

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5683813号
(P5683813)

(45) 発行日 平成27年3月11日 (2015. 3. 11)

(24) 登録日 平成27年1月23日 (2015. 1. 23)

(51) Int. Cl.	F I
GO 6 F 12/06 (2006. 01)	GO 6 F 12/06 5 1 0 B
GO 6 F 12/04 (2006. 01)	GO 6 F 12/06 5 1 5 F
	GO 6 F 12/04 5 1 0 E

請求項の数 8 (全 53 頁)

(21) 出願番号	特願2009-539576 (P2009-539576)	(73) 特許権者	508034325
(86) (22) 出願日	平成19年12月4日 (2007. 12. 4)		コンバーサント・インテレクトチュアル・プロパティ・マネジメント・インコーポレイテッド
(65) 公表番号	特表2010-511943 (P2010-511943A)		CONVERSANT INTELLECTUAL PROPERTY MANAGEMENT INC.
(43) 公表日	平成22年4月15日 (2010. 4. 15)		カナダ・オンタリオ・K2K・OG7・オタワ・マーチ・ロード・390・スイート・100
(86) 国際出願番号	PCT/CA2007/002182		
(87) 国際公開番号	W02008/067658	(74) 代理人	100153811
(87) 国際公開日	平成20年6月12日 (2008. 6. 12)		弁理士 青山 高弘
審査請求日	平成22年11月29日 (2010. 11. 29)	(74) 代理人	100199277
審判番号	不服2013-24506 (P2013-24506/J1)		弁理士 西守 有人
審判請求日	平成25年12月13日 (2013. 12. 13)		
(31) 優先権主張番号	60/868, 773		
(32) 優先日	平成18年12月6日 (2006. 12. 6)		
(33) 優先権主張国	米国 (US)		
(31) 優先権主張番号	60/870, 892		
(32) 優先日	平成18年12月20日 (2006. 12. 20)		
(33) 優先権主張国	米国 (US)		

最終頁に続く

(54) 【発明の名称】 混合されたタイプのメモリデバイスを動作させるシステムおよび方法

(57) 【特許請求の範囲】

【請求項 1】

2 以上のタイプの複数の半導体デバイスがシリアル相互接続されたシリアル相互接続構成で使用される半導体デバイスであって、

前記シリアル相互接続構成の第1半導体デバイスが、デバイスタイプ識別、コマンド、およびデバイスアドレス識別を含むシリアル入力を受け取り、前記シリアル入力が、前記シリアル相互接続構成を介して伝搬され、

前記半導体デバイスは、

前記シリアル入力を受け取るための入力接続を備え、

前記シリアル入力のデバイスタイプ識別が当該半導体デバイスに関連するデバイスタイプと一致するかどうか判定し、デバイスタイプ識別が関連するデバイスタイプと一致した場合にデバイスタイプ一致結果を提供し、デバイスタイプ識別が関連するデバイスタイプと一致しない場合に非デバイスタイプ一致結果を提供し、

前記シリアル入力のデバイスアドレス識別が当該半導体デバイスに関連するデバイスアドレスと一致するかどうか判定し、デバイスアドレス識別が関連するデバイスアドレスと一致した場合にデバイスアドレス一致結果を提供し、デバイスアドレス識別が関連するデバイスアドレスと一致しない場合に非デバイスアドレス一致結果を提供し、

前記デバイスタイプ一致結果と前記デバイスアドレス一致結果にตอบสนองして前記コマンドを実行し、

前記非デバイスタイプ一致結果または前記非デバイスアドレス一致結果にตอบสนองして前記

10

20

シリアル入力を前記シリアル相互接続構成の次の半導体デバイスに送る、
よう構成されている、半導体デバイス。

【請求項 2】

前記半導体デバイスの前記タイプを示す前記デバイスタイプを保持するデバイスタイプホルダと、

前記デバイスアドレスを保持するアドレスホルダであって、該アドレスホルダに保存されたアドレスが、前記シリアル相互接続構成における前記半導体デバイスのアドレスを示す割り当てられたアドレスである、アドレスホルダと、

をさらに含む、請求項1に記載の半導体デバイス。

【請求項 3】

前記デバイスタイプ識別が前記デバイスタイプホルダによって保持される前記デバイスタイプと一致するかどうかを判定するタイプ一致判定部であって、それぞれ、前記デバイスタイプ識別が前記保持されるデバイスタイプと一致する場合に前記デバイスタイプ一致結果を、前記デバイスタイプ識別が前記保持されるデバイスタイプと一致しない場合に前記非デバイスタイプ一致結果を提供することができる、タイプ一致判定部と、

