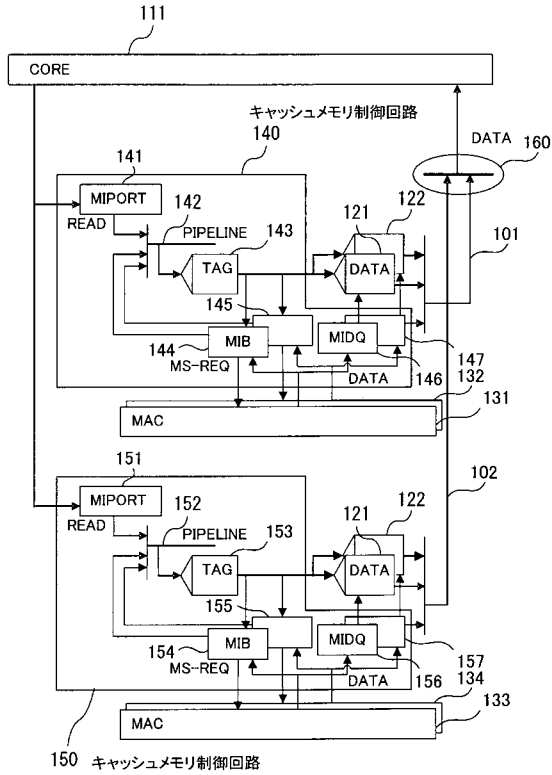
【 図 1 】



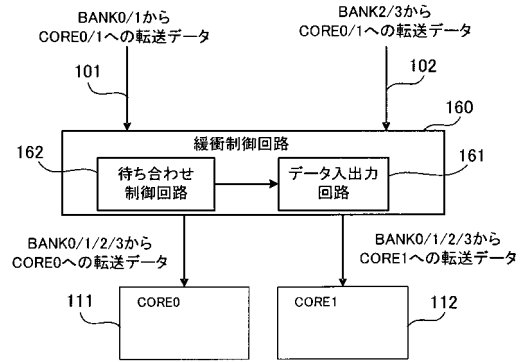
【 図 2 】



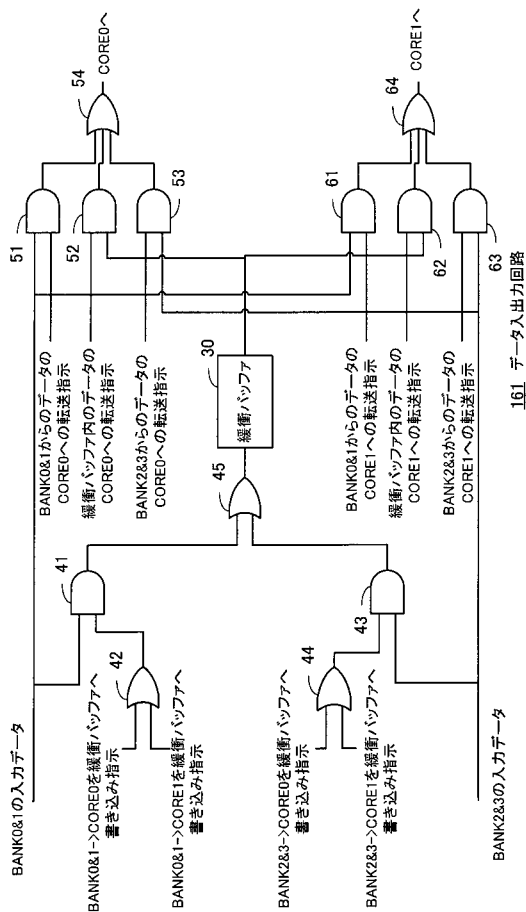
【図 3】



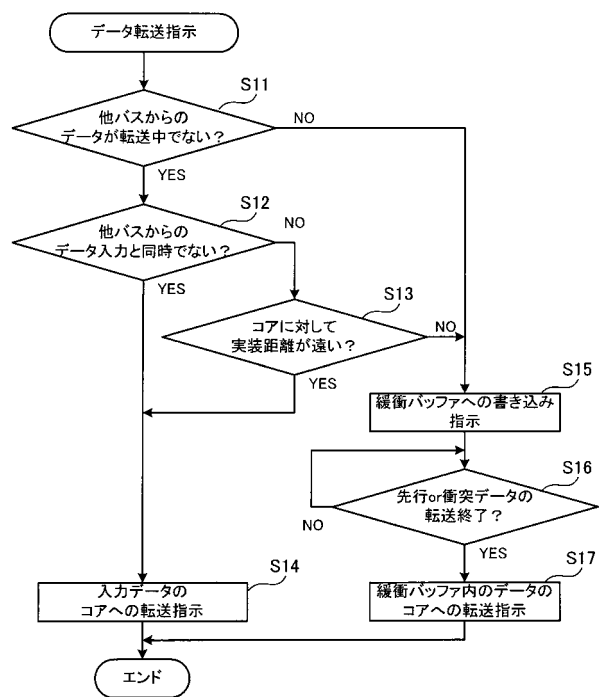
【図 4】



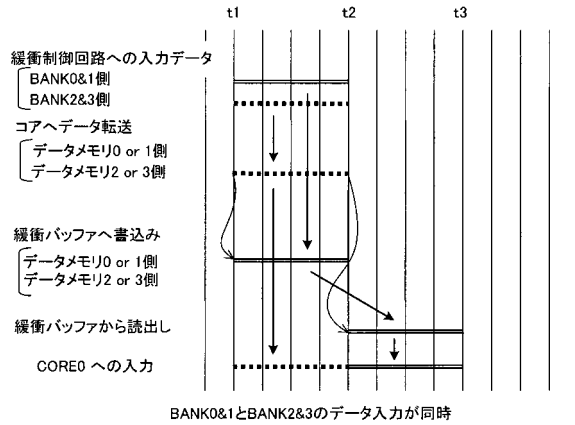
【図 5】



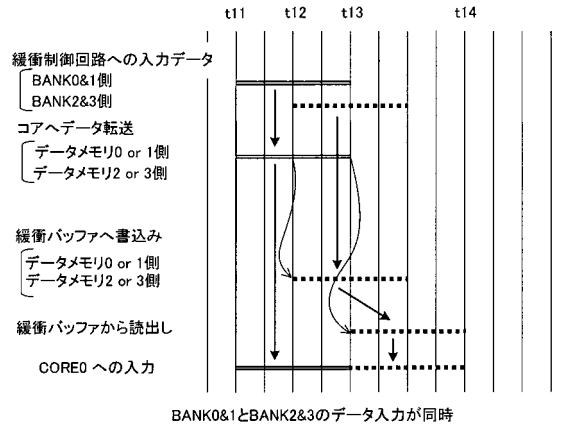
【図 6】



