(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3546582号
(P3546582)
(45)発行日　平成16年7月28日(2004.7.28)　(24)登録日　平成16年4月23日(2004.4.23)

(51)Int.Cl.[7]　F I
H01L 21/822　H01L 27/04　A
H01L 21/82　H01L 21/82　S
H01L 27/04

請求項の数 7　(全 17 頁)

(21)出願番号　特願平8-51321
(22)出願日　平成8年3月8日(1996.3.8)
(65)公開番号　特開平9-331019
(43)公開日　平成9年12月22日(1997.12.22)
審査請求日　平成13年10月3日(2001.10.3)

(73)特許権者　503121103
株式会社ルネサステクノロジ
東京都千代田区丸の内二丁目４番１号
(74)代理人　100081938
弁理士　徳若　光政
(74)代理人　100085811
弁理士　大日方　富雄
(74)代理人　100075096
弁理士　作田　康夫
(72)発明者　渡部　隆夫
東京都国分寺市東恋ケ窪１丁目２８０番地
株式会社日立製作所中央研究所内

最終頁に続く

(54)【発明の名称】半導体装置

(57)【特許請求の範囲】
【請求項１】
複数の第１I/0線を持つメモリコアと、
複数の第２I/0線を持つ論理回路と、
前記複数の第１I/0線と前記複数の第２I/0線との間のデータ転送を行うデータ転送回路とを具備し、
前記データ転送回路は、上記複数の第１I/0線と同一のピッチを有し、前記複数の第１I/0線と接続するための複数の第３I/0線と、前記論理回路とのデータの授受を行う複数の第４I/0線と、前記複数の第３I/0線と前記複数の第４I/0線との間に接続された複数のスイッチとを有し、
前記複数のスイッチのオンオフ状態を前記データ転送回路に供給されるスイッチ制御信号で切り替えることにより、前記複数の第３I/0線と前記複数の第４I/0線との間のデータ転送パターンを実時間で可変とすることが可能な半導体集積回路装置。
【請求項２】
請求項１において、
前記スイッチ制御信号の配線は前記複数の第１I/0線又は前記複数の第２I/0線の配線と直交して配置されることを特徴とする半導体集積回路装置。
【請求項３】
請求項１または請求項２において、
前記複数のスイッチは、前記複数の第１I/0線と平行な方向に前記複数の第１I/0線と同一

のピッチで配置されたものが、前記複数の第１I/0線と直交する方向に並んで配置されていることを特徴とする半導体集積回路装置。
【請求項４】
請求項１乃至請求項３のいずれかに記載の半導体集積回路装置において、
前記論理回路は基本論理ゲートを組み合わせることによって合成され、
前記複数のスイッチはＭＯＳトランジスタで形成され、形状は共通であることを特徴とする半導体集積回路装置。
【請求項５】
請求項１乃至請求項４のいずれかに記載の半導体集積回路装置において、
前記データ転送パターンの少なくとも一つは複数の第４I/0線のうちの一つに対し、前記複数の第２I/0線の複数と接続されることを特徴とする半導体集積回路装置。
【請求項６】
請求項１から５の何れか１つにおいて、
前記メモリコアが１つのトランジスタと１つのキャパシタからなるＤＲＡＭタイプのセルを含むことを特徴とする半導体集積回路装置。
【請求項７】
複数の第１I/0線を持つメモリコアと、論理ライブラリと、転送回路用モジュールのライブラリを作成する工程と、
前記論理ライブラリから所望の論理回路を合成する工程と、
前記論理回路と前記転送回路との間の配線と、前記複数の第１I/0線と前記転送回路用モジュールとの間の配線を設定する工程とを有する半導体集積回路装置の設計方法であって、
前記転送回路用モジュールは、上記複数の第１I/0線と同一のピッチを有し、上記複数の第１I/0線と接続するための複数の第２I/0線と、前記論理回路とのデータの授受を行うための複数の３I/0線と、前記複数の第２I/0線と前記複数の第３I/0線とを接続する複数のスイッチとを有し、
前記配線を設定する工程において、前記複数のスイッチのオンオフ状態を切り替えるスイッチ制御信号を前記転送回路用モジュールに供給する配線を設定することにより、前記複数の第２I/0線と前記複数の第３I/0線との間のデータ転送パターンを設定する半導体集積回路装置の設計方法。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】
本発明は、メモリを集積した半導体装置に係わり、特に複数のI/0線をもつメモリと論理回路とを同一の半導体チップ上に集積した半導体装置において、種々の目的に応じたLSIチップを短期間で設計するための方式およびそれによる製品群を与え、また上記メモリ・論理回路間のデータの転送パターンを高速に変化させることのできる高集積なデータ転送回路の方式を与えるものである。
【０００２】
【従来の技術】
多数の演算器、プロセッサおよびメモリを相互結合する並列計算システムを同一の半導体チップ上に集積する試みとして、米国特許5371896をあげることができる。この従来例では、複数のメモリと複数の演算回路を同一の半導体チップ上に集積され、両者の間がクロスバスイッチからなるネットワークで結合される。この従来例は、必要に応じてSIMD(Single Instruction Multi Data Stream)動作とMIMD(Multi Instruction Multi Data Stream)動作切り換えて行うことができることが特長である。SIMD動作時には、複数のメモリのうち１つがインストラクションメモリとして使われ、残りのメモリがデータメモリとして使われる。演算回路には、インストラクションメモリからの命令が共通に与えられる。MIMD動作時には、SIMD動作時にデータメモリとして使われたメモリの一部がインストラクションメモリとして使われることにより、個々の演算回路に、別々のインストラクションメ

モリからの命令が与えられる。個々のメモリと演算回路との間のデータ転送経路は、クロスバネットワークにより様々に切り換えることができる。

【０００３】

【発明が解決しようとする課題】

メモリを集積した半導体装置は上記のほかにも種々考案されているが、最近特にDRAM(Dynamic Random Access Memory)などからなる比較的高集積のメモリと論理回路とを同一の半導体チップに集積したものが注目を集めている。このようなLSIは、一般にユーザの要求を受けて半導体メーカがそれを作り始めるため、チップの完成までの時間(Time to Customers)の短縮が必要である。しかし、一方、必要となるメモリ容量や演算回路の種類は用途によって異なる。この相反する要求を満たすには、設計方式から改革する必要がある。しかし、従来の高集積メモリ、特にDRAMでは、仕様が標準化されているため、そのままの設計方式では、上記の要求に対応することができなかった。

【０００４】

さらに、DRAMなどの高集積メモリと論理回路とを同一の半導体チップに集積する場合には、それを単に集積しただけでは個別チップに対して大きなメリットが生じにくい。コストと要求性能を考慮すると、1cm角程度の半導体チップ上に大容量のメモリと大規模な論理回路や演算回路を集積し、両者の間の結合線の本数を数百本以上とし、かつてデータ転送速度を１GigaByte/sec以上にする必要がある。したがって、メモリと論理回路とを結合する結合回路として、高速かつ高集積なものが必要である。しかし、上記従来例のようにクロスバスイッチを用いた場合、結合線の数が増加するとスイッチの個数が膨大となりハードウエアの規模が増大し、遅延も増大してしまう。上記の従来技術のように独立した複数のメモリと複数の演算回路との間のデータ転送経路を切り換える場合には一般にメモリや演算回路の数も少ないので従来の並列計算機で使われていた方式をそのまま同じチップ上に実現するのも可能である。しかしながら数百本以上ものメモリのI/O線群と論理回路や演算回路とのI/O線群の間の対応を切り換える場合には、集積度と動作速度の要求がきびしく、従来の方式をそのまま利用するのは困難である。