前記デバイスアドレス識別が前記アドレスホルダによって保持される前記デバイスアドレスと一致するかどうかを判定するアドレス一致判定部であって、それぞれ、前記デバイスアドレス識別が前記保持されるデバイスアドレスと一致する場合に前記デバイスアドレス一致結果を、前記デバイスアドレス識別が前記保持されるデバイスアドレスと一致しない場合に前記非デバイスアドレス一致結果を提供することができる、アドレス一致判定部とをさらに含む、請求項2に記載の半導体デバイス。

【請求項 4】

前記受け取られたシリアル入力に応答して前記半導体デバイスの動作を制御し、

前記デバイスタイプ一致結果および前記デバイスアドレス一致結果に応答して、前記受け取られたシリアル入力に含まれる前記コマンドを実行するデバイスコントローラをさらに含む、請求項3に記載の半導体デバイス。

【請求項 5】

前記デバイスコントローラが、前記非デバイスタイプ一致結果および前記非デバイスアドレス一致結果のうちのいずれか1つに応答して、前記シリアル相互接続構成の前記次の半導体デバイスに、前記受け取られたシリアル入力の前記デバイスタイプ識別、前記コマンド、および前記デバイスアドレス識別を転送する、請求項4に記載の半導体デバイス。

【請求項 6】

前記半導体デバイスが、メモリを含み、

前記シリアル入力に含まれる前記データ情報が、前記メモリに関する情報データをさらに含む、

前記デバイスコントローラが、前記デバイスタイプ一致結果および前記デバイスアドレス一致結果に応答して、前記メモリにアクセスするために、前記メモリに関する情報データに基づいて前記コマンドを実行することができる、

請求項4に記載の半導体デバイス。

【請求項 7】

前記デバイスコントローラが、前記非デバイスタイプ一致結果および前記非デバイスアドレス一致結果のいずれかに応答して、前記シリアル相互接続構成の前記次の半導体デバイスに、前記受け取られたシリアル入力に含まれる前記デバイスタイプ識別、前記デバイスアドレス識別、前記コマンド、および前記データを転送することがさらにできる、請求項6に記載の半導体デバイス。

【請求項 8】

前記メモリが、NANDフラッシュEEPROM、NORフラッシュEEPROM、ANDフラッシュEEPROM、DiNORフラッシュEEPROM、シリアルフラッシュEEPROM、DRAM、SRAM、ROM、EPROM、FRAM、MRAM、およびPCRAMのうちのいずれか1つを含む、請求項6に記載の半導体デバイス。

【発明の詳細な説明】

【技術分野】

【0001】

関連出願の相互参照

本願は、その開示が参照によってその全体を本明細書に組み込まれている、2006年12月6日に出願した米国仮出願第60/868,773号、2006年12月20日に出願した米国仮出願第60/870,892号、2007年1月12日に出願した米国特許出願第11/622,828号、および2007年6月29日に出願した米国特許出願第11/771,241号の利益を主張するものである。

【0002】

本発明は、全般的には半導体デバイスシステムに関するものである。より具体的には、本発明は、例えば変化するタイプまたは混合されたタイプのメモリデバイスを有するメモリシステムなどの半導体デバイスを制御する装置および方法に関するものである。

10

【背景技術】

【0003】

コンピュータベースのシステムは、例えばメモリデバイスおよび処理デバイスなどの半導体デバイスを含む。メモリは、情報が、コンピュータの中央処理装置(CPU)によって操作されるのを待ちながら格納される場所である。メモリは、メモリコントローラによって制御され、このメモリコントローラは、CPUの一部を形成することができ、あるいは、CPUとは別々としてすることができる。メモリコントローラは、情報を通信するためのメモリとのインターフェースを有する。既知のインターフェースは、パラレルインターフェースおよびシリアルインターフェースを含む。

20

【0004】

パラレルインターフェースは、データを読み取り、書き込むのに多数のピンを使用する。残念ながら、入力ピンおよびワイヤの数が増えるにつれて、望ましくない影響の数も増える。これらの望ましくない影響には、シンボル間干渉、信号スキュー、およびクロストークが含まれる。したがって、当技術分野には、メモリモジュールにアクセスするための入力ピンおよびワイヤの数を最小にしながら、増やされたメモリ容量および/または高められた動作速度を有するメモリモジュールが必要とされる。

【0005】

シリアルインターフェースは、データを読み取り、書き込むのにより少数のピンを使用する。シリアルフラッシュメモリが、現在入手可能であるが、これは、非常に低速である傾向がある。例えば、多数の従来のメモリは、SPI (Serial Peripheral Interface) 互換インターフェースまたはI²C (Inter-Integrated Circuit) 互換インターフェースを用いて1MHz~20MHzの範囲で動作するシリアルバスインターフェース方式を使用している。しかし、このシリアルインターフェース標準規格は、通常、そのパラレル対応物より低速である。