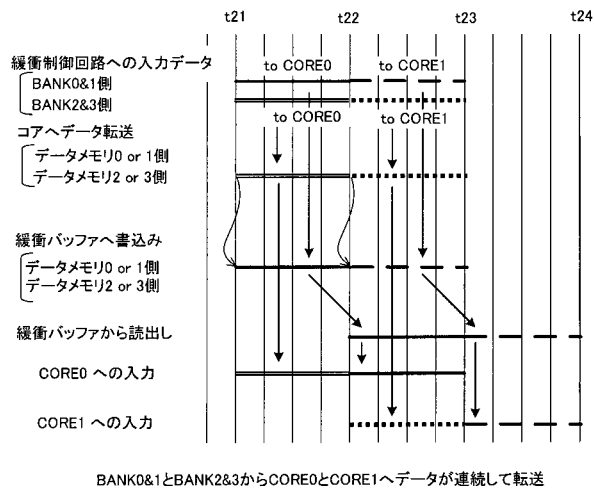
【図 7】



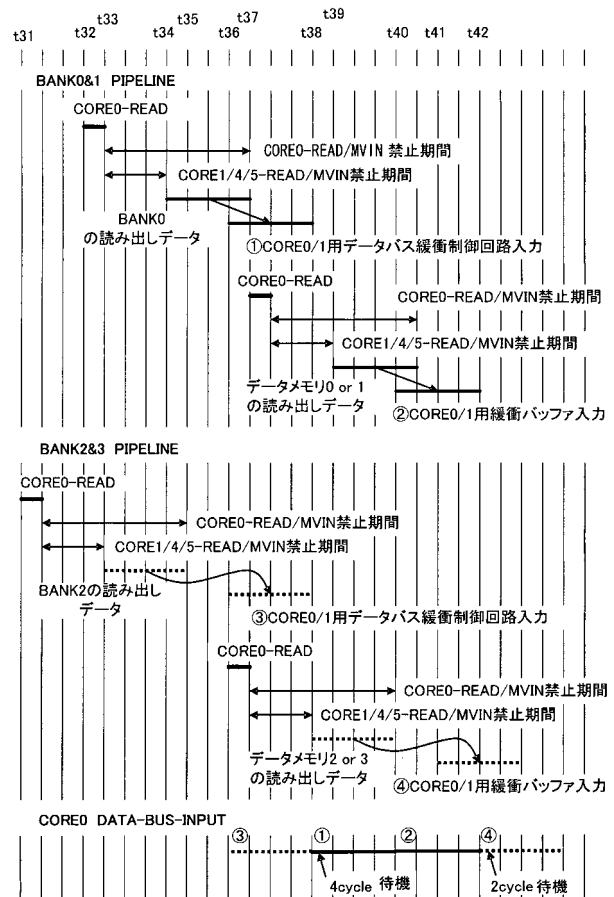
【図 8】



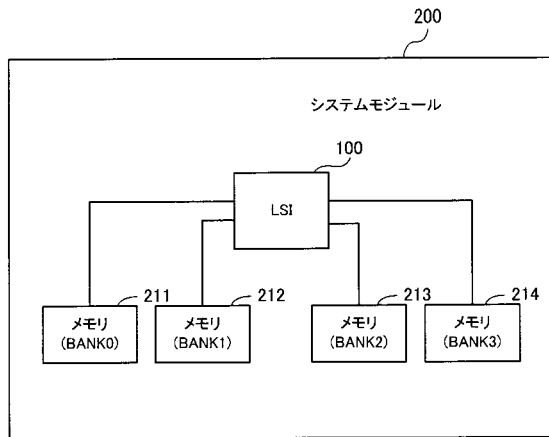
【図 9】



【図 10】



【図 11】



フロントページの続き

(51) Int.Cl.

F I

テーマコード (参考)

G 0 6 F	12/00	5 7 1 B
G 0 6 F	12/06	5 2 5 A
G 0 6 F	12/00	5 5 0 K
G 0 6 F	12/08	5 5 3 D