【０００５】

【課題を解決するための手段】

上記の第一の課題を解決するために、本発明では多くのI/O線をもち、容量の異なるメモリコアとメモリコアのI/O線のピッチに合わせて設計した結合回路用のモジュールのレイアウトパターンをあらかじめ作ってデータベースに記憶させる。さらに、論理回路を合成するための論理ライブラリも作成し、データベースに記憶させる。データベースには、それらのレイアウトパターンや仕様、特性など設計に必要なデータを記憶させる。

【０００６】

結合回路用のモジュールは、スイッチ群とバッファ群とからなり、組合せて結合回路を構成できる。スイッチ群は、入力されたデータをその中でその順番を入れ替えることができるものである。複数のスイッチ群を接続して、転送パターンに合わせて所望の転送パターンに対応するスイッチ群を活性化することにより、高速に転送パターンを切り替えることができる。これらのモジュールは、メモリコアのI/O線のピッチに合わせて作られることとされ、レイアウトパターンを変更することなくメモリコアのI/O線にそのまま結合できる。

【０００７】

上記のように本発明によれば、メモリコア、結合回路用モジュール、論理ライブラリのレイアウトパターンがデータベースにあらかじめ登録されており、なおかつメモリコアと結合回路用モジュールとの配線ピッチがそろえられることとされ、そのまま結合して使うことができる。したがって、ユーザからLSIチップの仕様が与えられてからの設計を短期間に終わらせることができる。すなわち、必要な容量のメモリコアと仕様に合った転送回路を作るためのモジュールとをデータベースから取り出して組合せ、さらに論理部分は、論理合成用のCADツールを用いて論理ライブラリから所望の論理回路を合成すればよい。それらの間の配線は、配置配線CADツールにより高速にできる。したがって、メモリと論理

回路とを集積したチップが短期間にできる。
【０００８】
さらに、上記の結合回路では、メモリと論理回路とで転送されるデータが通過するのは活性化されたスイッチ群のみであるために高速なデータ転送が実現できる。さらに、転送パターン数に合わせて段数を増減するため転送パターンが少ない場合には無駄な占有面積がない。
【０００９】
【発明の実施の形態】
[メモリコアを用いたシステムLSIの設計方法]
図１には、本発明に係るメモリコアを内蔵したシステムLSIの概念をが示されている。図１を用いて本発明に係るLSIの設計方法を説明する。
【００１０】
図１の左に示されるのは、コア回路、論理ライブラリのレイアウトパターンや特性を登録したデータベース用記憶装置DBである。ここには、多くのI/O線をもち容量の異なる複数のメモリコアMRと、メモリコアのI/O線のピッチに合わせて設計した転送回路（結合回路）TG用のモジュール群と、論理回路を合成するための基本ゲートからなる論理ライブラリLLとのレイアウトパターンや仕様、特性など設計に必要なデータをあらかじめ記憶する。
【００１１】
ここで、転送回路TG用のモジュールは、スイッチ群SWGとバッファ群TGBUFi
からなり、組合せて転送回路TGを合成できる。詳しくは後述するが、複数のスイッチ群を接続することにより様々な転送パターンを持つ転送回路TGを合成することができる。これらのモジュールは、メモリコアMRのI/O線のピッチに合わせて作られているので、レイアウトパターンを変更することなくメモリコアMRのI/O線にそのまま結合できる。
【００１２】
LSIチップの仕様が与えられると上記データベースDBから必要なデータを設計用ワークステーションWSに転送しながら設計を行なう。メモリコアMRと転送回路TG用モジュールの配線ピッチがそろっているため、こららはそのまま結合して使うことができる。すなわち、必要な容量のメモリコアMRと仕様に合った転送回路TGを作るためのモジュールをデータベースDBから取り出して組合せればよい。論理部分は、論理合成用のCADツールを用いることにより、論理ライブラリLLから所望の論理回路LCを容易に合成できる。最後にチップのフロアプランに合わせてそれらを配置し、その間の配線を、配置配線CADツールにより行なえばチップのレイアウトデータが完成する。このようにして、メモリコアMRを内蔵したシステムLSIの製品群を短期間に設計できる。
【００１３】
なお、ここでは論理ライブラリLLを用いて論理を合成する例を示したが、場合によってはチップの一部をゲートアレイにして論理を合成してもよい。その場合は、メモリコアMRが共通で論理が異なるチップを容易に製造できるという利点がある。
【００１４】
図１の右下に上記のようにして設計したチップの例が２つ示される。半導体チップLSI-Aは、メモリコアMRと論理回路LCを転送回路TGで結合したブロックA,B,C,Dを４つ並べて、その中心にチップ全体を制御する制御回路CCを配置したものである。半導体チップLSI-Bは、メモリコアMRと論理回路LCを転送回路TGで結合したブロックA,Bを２つ並べて、中心にチップ全体を制御する制御回路CCを配置したものである。
【００１５】
本発明では、もちろん一つのメモリコアMRを用いるチップも実現できるが、本例のように複数のブロックを集積するチップも容易に設計できる。その場合、各ブロックのメモリコアMR、論理回路LCを異なるものとしてもよいし、同一の構成にしてもよい。前者は、異なる処理を同一のチップで並列に行なうものに適しており、後者は、同一の処理を並列に行なうものに適している。特に後者は、グラフィックス、自然画像処理、ニューラルネットワークなど並列動作が可能な処理を行なうものに適している。