30

【0006】

図1A、1B、1C、および1Dを参照すると、4つの主要なフラッシュメモリアーキテクチャが示されている。この4つの主要なフラッシュメモリアーキテクチャは、図1Aに示された伝統的なXIPモデル、図1Bに示されたシャドウモデル、図1Cに示されたNANDを用いるストアアンドダウンロード(store-and-download)モデル、および図1Dに示されたハイブリッド化されたNANDフラッシュメモリを用いるより新しいストアアンドダウンロードモデルを含む。

40

【0007】

図1Aを参照すると、伝統的なXIPモデルは、アプリケーションプロセッサ101に接続された、NORフラッシュメモリ102とおそらくはSRAM (スタティックランダムアクセスメモリ) またはDRAM (ダイナミックランダムアクセスメモリ)である揮発性メモリ103とを有する。XIPモデルでは、NORフラッシュメモリ102は、コードを実行し、揮発性メモリ103は、変数、スタック、およびヒープなどの常に変化するシステム要素からなる。XIPモデルでは、NORフラッシュメモリ102は、データおよびコードストレージをも提供することもできる。XIPモデルの利益は、単純さであるが、不利益は、その遅い書き込み速度である。

50

【 0 0 0 8 】

図1Bを参照すると、シャドウモデルは、アプリケーションプロセッサ104に接続された、NORフラッシュメモリ105、NANDフラッシュメモリ106、およびおそらくはSRAMまたはDRAMである揮発性メモリ107を有する。ユーザは、NORフラッシュメモリ105を用いてシステムをブートし、ストレージにNANDフラッシュメモリ106を使用する。揮発性メモリ107は、実行のすべてを処理する。シャドウモデルは、相対的に高価なNORフラッシュメモリ105をシステムのブートアップだけに使用しているという点で、高価なモデルである。このアーキテクチャは、多少より複雑でもあり、これは、このアーキテクチャがより多くの設計時間およびコストを消費することを意味する。シャドウモデルは、揮発性メモリが常にアクティブなので、電力を多く必要とする傾向もある。

10

【 0 0 0 9 】

例えばモバイルハンドヘルドデバイスで莫大な要因である、スペース問題を克服するために、図1Cに示された、ストアアンドダウンロードアーキテクチャが使用される。ストアアンドダウンロードアーキテクチャは、アプリケーションプロセッサ108に接続された、NANDフラッシュメモリ110およびおそらくはSRAMまたはDRAMである揮発性メモリ111を有する。ストアアンドダウンロードアーキテクチャは、NORフラッシュメモリを有しないが、アプリケーションプロセッサ108内に設計されたOPT (one-time-programmable) ストレージ109またはROM(読取り専用メモリ)コアがある。アプリケーションプロセッサ108は、情報を揮発性メモリ111にロードし、この揮発性メモリ111は、データストレージのためにNANDフラッシュメモリ110にアクセスする。このアーキテクチャは、多少より複雑であり、より多くの初期エンジニアリングコストを必要とするが、最終的に、このシステムのユニットコストは、より安価である。このモデルの主な問題は、NANDフラッシュメモリが、通常はより信頼性が低いので、ユーザが、広範囲のエラー訂正およびエラー検出コーディングを使用しなければならないことである。ストアアンドダウンロード設計は、RAMがより能動的な役割を引き受けるので、より多くの電力を必要とする傾向がある。

20

【 0 0 1 0 】

図1Dを参照すると、ハイブリッドストアアンドダウンロードモデルは、アプリケーションプロセッサ112に接続された、ハイブリッドNANDフラッシュメモリ113およびおそらくはSRAMまたはDRAMである揮発性メモリ114を有する。ハイブリッドNANDフラッシュメモリ113は、SRAM、制御ロジック、およびNANDフラッシュメモリを混合して、NORフラッシュデバイスのように見えると思われるメモリデバイスを作り出す。このハイブリッドモデルは、標準NANDフラッシュデバイスよりはるかに高速に、NORフラッシュデバイスと同一の速度で読み取る。このハイブリッドモデルは、NORフラッシュデバイスよりよい書込み性能をも提供する。ハイブリッドNANDフラッシュメモリは、現在入手可能である。ハイブリッドモデルは、標準NANDフラッシュメモリを用いるストアアンドダウンロードモデルより少ないエラー訂正およびエラー検出コーディングを必要とする。ハイブリッド化されたNANDフラッシュメモリのユニットコストは、例えば、同一密度のNORフラッシュメモリより30%から40%安い。独立NANDフラッシュメモリのコストは、ハイブリッド化されたNANDフラッシュメモリのコストよりわずかに低い。

30

【 0 0 1 1 】

上記の4つの主要なフラッシュメモリアーキテクチャのいずれかを使用するメモリシステムは、エンジニアリング設計、ソフトウェア開発、および検証に長い時間を必要とする。

40

【 先行技術文献 】

【 特許文献 】

【 0 0 1 2 】

【 特許文献 1 】 米国仮出願第60/868,773号

【 特許文献 2 】 米国仮出願第60/870,892号

【 特許文献 3 】 米国特許出願第11/622,828号

【 特許文献 4 】 米国特許出願第11/771,241号

50

【特許文献5】米国特許出願第11/324,023号

【特許文献6】米国仮出願第60/787,710号

【特許文献7】米国仮出願第60/802,645号

【特許文献8】米国特許出願第11/529,293号

【発明の概要】

【課題を解決するための手段】

【0013】

広義の態様によれば、変化するタイプまたは混合されたタイプのメモリデバイスを含むシステムまたは装置であって、メモリデバイスがシリアル相互接続され、その結果、入力データがデバイスからデバイスへシリアル転送されるようになっているシステムまたは装置が提供される。

10

【0014】

もう1つの広義の態様によれば、混合されたタイプの複数のデバイスのシリアル相互接続構成で使用される半導体デバイスであって、複数のデバイスが、シリアル相互接続される、半導体デバイスが提供される。シリアル相互接続構成の第1デバイスは、シリアル入力を受け取る。シリアル入力は、シリアル相互接続構成を介して伝搬される。シリアル入力は、デバイスタイプ識別、コマンド、およびデバイスアドレス識別を含む。デバイスは、デバイスタイプ識別およびデバイスアドレス識別に基づいてコマンドを実行する。

【0015】

半導体デバイスは、受け取られたシリアル入力に応答してデバイスの動作を制御するデバイスコントローラを含むことができる。

20

【0016】

例えば、半導体デバイスは、デバイスタイプ識別を保持するデバイスタイプホルダであって、保持されるタイプ識別がデバイスのタイプを示すために提供される、デバイスタイプホルダと、提供されるシリアル入力に応答して割り当てられるデバイスアドレスを保持するアドレスホルダであって、割り当てられたアドレスが、デバイスのアドレスの表示のために提供される、アドレスホルダとをさらに含むことができる。

【0017】

もう1つの広義の態様によれば、混合されたタイプの複数のデバイスであって、デバイスが、デバイスがシリアル接続されるシリアル相互接続構成で構成される、複数のデバイスを含むシステムが提供される。デバイスのそれぞれは、シリアル入力接続およびシリアル出力接続を有する。このシステムは、シリアル相互接続構成の第1デバイスのシリアル入力接続にシリアル入力を提供するシリアル出力接続を有するシリアル出力/入力コントローラをさらに含む。シリアル入力は、シリアル相互接続構成を介して伝搬される。シリアル出力/入力コントローラは、シリアル相互接続構成の最後のデバイスからシリアル出力を受け取るシリアル入力接続を有する。シリアル入力は、デバイスタイプ識別、コマンド、およびデバイスアドレス識別を含む。

30

【0018】

例えば、複数のデバイスは、1つのシリアル相互接続構成で構成され、デバイスのタイプは、混合されている。デバイスのそれぞれは、受け取られたシリアル入力に応答してデバイスの動作を制御するデバイスコントローラを含むことができる。また、デバイスのそれぞれは、デバイスに割り当てられるデバイスアドレスを示すデバイスアドレスインジケータと、そのデバイスのデバイスタイプ識別を示すデバイスタイプインジケータとをさらに含むことができる。

40

【0019】

もう1つの広義の態様によれば、混合されたタイプの複数のデバイスを動作させる方法であって、デバイスが、デバイスがシリアル相互接続される少なくとも1つのシリアル相互接続構成で構成され、シリアル相互接続構成の第1デバイスにシリアル入力を提供するステップであって、シリアル入力が、シリアル相互接続構成を介して伝搬され、シリアル入力が、デバイスタイプ識別、コマンド、およびデバイスアドレス識別を含む、提供する

50

ステップを含む、方法が提供される。

【0020】

この方法は、デバイスのデバイスタイプ識別を保持するステップと、提供されるシリアル入力に応答して割り当てられるデバイスアドレスを保持するステップとをさらに含むことができる。この方法は、受け取られたデバイスタイプ識別が、保持されるデバイスタイプ識別と一致するかどうかを判定するステップをさらに含むことができる。有利なことに、受け取られたデバイスタイプ識別が、保持されるデバイスタイプ識別と一致する場合に、デバイスタイプ一致結果が提供され、受け取られたデバイスタイプ識別が、保持されるデバイスタイプ識別と一致しない場合に、非デバイスタイプ一致結果が提供される。