【００１６】
半導体チップLSI-A,LSI-Bのどちらもメモリコアメモリ(MR)とデータの授受を行なう論理回路(LC)をメモリコアメモリ(MR)に近接しているため配線遅延の影響が少なく高速のデータ転送が実現できる。また、制御回路(CC)から各ブロックへの距離が半導体チップLSI-Bでは等しく、半導体チップLSI-Aでも差が少ないので、制御信号のスキューが小さくできるという利点がある。
【００１７】
半導体チップLSI-Bでは、論理回路(LC)を制御回路(CC)に近接したが、メモリコアメモリ(MR)の制御信号の配線を短くして配線遅延を少なくする必要がある場合には、ブロックを制御回路(CC)に対して反転させメモリコアメモリ(MR)を制御回路(CC)に近接してもよい。
なお、半導体チップLSI-Aにおいて制御回路(CC)からの距離がブロックAとBおよびDとCで異なることが問題となる場合も考えられる。その場合には、半導体チップLSI-Bのような配置を行なって制御回路(CC)の左右にブロックを２つずつ配置すればよい。
【００１８】
ブロックの形状が横に長い場合には、そのようにするとチップの短辺と長辺の差が大きくなりすぎる場合がある。そのような場合には、図１に示される半導体チップLSI-Aの配置のまま、制御信号の入力端子をブロックの片側の面に集中させ、ブロックAとBおよびDとCを反転して配置することにより、ブロック同士が隣接する面に制御信号の入力端子が来るようにできる。これにより制御信号のスキューを減少することができる。以下では、図１に示される転送回路(TG)について詳しく説明する。
【００１９】
[多重I/Oメモリコア内蔵LSI]
図２には、本発明に係る多重I/Oのメモリ内蔵LSIの例が示される。図２に示される半導体チップSICは、複数のI/O線MIOiをもつメモリコアメモリ(MR)と、複数のI/O線LIOiをもつ論理回路(LC)と、メモリコアメモリ(MR)と論理回路(LC)の間のデータの転送パターンを制御する転送回路(TG)等とを単一の単結晶シリコン等からなる半導体基板に集積したものである。
【００２０】
論理回路(LC)の内容は論理ライブラリLLを用いて目的に応じたものを合成すればよい。ここでは、画像あるいはグラフィックスに適する一例が示される。メモリコアメモリ(MR)に記憶された画素に対して演算を行なう演算器群ARWとメモリコアメモリ(MR)の内容を画面に表示するために一定の速度で読み出すための表示用バッファDBRならびにそれらとメモリコアメモリ(MR)を制御するための制御回路(LCC)から構成されている。
【００２１】
メモリコアメモリ(MR)は、複数のデータ線DLと複数のワード線WLならびにそれらの交点に形成されたメモリセルMCを有する。メモリセルMCは、１トランジスタ・１キャパシタのDRAMセル，４又は６トランジスタのSRAM(Static Random Access Memory)セル,１トランジスタの不揮発性のフラッシュメモリセルなどを用いることができる。なお、以下では書き込みと読み出しのできるいわゆるRAM型を仮定するが読みだし専用のいわゆるROM型のものを使用する場合にも本発明は有効である。メモリコアメモリ(MR)へのデータの書込み読み出しは、読み出し書込み回路RWCにより制御され、周辺回路PERによって選択された、複数のメモリセルMCに複数のI/O線から並列にデータを読み書きすることができる。周辺回路PERには、論理回路(LC)からのメモリコア制御信号MRC、制御信号CTL及びアドレス信号DATA等のバスが接続されている。メモリコアメモリ(MR)は、論理回路(LC)の基準信号であるクロック信号と同期して制御信号、アドレス信号、I/O信号を入力したり、出力したりする。
【００２２】
論理回路(LC)は、メモリコアメモリ(MR)から転送回路(TG)を通じて読み出されるデータや半導体チップSIC外部からのデータに対して演算を行う。その結果を再び転送回路(TG)を通じてメモリコアメモリ(MR)へ書き込んだり、半導体チップSIC外部へ出力する。
【００２３】
転送回路(TG)は、多段に接続されたスイッチ群SWGより構成され、制御信号TGCiによってメ

モリコアMRの複数のI/O線MIOiと論理回路LCの複数のI/O線LIOiとの間の接続関係（以下、転送パターンという。）を切り換えることができる。

【００２４】

図３には、転送パターンの例としてP0からP7まで８つのパターンを実現する場合が示される。この例は、2のn乗本のI/O線MIOiとLIOiに対して、その1/4(２の(n－2)乗)を単位としたMIO0,1,2,3とLIO0,1,2,3の対応を切り換えるものである。転送単位が2のn乗本である必要はなく、また全ての転送単位が等しくなくとも本発明を適用できることはもちろんである。矢印の向きはデータの流れを示しており転送パターンP1はメモリへの書込みのみに使用し、残りのパターン(P0,P2～P7)は読み出し、書込み両方に使う。

【００２５】

転送パターンP0は、そのままデータの入れ替えなしに転送するパターンである。転送パターンP1は、(LIO0,1)に入力するデータを(MIO0,1), (MIO2,3)に伝達してメモリへ書き込むためのものである。この例は他のパターンとは異なり異なるメモリのI/O線が導通する。このため、読出し時には異なるデータが衝突する場合があるので書込み時にのみ使用する。このパターンは後述するようにメモリの内容を高速に初期化するなどに有効である。

【００２６】

転送パターンP2およびP3はそれぞれ(LIO0,1)と(MIO0,1)、(LIO0,1)と (MIO2,3)との間に転送経路を形成するものである。転送パターンP4からP7はそれぞれ(LIO1)と(MIO0)、(LIO1)と(MIO1)、(LIO1)と(MIO2)、(LIO1)と (MIO3)との間に転送経路を形成するものである。

【００２７】

８つの転送パターン(P0～P7)は、制御信号TGCiにより自由に切り換えることができる。それぞれの転送パターンは、転送回路TG内のひとつのスイッチ群SWGをオンすることにより実現できる。たとえば、転送パターンP0は、図２に示されるスイッチ群SWG#0をオンすることにより実現できる。転送回路TGの具体的な構成は後述する。

【００２８】

本実施例では、メモリコアMR、転送回路TG、論理回路LCを同一の半導体チップ上に形成するので数十本から数百本のI/O線を容易に配線することが可能である。

【００２９】

以下、図２に示される多重I/Oメモリコア内蔵LSIの動作を説明する。

【００３０】

まず、読出し動作を説明する。メモリコアMR内の周辺回路PERにより一本のワード線WLを選択すると、そのワード線WL上のメモリセルMC群からデータ線DLにデータが読み出され、読出し書込み回路RWCを通じて複数のI/O線MIOiに並列にデータが読み出される。制御信号TGCiにより、転送回路TG内のスイッチ群SWGのうち一つが活性化されるとメモリコアMRの複数のI/O線MIOiと論理回路LCの複数のI/O線LIOiの間の転送パターンが確定しI/O線MIOiからI/O線LIOiにデータが転送され、論理回路LCに入力される。

【００３１】

書込み動作もデータの流れが逆になる以外は同様である。すなわち、論理回路LCから複数のI/O線LIOiに出力されたデータは、制御信号TGCiにより確定した転送パターンにしたがってI/O線LIOiからI/O線MIOiに転送され、読出し書込み回路を通じてデータ線DLに伝達され、さらに選択されたワード線WL上のメモリセルMCに並列に書き込まれる。

【００３２】

読出しあるいは書込みを連続に行ったり、交互に行ったりする際には、サイクル毎に選択択するワード線WLや転送パターンを切り換えて動作させることができる。したがって、論理回路LCの要求に応じてサイクル毎に異なるアドレスに対応するメモリセルMCに並列に読出し書込みを行うことができる。

【００３３】

本実施例によれば、メモリコアMRと論理回路LCの間のデータの授受は、一段のスイッチ群SWGを通じて行われるため、非常に高速なデータ転送が実現できる。また、I/O線MIOiとLI

Oiが同一方向に走るようにメモリコアMRと論理回路LCを配置するため、メモリコアMRと論理回路LCの間に転送回路TGを配置することができる。転送回路TGのスイッチ群SWGの段数は転送パターンに応じて決まるため転送パターン数が少ない場合には転送回路のデータ線方向の寸法（図２の横方向）を小さくすることが可能である。したがって、図２に示されるようにメモリコアMRのワード線WL方向の寸法（図２の縦方向）に収まるように転送回路TGと論理回路LCをレイアウトすると余分な面積をとることなく全体の面積を小さくすることができる。

【００３４】

なお、周辺回路PERは、上記のようにワード線WLを選択するXデコーダのみを含んでもよいし、データ線の一部を選んでI/O線に接続するYデコーダを含んでもよい。本実施例によればI/O線は多数設けることができるので、通常、Yデコーダは、例えば1024本のデータ線のうち128本を選ぶような簡単なものでよい。