【0021】

この方法は、受け取られたデバイスアドレス識別が、保持されるデバイスアドレスと一致するかどうかを判定するステップをさらに含むことができる。有利なことに、受け取られたデバイスアドレス識別が、保持されるデバイスアドレスと一致する場合に、デバイスアドレス一致結果が提供され、受け取られたデバイスアドレス識別が、保持されるデバイスアドレスと一致しない場合に、非デバイスアドレス一致結果が提供される。この方法は、デバイスタイプ一致結果およびデバイスアドレス一致結果に応答して、シリアル入力の受け取られたコマンドを実行することができる。

【0022】

もう1つの広義の態様によれば、混合されたタイプの複数のデバイスを動作させる装置であって、デバイスが、デバイスがシリアル相互接続される少なくとも1つのシリアル相互接続構成で構成され、装置が、シリアル相互接続構成の第1デバイスにシリアル入力を提供するコントローラであって、シリアル入力が、シリアル相互接続構成を介して伝搬され、シリアル入力が、デバイスタイプ識別、コマンド、およびデバイスアドレス識別を含む、コントローラを含む、装置が提供される。

【0023】

例えば、デバイスのそれぞれが、シリアル入力接続およびシリアル出力接続を有し、コントローラが、第1デバイスのシリアル入力接続に接続されたシリアル出力接続と、シリアル相互接続構成の最後のデバイスのシリアル出力接続に接続されたシリアル入力接続とを有する。

【0024】

もう1つの広義の態様によれば、混合されたタイプの複数のデバイスにデバイスアドレスを割り当てるステップであって、デバイスが、デバイスがシリアル相互接続される少なくとも1つのシリアル相互接続構成で構成される、割り当てるステップと、デバイスタイプおよびデバイスアドレスに基づいてシリアル相互接続構成のデバイスにアクセスするステップとを含む方法が提供される。

【0025】

この方法は、1つのタイプの前記デバイスのそれぞれにおいてアドレスを確立するステップをさらに含むことができる。例えば、確立するステップは、デバイスタイプ識別、デバイスアドレス識別、およびアドレス番号を含むシリアル入力をシリアル相互接続構成の第1デバイスに提供するステップを含む。アクセスするステップは、シリアル相互接続構成内の第1タイプの少なくとも1つのデバイスを処理することを可能にするステップと、第1タイプのデバイスの処理中にシリアル相互接続構成内の第2タイプの少なくとも1つのデバイスを処理することを可能にするステップであって、第1タイプのデバイスの処理時間が、第2タイプのデバイスの処理時間より長い、可能にするステップとを含むことができる。

【0026】

本発明の実施形態によれば、メモリコントローラがシリアルリンクと相互接続されたメモリデバイスを制御する、メモリシステムアーキテクチャが提供される。メモリコントローラは、メモリコマンドを送る出力インターフェースと、応答を要請するメモリコマンドに関するメモリ応答を受け取る入力インターフェースとを有する。各メモリデバイスは

10

20

30

40

50

、例えばNANDフラッシュまたはNORフラッシュなど、任意のメモリタイプとすることができ。各メモリコマンドは、所期のメモリデバイスのメモリタイプに固有である。メモリコマンドおよびメモリ応答のデータパスは、リンクおよび相互接続されたデバイスを介して供給される。所与のメモリコマンドが、その所期のメモリデバイスに達するために多数のメモリデバイスをトラバースすることができる。その受取の際に、所期のメモリデバイスは、所与のメモリコマンドを実行し、適当な場合に、メモリコントローラにメモリ応答を送る。

【0027】

一実施形態で、メモリコントローラによって送られるメモリコマンドは、クロックに回答して、シリアル相互接続されたメモリデバイスを介して伝搬される。あるメモリデバイスのコマンド実行は、クロックタイミングにおいて別のメモリデバイス(例えば、次のデバイス)のコマンド実行とオーバーラップされない。また、メモリデバイスのコマンド実行を、互いにオーバーラップさせることができる。アドレス割当て動作では、あるデバイスによるアドレス番号変更は、別のデバイスがアドレス番号変更を実行する前に完了される。

【0028】

本発明の実施形態によれば、例えば、NANDフラッシュEEPROM、NORフラッシュEEPROM、ANDフラッシュEEPROM、DiNORフラッシュEEPROM、シリアルフラッシュEEPROM、DRAM、SRAM、ROM、EPROM、FRAM、MRAM、およびPCRAMのメモリタイプを有するメモリデバイスが提供される。ターゲットアドレスに基づく、混合されたタイプのメモリデバイスのシリアル相互接続構成を有するメモリシステムでは、各デバイスのメモリタイプを読み取ることができる。

【0029】

本発明の他の態様および特徴は、添付図面と共に本発明の特定の実施形態の次の説明を再検討する時に、当業者に明白になるであろう。

【0030】

これから、例としてののみ、添付図面を参照して本発明の実施形態を説明する。

【図面の簡単な説明】

【0031】

【図1A】フラッシュメモリアーキテクチャを示すブロック図である。

【図1B】フラッシュメモリアーキテクチャを示すブロック図である。

【図1C】フラッシュメモリアーキテクチャを示すブロック図である。

【図1D】フラッシュメモリアーキテクチャを示すブロック図である。

【図2A】本発明の実施形態によるメモリシステムを示すブロック図である。

【図2B】本発明の実施形態によるメモリシステムを示すブロック図である。

【図2C】図2Bに示されたメモリシステムの動作を示す流れ図である。

【図3A】本発明の実施形態による特定の例のメモリシステムを示す概略図である。

【図3B】本発明の実施形態による特定の例のメモリシステムを示す概略図である。

【図3C】本発明の実施形態による特定の例のメモリシステムを示す概略図である。

【図3D】本発明の実施形態による特定の例のメモリシステムを示す概略図である。

【図3E】本発明の実施形態による特定の例のメモリシステムを示す概略図である。

【図3F】本発明の実施形態によるもう1つの例のメモリシステムを示す概略図である。

【図3G】メモリデバイスのシングルデータレート動作を示すタイミング図である。

【図3H】メモリデバイスのダブルデータレート動作を示すタイミング図である。

【図4A】図3A、3B、3C、3D、および3Eに示されたメモリデバイスとして使用される例のメモリデバイスを示す概略図である。

【図4B】図3Fに示されたメモリデバイスとして使用される例のメモリデバイスを示す概略図である。

【図5A】メモリデバイスのメモリタイプを識別するのに使用される例のレジスタブロックを示す概略図である。

10

20

30

40

50

【図 5 B】各メモリデバイスタイプの例の符号化方式を示す表である。

【図 6 A】タイプ依存アドレッシングを用いてデバイスアドレスを割り当てる方法を示す流れ図である。

【図 6 B】図 6A に示された方法のデバイスアドレス割当てステップの詳細を示す流れ図である。

【図 7 A】タイプ依存アドレッシングを用いるデバイスアドレスの割当てを示すタイミング図である。

【図 7 B】タイプ依存アドレッシングを用いるデバイスアドレスの割当てを示すタイミング図である。

【図 7 C】タイプ依存アドレッシングを用いるデバイスアドレスの割当てを示すタイミング図である。

10

【図 7 D】タイプ依存アドレッシングを用いるデバイスアドレスの割当てを示すタイミング図である。

【図 8】タイプ依存アドレッシングを用いる例の入力を示すタイミング図である。

【図 9】2つの隣接するメモリデバイスを介する例のシグナリングを示すタイミング図である。

【図 10】タイプ依存アドレッシングを用いるメモリ動作の例の所定のフォーマットを示す表である。

【図 11】タイプ依存アドレッシングの例の符号化方式を示す表である。

【図 12】タイプ依存アドレッシングを用いる NAND フラッシュコマンドの例の符号化方式を示す図である。

20

【図 13】タイプ依存アドレッシングを用いる NOR フラッシュコマンドの例の符号化方式を示す図である。

【図 14】タイプ依存アドレッシングを用いてメモリ動作を処理する方法を示す流れ図である。

【図 15 A】タイプ依存アドレッシングを用いるメモリ動作の処理を示すタイミング図である。

【図 15 B】タイプ依存アドレッシングを用いるメモリ動作の処理を示すタイミング図である。

【図 16 A】図 3A、3B、3C、3D、および 3E に示されたメモリデバイスとして使用されるもう1つの例のメモリデバイスブロックを示す概略図である。

30

【図 16 B】図 16A に示されたデバイスによるデバイスアドレス割当て動作を示す流れ図である。

【図 16 C】図 16A に示されたデバイスによるもう1つのデバイスアドレス割当て動作を示す流れ図である。

【図 17】図 3F に示されたメモリデバイスとして使用されるもう1つの例のメモリデバイスブロックを示す概略図である。

【図 18】本発明のもう1つの実施形態による2チャネルのメモリシステムを示す図である。

【図 19 A】図 18 に示されたメモリシステム内で使用される特定の例のメモリデバイスを示す概略図である。

40

【図 19 B】図 18 に示されたメモリシステム内で使用される特定の例のメモリデバイスを示す概略図である。

【図 20 A】図 18 に示されたメモリシステム内で使用される他の特定の例のメモリデバイスを示す概略図である。