【００３５】

[転送回路の第１の具体例]

次に転送回路TGの具体的な回路例を図４を使って説明する。図４には、図２に示される転送パターンを実現する転送回路TGの回路例実施が示される。

【００３６】

図４において、MIO0，MIO1，MIO2，MIO3はメモリコアMRのI/O線であり、LIO0,LIO1，LIO2，LIO3は論理回路LCのI/O線である。また、SWG0，SWG1，..，SWG7はスイッチ群であり、TGBUF0，TGBUF1，TGBUF2，TGBUF3はバッファ回路である。

【００３７】

TGC0，TGC1,..，TGC7はそれぞれスイッチ群SWG0，SWG1，..，SWG7をオンオフする信号である。スイッチ群SWGのスイッチSWをどの種類のトランジスタで構成するかに依存するが、ここでは制御信号TGCiを高電位とするとその制御信号TGCiが印加されているスイッチSWがオンし、低電位とするとオフするとする。たとえば制御信号TGC3を高電位とし、他の制御信号を低電位とするとスイッチ群SWG3内の矢印で示した２つのスイッチSWがオンする。その結果P3の転送パターンが形成され、メモリコアのI/O線MIO2，MIO3と論理回路LCのI/O線LIO0，LIO1との間に転送経路ができる。その他の転送パターンも同様にして制御信号TGCiのうちの一つを高電位とすれば実現できる。

【００３８】

バッファ回路TGBUF0，TGBUF1，TGBUF2，TGBUF3はI/O線の付加容量の影響で
信号が遅延するのを避けるためにつけたものである。この回路の構成例は図５に示される。図５を用いてバッファ回路TGBUFiの動作を説明する。

【００３９】

バッファ回路TGBUFiは、メモリコアMRの読み出し書き込み動作に合わせてデータの流れを切り替える両方向のバッファであるとともに、転送パターンが形成されたときに使用しない論理回路LCのI/O線LIOiの電位をラッチする働きを持つ回路である。

【００４０】

図３に示される例では転送パターンP0を除いていずれも論理回路LCのI/O線LIOiの一部は使用されない。使用されないI/O線の電位が確定しないでいわゆるフローティング状態になると電荷のリークにより中間電位となる可能性がある。その場合、論理回路LC側でCMOS(Cmplement Metal Oxide Semiconductor)トランジスタのゲートに入力されていると過剰な電流が定常的に流れてしまう。それを避けるために論理回路LCのI/O線LIOiのうち使用しないものは電位をラッチする。

【００４１】

論理回路LCのバッファ回路TGBUFiのイネーブル信号LIOEiを低電位とすると図４に示される論理から明らかなように信号TGWi，TGRiが低レベル、信号TGWBi，TGRBiが高レベルとなりクロックドインバータ回路RINV,WINVがオフする。さらに信号LIOPRiがゲートに入力されているMOSトランジスタQ1がオンしてI/O信号LIOiは低レベルにラッチされる。使用するI/O信号LIOiについては、イネーブル信号LIOEiを高電位とする。データ方向の切り替えは

以下のように行う。

【００４２】

メモリコアMRが読み出し動作のときには、信号TGRWを低電位とする。するとイネーブル信号LIOEiが高電位であるときには、読み出し用クロックドインバータRINVのみが活性化されI/O線LIOi'からI/O線LIOiにデータが転送される。一方、メモリコアMRが書き込み動作のときには、信号TGRWを高電位とする。するとイネーブル信号LIOEiが高電位であるときには、書き込み用クロックドインバータWINVのみが活性化されI/O線LIOiからI/O線LIOi'にデータが転送されスイッチSWを通じてメモリコアMRのI/O線MIOiにデータが転送される。

【００４３】

以上説明したように図４、図５に示される実施例を用いれば、転送されるデータが通過するスイッチSWの段数は一段なので高速な動作が実現できる。また、スイッチSWの段数は転送パターン数に等しいので、無駄なレイアウト領域が不要で高集積化が可能である。さらに、論理回路LCのI/O線LIOiのうち使用しないI/O線のバッファ回路TGBUFiを停止し、さらに電位がフローティング状態になることを避けらるので無駄な電力消費がなく論理回路LCのゲートに過剰な電流が流れることを防止できる。したがってI/O線の一部を使用しない転送パターンを自由に設定できる。

【００４４】

なお、図４では、スイッチ群SWG内のスイッチSWのうち制御信号TGCiが入力されない不必要なものも設けてある。これは以下の理由による。図４からわかるように転送回路TGのスイッチ群SWGはスイッチSWの制御信号TGCiとの接続、I/O線MIOiとの接続に必要な配線、コンタクト以外は転送パターンによらず共通の形状をしている。したがって、スイッチのTGCiとの接続、MIOiとの接続に必要な配線、コンタクトをのぞく共通部分をレイアウトライブラリとして用意しておけば、チップのレイアウト設計が容易となるためである。さらに、万一転送パターンを修正する場合にもスイッチ群SWG内のスイッチSWをすべて作っておけばスイッチSWを構成するトランジスタ部のマスクの修正は不要なため修正するマスクの枚数を削減できる。本発明のようなメモリ、論理混載のチップは用途によってメモリ容量や論理の構成を変える必用がある。したがって、何種類かのメモリコアMRと転送回路TG用の上記スイッチSWG群の基本パターンをライブラリとして用意しておけば、それらから必要なものを選んで、さらに論理部分を論理用基本ライブラリLLを用いて合成して配置配線を行うことによりLSIチップのマスクを迅速に設計することができる。

【００４５】

[転送回路の第２の具体例]

もちろんI/O線に接続されるスイッチSWの数が増えると付加容量の増大による遅延が問題になる場合もある。したがって、スイッチ群SWGの段数が非常に多い場合には不要なスイッチSWは省略してもよい。

【００４６】

図６には、図２の転送回路TGを図４に示されるより少ない７段のスイッチ群SWGで実現する転送回路TGの第２の具体例が示される。図４では、一つの転送パターンに一つのスイッチ群SWGが対応していた。しかし、図３の転送パターンP0，P1，P2には、メモリコアMRのI/O線のうちMIO0，MIO1と、論理回路LCのI/O線LIO0,LIO1を接続する共通点がある。また、、転送パターン P1とP3には、メモリコアMRのI/O線のうちMIO2，MIO3と、論理回路LCのI/O線LIO0，LIO1を接続する共通点がある。これに着目してスイッチ群SWGＯを削除してスイッチ群SWG1とSWG2を変更したのが図６の実施例である。

【００４７】

図７の下部には、各転送パターン(P0～P7)を実現するための制御信号TGCi、TGRW、LIOEiの設定法が示される。ここで"1"は高電位、"0"は低電位を示す。なお、転送パターンP1は前述の理由から書き込み動作しかできないので制御信号TGRWは"1"にしか設定できない。転送パターンP0，P1を実現するための制御信号TGCiの設定が図４の実施例と異なる。

【００４８】

転送パターンP0を実現するためには、制御信号TGC1とTGC2の二つの制御信号を高電位にすればよい。制御信号TGC1により、I/O線のうちMIO2とLIO2、MIO3とLIO3が接続され、制御信号TGC2により、I/O線のうちMIO0とLIO0，LIO1とMIO1が接続される。

【００４９】

転送パターンP1を実現するためには、TGC2とTGC3の二つの制御信号を高電位にすればよい。制御信号TGC2により、I/O線のうちMIO0とLIO0，LIO1とMIO1が接続され、制御信号TGC3により、I/O線のうちMIO2とLIO0、MIO3とLIO1が接続される。本実施例では、このようにスイッチ群SWGの段数を削減できる。ここで２つのスイッチ群SWGを活性化して転送パターンP0とP1を実現しているが、データが通過するのはスイッチSW一段であるところが第２の特長である。この点は、従来のオメガネットワークなど複数の段数をデータが通過するものと異なる。以上のように本実施例によれば、高速性を損なうことなくより高速化を達成できる。

【００５０】

[転送回路の第３の具体例]

図８には、スイッチSWを並列に接続することにより、図６の実施例よりさらにスイッチ群SWGの段数を削減した例が示される。この例では、スイッチ群SWGを３段に削減できる。制御信号の設定方法は図７に示される実施例と同じである。図８に示される例では、各スイッチ群SWGにおいてI/O線LIOi'の両側にスイッチSWが配置される。スイッチSWの回路構成とレイアウトの例を図９に示される。各スイッチSWはnMOS(nチャネルMOS)とpMOS(pチャネルMOS)より構成し、nMOSのゲートには制御信号TGCi，TGCjをpMOSのゲートにはその逆相信号TGCiB，TGCjBを入力する。

【００５１】

図９右側には、スイッチ部のレイアウト例が示される。ここではnMOSのみを示した。M2，M1，CONT1，CONT2，Lはそれぞれ第２配線層、第１配線層， M1とLとのコンタクト、第１配線層と第２配線層とのコンタクト、拡散層である。本実施例では２つのスイッチSWを構成するMOS同士の拡散層LをI/O線LIOi'のところで共通化できるので狭いI/O線のピッチに収めることができる。なお、ここでは並列に接続するスイッチSWの数を２つとしたが、I/O線のピッチにが広い場合には３つ以上のスイッチSWを並列に接続してさらに段数の削減を行ってもよいのはもちろんである。

【００５２】

[メモリ読み出し書き込み回路制御信号による低消費電力化]

図４、図６、図８に示される実施例では、転送回路TGのバッファ回路TGBUFiをイネーブル信号でコントロールし無駄な消費電力が削減されるとともに論理回路LCのゲート電位がフローティング状態になることが防止される。

【００５３】

図１０には、さらにメモリコアMRの読み出し書き込み回路RWCを転送パターンに応じて制御することにより、使用しないメモリコアMRのI/O線MIOiを駆動することによる読み出し時の無駄な消費電力を削減し、さらに書き込み時に使用しないI/O線MIOiからメモリコアMRに誤ったデータが書き込まれるのを防止する例が示される。

【００５４】

図２の転送パターンのうちP2からP7は、メモリコアMRのI/O線MIOiの一部しか使用しない。したがって、メモリコアMRの書き込み読み出し回路RWCを制御する信号を設けて使用しないメモリコアMRのI/O線MIOiを受け持つ書き込み読み出し回路RWCiを停止する。図１０においてRWC0，RWC1，RWC2，RWC3は、読出し書込み回路RWCを構成する回路で、各々メモリコアMRのI/O線MIO0，MIO1，MIO2，MIO3用の書き込み読み出し回路RWCiである。また、MIOE0，MIOE1，MIOE2，MIOE3は、各々書き込み読み出し回路RWC0，RWC1，RWC3を制御するイネーブル信号である。

【００５５】

各々の転送パターンにおいて書き込み読み出し回路RWCiを制御するイネーブル信号MIOE0，MIOE1，MIOE2，MIOE3と論理回路LCのバッファ回路TGBUFiのイネーブル信号LIOEiの設定

法が図１１に示される。ここで、イネーブル信号の"1"は高電位を示し、活性状態で"0"は低電位で停止状態を示す。なお、イネーブル信号MIOE0，MIOE1，MIOE2，MIOE3をメモリコアMRに隣接した論理回路LCから発生する場合、図１１に示されるように転送回路TGを貫通して配線するとレイアウトを高密度にできる。

【００５６】

本実施例によれば、転送パターンに応じてメモリコアMRの読み出し書き込み回路RWCを制御することにより、使用しないI/O線MIOiを駆動することによる読み出し時の無駄な消費電力を削減し、さらに書き込み時に使用しないI/O線MIOiからメモリコアMRに誤ったデータが書き込まれるのを防止することができる。

【００５７】

[メモリ読み出し書き込み回路とバッファ制御信号の共用化]

図１０に示される実施例では、書き込み読み出し回路RWCを制御するイネーブル信号MIOEiと論理回路LCのバッファ回路TGBUFiのイネーブル信号LIOEiを独立にした。このようにすると図１１に示されるように転送パターンに合わせてそれぞれ異なる設定をする必要がある。I/O線の数と転送パターンの数が増えるとイネーブル信号MIOEiとLIOEiとを独立に設定するのは繁雑である。

【００５８】

図１２から図１５には、論理回路LCのバッファ回路TGBUFiのイネーブル信号LIOEi用の転送回路CTGを設けて書き込み読み出し回路RWCのイネーブル信号MIOEiをイネーブル信号LIOEiより自動的に発生するようにした例のが示される。図１２には、図２に示されるデータの転送パターンが示され。図１３には、図１２のデータ転送パターンに対応するバッファ回路TGBUFiの制御信号LIOEiの転送パターンでが示される。

【００５９】

この転送パターンに従ってバッファ回路TGBUFiの制御信号LIOEiをメモリコアMR側に転送してやれば、その信号をそのままメモリコアMRの書き込み読み出し回路RWCのイネーブル信号MIOEiとして使用することができる。

【００６０】

ここでデータが使用しないI/O線を受け持つ制御信号もメモリコアMRの書き込み読み出し回路RWCを停止するために転送する必要があることに注意しなければならない。すなわち、転送パターンP1からP7のようにデータは一部のI/O線しか使用しない場合にも、図８の中段に示されるように制御信号は全て転送される。

【００６１】

バッファ回路TGBUFiの制御信号LIOEiの転送回路CTGの具体的な構成が図１４に示される。データの転送回路TGと同じようにスイッチ群SWGEiからなる。この転送回路CTGによれば図１５に示されるように転送パターンに応じて制御信号ECiを設定することにより図８の中段に示される転送パターンが実現できる。

【００６２】

ここで、図１３に示される転送パターンを見るとP0，P2，P5の形が同じであることがわかる。そこで制御信号EC0，EC2，EC5に関するスイッチ群SWGEは一つにまとめて制御信号EC0，EC2，EC5のOR論理をとって入力される。これによりスイッチ群SWGEの段数を削減して高集積化を図ることができる。動作原理はこれまで説明してきたデータの転送回路TGと同じなので省略する。

【００６３】

本実施例によれば、データの転送回路TGに加えてバッファ回路TGBUFiの制御信号LIOEiの転送回路CTGを設けることにより、書き込み読み出し回路RWCのイネーブル信号MIOEiとバッファ回路TGBUFiのイネーブル信号LIOEiをそれぞれ独立に設定する必要がない。このため、I/O線の数や転送パターンの数が増えてもイネーブル信号の設定が繁雑になるのを避けることができる。