【図 20 B】図 18 に示されたメモリシステム内で使用される他の特定の例のメモリデバイスを示す概略図である。

【図 21】メモリシステムの初期化のもう1つの例を示すタイミング図である。

【図 22】本発明の実施形態によるもう1つの例のメモリシステムを示す概略図である。

【図 23】本発明のもう1つの実施形態によるメモリシステムを示すブロック図である。

50

【図24】図23に示されたメモリシステム内で使用される特定の例のメモリデバイスを示す概略図である。

【図25】本発明のもう1つの実施形態によるメモリシステムを示すブロック図である。

【図26】図25に示されたメモリシステム内で使用される特定の例のメモリデバイスを示す概略図である。

【発明を実施するための形態】

【0032】

本発明のサンプル実施形態の次の詳細な説明では、添付図面を参照するが、添付図面は、本明細書の一部を形成し、本発明を実践できる特定のサンプル実施形態を例として示すものである。これらの実施形態は、当業者が本発明を実践することを可能にするのに十分に詳細に説明されるが、他の実施形態を利用できることと、本発明の範囲から逸脱せずに論理的变化、機械的变化、電気的变化、および他の変化を行えることを理解されたい。したがって、次の詳細な説明は、限定的な意味で解釈されてはならず、本発明の範囲は、添付の特許請求の範囲によって定義される。

【0033】

全般的に、本発明は、例えば混合されたタイプのメモリデバイスを有するメモリシステムなど、半導体デバイスを制御する装置および方法を提供する。

【0034】

本発明の実施形態を、これから、メモリシステムの態様で説明する。メモリシステムは、メモリコントローラおよびメモリデバイスのシリアル相互接続構成を含む。

【0035】

一部のメモリサブシステムは、インターフェースと共に、例えばフラッシュメモリデバイスなどの複数のメモリデバイスを使用する。ここで、コマンドストリングを、そのコマンドがデバイスのうちの1つで実行されるのみである可能性がある場合であっても、デバイスのすべてに供給することができる。コマンドがその上で実行されるデバイスを選択するために、コマンドストリングは、そのコマンドが向けられるメモリデバイスを識別するデバイス識別子(ID)またはデバイスアドレスを含むことができる。コマンドストリングを受け取る各デバイスは、コマンドストリングに含まれるIDをそのデバイスに関連するIDと比較する。この2つが一致する場合には、そのデバイスは、そのコマンドがそのデバイスに向けられると仮定して、そのコマンドを実行する。

【0036】

前に述べたように、異なるインターフェース仕様を有する多数の異なるメモリデバイスタイプがある。伝統的なアーキテクチャを使用する変化するデバイスタイプまたは異なるデバイスタイプを有するメモリシステムの設計は、エンジニアリング設計、ソフトウェア開発、および検証に長い時間を伴う。また、パラレルインターフェース方式は、PCB (プリント回路基板)またはMCP (マルチチップパッケージ)上の多すぎる物理的配線またはルーティングを伴う可能性があり、この配線またはルーティングは、より高速の動作でさまざまな雑音問題を引き起こす可能性がある。より多くの信号線は、システムが密度および特徴において増大する時に、より複雑な基盤設計およびより多くの空間要件を意味する。改善されたメモリシステムアーキテクチャの必要がある。

【0037】

メモリデバイスのシリアル相互接続の詳細は、その内容が全体的に参照によって本明細書に組み込まれている、2005年12月30日に出願した米国特許出願第11/324,023号、2006年3月28日に出願した米国仮出願第60/787,710号、名称「Serial Interconnection of Memory Devices」、および2006年5月23日に出願した米国仮出願第60/802,645号、名称「Serial Interconnection of Memory Devices」に開示されている。

【0038】

図2Aに、本発明の実施形態によるメモリシステムを示す。図2Aを参照すると、メモリシステムは、コントローラ100ならびにシリアル相互接続構成の複数のデバイス300-0、300-1、...、および300-Nを含む。Nは、1より大きい整数である。この特定の例では、シリアル

10

20

30

40

50

相互接続されたメモリデバイスの個数は、 $(N+1)$ である。コントローラ100ならびにデバイス300-0、300-1、...、および300-Nは、例えばリンクなど、任意の適当な接続を使用して相互接続される。図示の例では、リンクは、シリアルリンクである。コントローラ100ならびにデバイス300-0、300-1、...、および300-Nは、シリアルリンクL0、L1、L2、...、LNおよびL(N+1)を介して相互接続される。