【００６４】

[データの転送単位を細かく設定可能なイネーブル信号]

これまでの実施例では、データの転送時にまとまって転送されるI/O線（図２では２の（n-2)乗）に対して書き込み読み出し回路RWCのイネーブル信号MIOEiとバッファのイネーブル信号LIOEiを設けていた。しかし、イネーブル信号の設定を細かくすることによりさらに多彩な転送パターンを実現することができる。

【００６５】

図１６及び図１７には、データの転送単位より細かく設定可能なイネーブル信号の例が示される。この実施例では、図３の転送パターンについてまとまって転送されるI/O線の単位を4Byteとし、イネーブル信号は、1Byte単位で設定した。すなわち、図１６に示されるように4Byteずつのメモリコア MRのI/O線MIOiと論理回路LCのI/O線LIOiとの間に図３に示される８種類の転送パターンが実現でき、イネーブル信号は4ByteのI/O線群に対して４本別々に設けてある。例えば、I/O線LIO0についてはLIOE-0，LIOE-1，LIOE-2，LIOE-3の４本のイネーブル信号がある。

【００６６】

図１７には、図１６の例で可能となる転送パターンの例とそのためのイネーブル信号の設定法が示される。イネーブル信号MIOEi-jはイネーブル信号LIOEi-jを転送することにより作ってもよいし、イネーブル信号LIOEi-jとは独立に設定してもよい。図１７の(A)は、転送回路TGで決まる基本転送パターンをP0とした状態でイネーブル信号を全て"1"とした場合である。これは、これまでのパターンと同じである。しかし、図１７の(B)のように基本転送パターンをP0としてイネーブル信号を2Byteづつ"0"と"1"とすると別の転送パターンを作ることができる。また、図１７の(C)は基本転送パターンP3で、図１７の(D)は、P3においてイネーブル信号の設定を変えたものである。

【００６７】

ここでは、二つの基本転送パターンについてそれぞれ一例のみ示したが、これ以外にもイネーブル信号を変えることによって基本転送パターンとは異なるさまざまな転送パターンができる。画像用途などでバイトごとにデータの属性が異なるような場合には、特定のバイトだけを転送する必要が有り得るが、そのような場合、本実施例が有用である。

【００６８】

図１８には、本発明を３次元コンピュータグラフィックス（以下3D-CGと記す）の描画処理を行うLSIに応用した例が示される。転送回路TGの基本転送パターンは、図３に示されるもので、図１８の上段に示されるようにメモリコアMRのデータをI/O線に割り付けられる。ここで、RGB-A，RGB-Bは、画素AおよびBの色、Z-A，Z-Bは画素AおよびBの奥行き座標で各々16bit長である。

【００６９】

3D-CGでは、Z比較という特別な処理がよく行われる。これは、よく知られているように新しくメモリへ画素の書き込みを行う場合、同じ位置の画素とZ値を比較して小さければ書き込み、大きければ書き込まないという処理である。このような処理を画素Aについて行う場合、図１８に示されるように、まず、転送パターンをP5として、メモリコアMRに既に記憶されているZ値Z-Aoldを読み出す。続いて、論理回路LCで新しい画素のZ値Zinと比較してZinが小さければ、新しい画素のRGBとZ値の書き込みを行う。ここで、転送パターンをP2に切り替えれば、RGBとZ値を並列に書き込むことができる。画素Bの場合には転送パターンP7とP3を用いればよい。なお、RGB値が3ByteでZ値が2Byteなどとビット数が異なる場合には、転送回路TGの基本転送パターンを3Byte単位として、Z値を扱う場合には図１６に示されるようなバイトごとのイネーブル信号を設けてマスクをかければよい。

【００７０】

3D-CGでは、さらにアルファブレンド処理という透明感を表わす処理がある。これを行うには図１８の中段に示されるようにすればよい。アルファブブレンド処理は、よく知られているように新しくメモリへ画素の書き込みを行う場合、同じ位置の画素を読み出して、新しい画素と係数αで重み付けして加算し、書き込むいう処理である。このような処理を画素Aについて行う場合、図１８に示されるように、まず、転送パターンをP4として、メモリコアMRに既に記憶されているRGB-Aoldを読み出す。続いて、論理回路LCで新しい画素

のRGBinと係数αで重み付けして加算し、書き込みを行う。転送パターンは、P4のままでよい。画素Bの場合には転送パターンP6を用いればよい。この場合もしも論理回路に重み付け加算を行う演算回路が一つしかない場合には、バイトごとのイネーブル信号を設けることによってRとGとBの1Byteづつアルファブレンド処理を行うことができる。

【００７１】

さらに画面クリアという処理も高速にできる。この処理では、メモリコアMR内のデータの初期化を行う。通常RGBについては、最小値か最大値、Z値については、奥行の最大になる最大値の書き込みを行う。図１８に示される実施例では、２つの画素分のI/O線があるので、転送パターンP1を利用すれば、２画素同時に書き込みが行えるため、高速にクリア処理ができる。さらに、図１８には示していないが、転送パターンP0とイネーブル信号を使えば２画素のRGBを同時に読み出すこともできるので高速の画面表示も行うことができる。以上述べたように、本発明の転送回路TGを用いれば高速の3D-CG描画処理を行うことができる。

【００７２】

[I/O線の割り付けをバイト毎にする例]

これまでは、説明を簡単にするため、転送の単位毎にI/O線MIOiやLIOiを割り振って図示してきた。実際のレイアウトでこのようにすると、特に転送の単位が大きい場合には多くのI/O線を横切ってデータが伝わるため、配線遅延や雑音の誘起など悪い影響が出る場合がある。

【００７３】

図１９には、I/O線の割り付けをバイト毎に変更した例が示される。図１９の例には、転送の単位が4Byteのときに1Byteづつ入れ子にする方法が示される。このようにするとデータの移動が少なくてすむ。たとえば、転送パターンP3では、図３に示されるようにすると8Byte分のI/O線を横切る必要があるが、図１９の例では2Byteで済む。ここではバイト毎に入れ替えたが、ビット毎に入れ替えてもよい。その場合にはさらに移動が少なくて済む。もちろん、本実施例のようにする場合は論理回路LCの受け口もそれに合わせて設計する必要があるが、配線遅延や雑音の誘起など悪い影響を避け、さらに配線の増加による面積の増大も低減することができる。

【００７４】

【発明の効果】

本発明によれば、メモリと論理回路を集積した半導体を短期間に設計でき、さらに面積の小さい転送回路によりリアルタイムで転送パターンを変えながら複数のI/O線をもつメモリコアと論理回路の間で高速なデータ転送を実現できる。

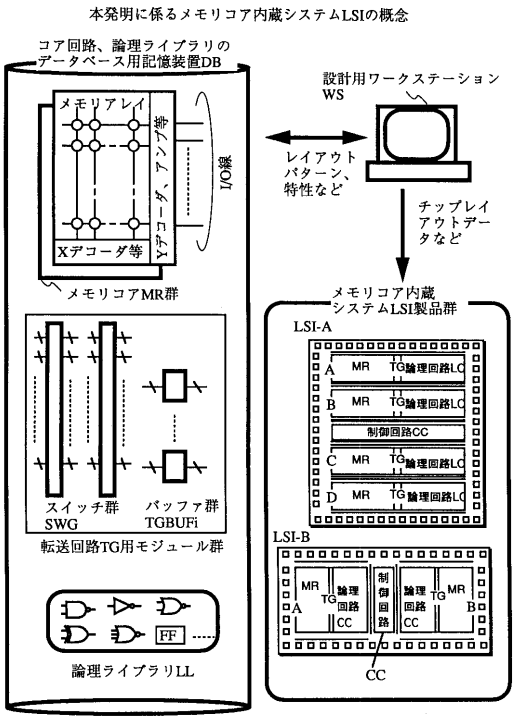
【図面の簡単な説明】

【図１】本発明に係る多重I/Oメモリコア内蔵システムLSIの概念。
【図２】本発明に係る多重I/Oメモリコア内蔵LSIの例。
【図３】図２の転送回路の転送パターン。
【図４】図３の転送パターンを実現する転送回路の第１の具体例。
【図５】転送回路のバッファ回路TGBUFiの具体例。
【図６】図３の転送パターンを実現する転送回路の第２の具体例。
【図７】図６の転送回路の制御信号の設定法。
【図８】図３の転送パターンを実現する転送回路の第３の具体例。
【図９】図８の転送回路の並列スイッチ部の回路構成とレイアウト例。
【図１０】メモリ読み出し書き込み制御信号による低消費電力化を行う例。
【図１１】図１０の転送回路の制御信号の設定法。
【図１２】データの転送パターン。
【図１３】バッファ制御信号の転送パターン。
【図１４】制御信号転送回路の例。
【図１５】図１４の制御信号転送回路の制御信号設定法。
【図１６】データの転送単位より細かく設定可能なイネーブル信号の例。

【図１７】図１６の例で可能となる転送パターンの例。
【図１８】３次元コンピュータグラフィックスへの応用例。
【図１９】I/O線のアドレスをバイト毎に変更した例。
【符号の説明】
MR　メモリコア
MC　メモリセル
DL　データ線
WL　ワード線
PER　周辺回路
RWC　読み出し書き込み回路
LC　論理回路
TG　転送回路
SWG　スイッチ群
TGBUFi　バッファ群
MIOi、TGCi、LIOi　制御信号
DB　コア回路、論理ライブラリのデータベース用記憶装置
LL　論理ライブラリ
WS　設計用ワークステーション
LSI-A、LSI-B　半導体チップ。

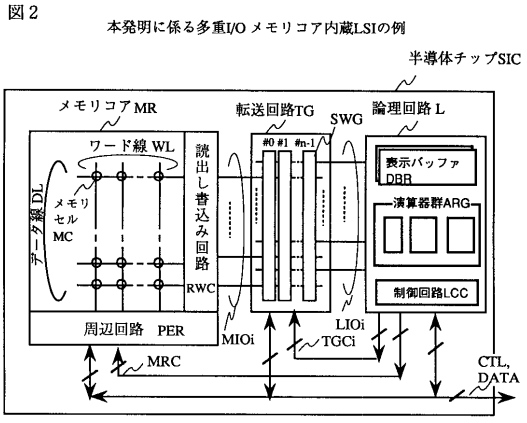
【図１】

図1



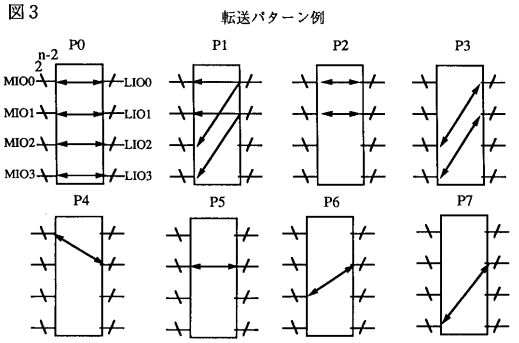
【図２】

図2



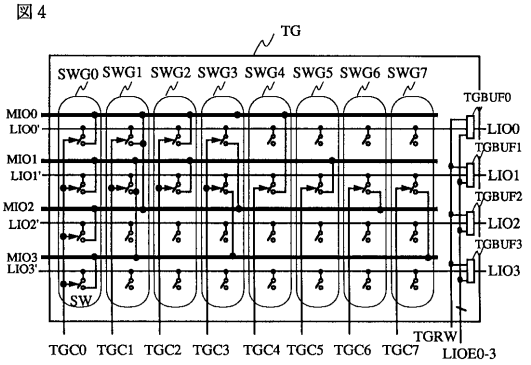
【図３】

図3



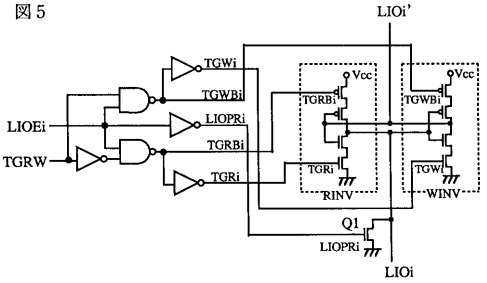
【図４】

図４



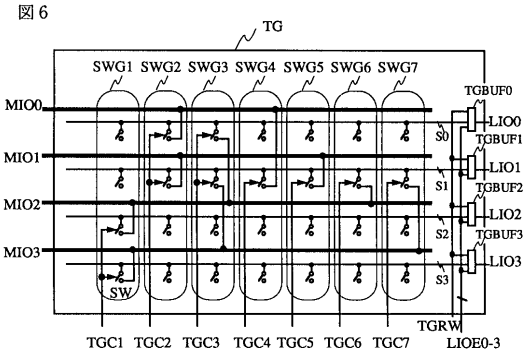
【図５】

図５



【図６】

図６



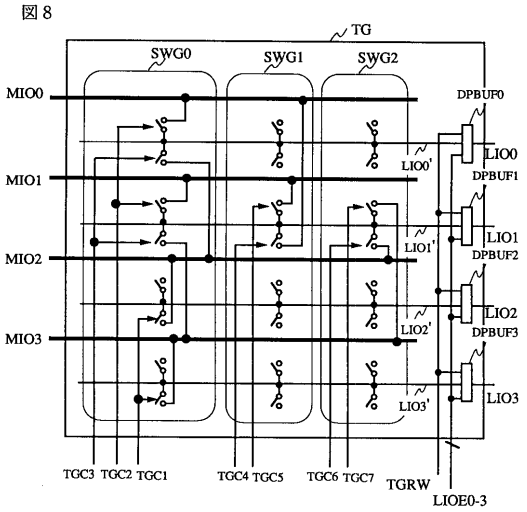
【図７】

図７

転送パターン	制御信号		
	TGCi 1 2 3 4 5 6 7	TGRW	LIOEi 0 1 2 3
P0	1 1 0 0 0 0 0	0/1	1 1 1 1
P1	0 1 1 0 0 0 0	1	1 1 0 0
P2	0 1 0 0 0 0 0	0/1	1 1 0 0
P3	0 0 1 0 0 0 0	0/1	1 1 0 0
P4	0 0 0 1 0 0 0	0/1	0 1 0 0
P5	0 0 0 0 1 0 0	0/1	0 1 0 0
P6	0 0 0 0 0 1 0	0/1	0 1 0 0
P7	0 0 0 0 0 0 1	0/1	0 1 0 0

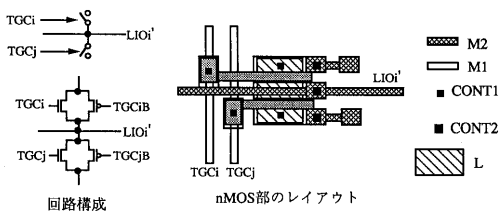
【図８】

図８



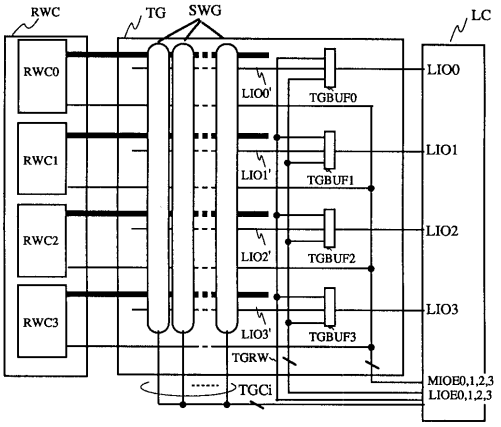
【図９】

図９



【図１０】

図１０



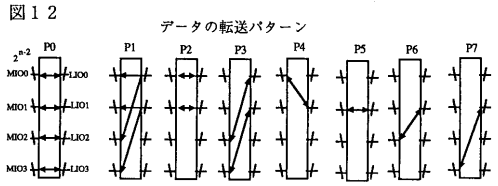
【図１１】

図１１

転送パターン	制御信号	
	MIOEi 0 1 2 3	LIOEi 0 1 2 3
P0	1 1 1 1	1 1 1 1
P1	1 1 1 1	1 1 0 0
P2	1 1 0 0	1 1 0 0
P3	0 0 1 1	1 1 0 0
P4	1 0 0 0	0 1 0 0
P5	0 1 0 0	0 1 0 0
P6	0 0 1 0	0 1 0 0
P7	0 0 0 1	0 1 0 0

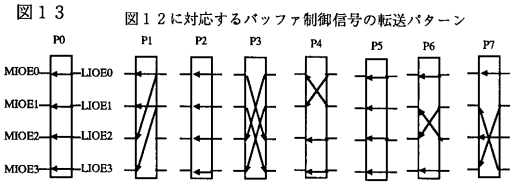
【図１２】

図１２



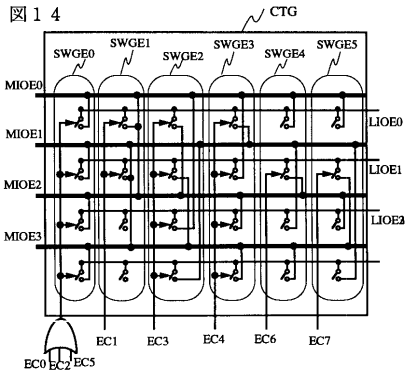
【図１３】

図１３



【図１４】

図１４



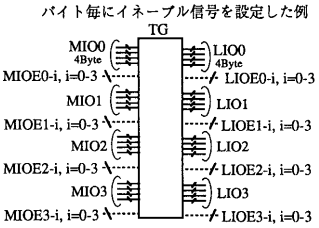
【図１５】

図１５

転送パターン	転送回路制御信号 ECi 0 1 2 3 4 5 6 7
P0	1 0 0 0 0 0 0 0
P1	0 1 0 0 0 0 0 0
P2	0 0 1 0 0 0 0 0
P3	0 0 0 1 0 0 0 0
P4	0 0 0 0 1 0 0 0
P5	0 0 0 0 0 1 0 0
P6	0 0 0 0 0 0 1 0
P7	0 0 0 0 0 0 0 1

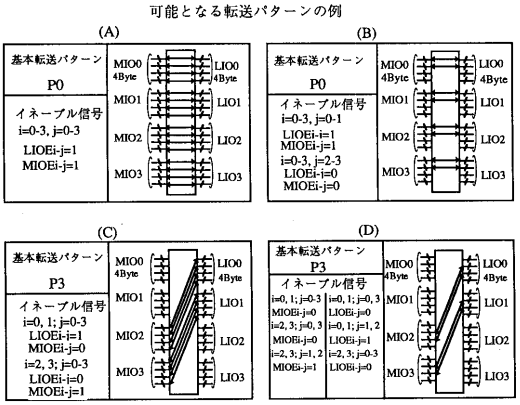
【図１６】

図１６



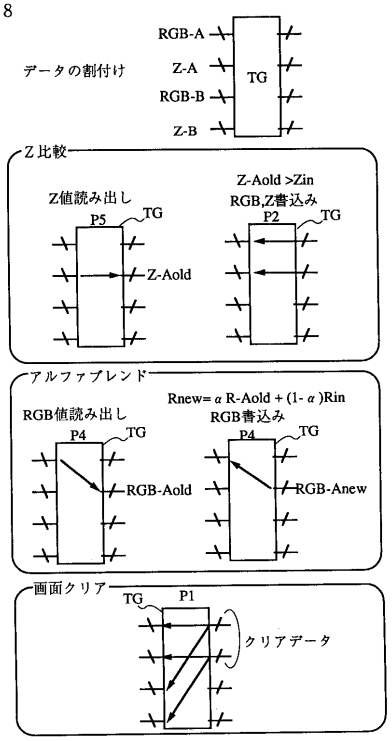
【図１７】

図１７



【図１８】

図１８



【図１９】

図１９

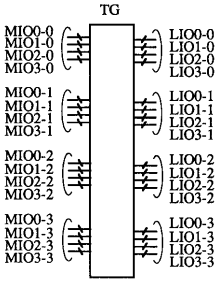
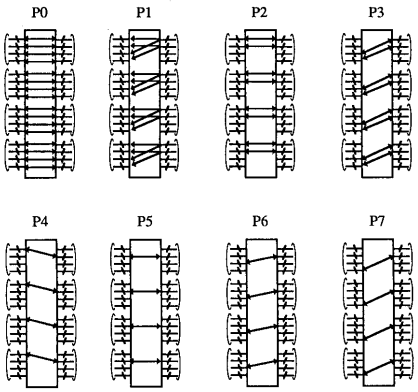


図３の転送パターンと等価な転送パターン



フロントページの続き

(72)発明者　鮎川　一重
東京都国分寺市東恋ケ窪１丁目２８０番地　　株式会社日立製作所中央研究所内
(72)発明者　藤田　　良
茨城県日立市大みか町七丁目１番１号　　株式会社日立製作所日立研究所内
(72)発明者　柳沢　一正
東京都小平市上水本町５丁目２０番１号　　株式会社日立製作所半導体事業部内
(72)発明者　田中　　均
東京都国分寺市東恋ケ窪１丁目２８０番地　　株式会社日立製作所中央研究所内

審査官　棚田　一也

(56)参考文献　特開平０５－２６７６２６（ＪＰ，Ａ）
特開平０７－０８４９８７（ＪＰ，Ａ）
特開昭６２－０８８３３６（ＪＰ，Ａ）
特開平０７－２９７２９１（ＪＰ，Ａ）
特開昭５７－０５４４３０（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.^{7}，ＤＢ名)
H01L 21/822
H01L 27/04
H03K 17/00
H01L 21/82
G11C 11/34
H03K 19/173
G06F 15/16