【0039】

コントローラ100は、コントローラ動作回路網130を有する。デバイス300-0、300-1、...、および300-Nのそれぞれは、デバイス動作回路網230を有し、デバイス動作回路網230は、メモリ動作制御およびメモリ初期化機能を実行する。デバイス300-0、300-1、...、および300-Nは、それぞれのメモリ320-0、320-1、...、および320-Nなどのそれぞれのメモリタイプ固有コンポーネントを有する。デバイス300-0、300-1、...、および300-Nのそれぞれは、複数のサポートされるメモリタイプからのメモリタイプを有する。複数のサポートされるメモリタイプは、実施態様固有の基礎で定義される。各デバイスのメモリタイプに関する情報またはメモリタイプの識別は、そのレジスタ250に格納される。しかし、デバイスのタイプは、コントローラ100には未知である。コントローラ動作回路網およびデバイス動作回路網のそれぞれは、入出力回路網、例えばインターフェース回路網を含む。

【0040】

図2Bに、本発明の一実施形態による例のメモリシステムを示す。図2Bを参照すると、メモリシステム40は、メモリコントローラ10ならびにシリアル相互接続構成の複数のメモリデバイス30-0、30-1、...、および30-Nを含む。Nは、1より大きい整数である。この特定の例では、シリアル相互接続されたメモリデバイスの個数は、 $(N+1)$ である。メモリコントローラ10ならびにメモリデバイス30-0、30-1、...、および30-Nは、例えば、シリアルリンクL0、L1、L2、...、LNおよびL(N+1)などを使用して相互接続される。

【0041】

メモリコントローラ10は、出力インターフェース11、入力インターフェース12、およびコントローラ動作回路網13を有する。いくつかの実施態様で、図示の例に示されているように、メモリコントローラ10は、別の電子回路(図示せず)との接続のための別のインターフェース14をも有する。メモリコントローラ10は、他のコンポーネントを有してもよいが、これらは、図を単純にするために図示されていない。

【0042】

メモリデバイス30-0、30-1、...、および30-Nのいくつかのコンポーネントは、同一の番号を付けられている。例えば、メモリデバイス30-0、30-1、...、および30-Nのそれぞれは、入力インターフェース21、出力インターフェース22、ならびにメモリ動作制御およびメモリ初期化機能を実行するメモリデバイス動作回路網23を有する。しかし、メモリデバイス30-0、30-1、...、および30-Nは、それぞれのメモリコア32-0、32-1、...、32-Nなどのそれぞれのメモリタイプ固有コンポーネントを有する。メモリデバイス30-0、30-1、...、および30-Nのそれぞれは、複数のサポートされるメモリタイプからのデバイスタイプを有する。複数のサポートされるメモリタイプは、実施態様固有の基礎で定義される。これは、固定されたものとしてすることができ、あるいは、いくつかの実施形態で、例えばメモリデバイスタイプを経時的に追加することによって、変更の対象にすることができる。所与の構成が、複数のサポートされるメモリデバイスタイプのそれぞれのメモリデバイスを必ずしも含まない可能性があるが、メモリコントローラ10およびインターフェースは、この機能性を可能にするように設計されている。複数のサポートされるメモリデバイスタイプに関する多数の可能性がある。

【0043】

複数のサポートされるメモリタイプは、例えば、NANDフラッシュEEPROM、NORフラッシュEEPROM、ANDフラッシュEEPROM、DiNORフラッシュEEPROM、シリアルフラッシュEEPROM、DRAM、SRAM、ROM、EPROM、FRAM、MRAM(磁気抵抗ランダムアクセスメモリ)、およびPCRAMのうちの任意の複数を含むことができる。サポートされるメモリタイプの他の組合せも可能である。

【 0 0 4 4 】

メモリデバイス30-0、30-1、...、および30-Nのそれぞれは、そのメモリタイプを知っている。これは、例えば、25によって示されるレジスタに格納することができるが、より一般的には、メモリデバイス30-0、30-1、...、および30-Nのそれぞれに、そのメモリデバイスタイプの識別を維持するための任意の適当な回路網を与えることができる。メモリデバイス30-0、30-1、...、および30-Nのそれぞれがそれによってそのデバイスタイプを知る、他の機構も、可能である。メモリデバイス30-0、30-1、...、および30-Nのそれぞれは、他のコンポーネントを有してもよいが、これらは、図を単純にするために図示されていない。

【 0 0 4 5 】

動作中に、メモリコントローラ10のコントローラ動作回路網13は、メモリ動作制御およびメモリ初期化機能を実行する。コントローラ動作回路網13は、出力インターフェース11を介してメモリコマンドを送る。各メモリコマンドのデータパスは、メモリデバイス30-0、30-1、...、および30-NとシリアルリンクL0、L1、L2、...、LNおよびL(N+1)との組合せによって与えられる。例えば、メモリコマンドが、第2メモリデバイス30-1宛である場合に、そのメモリコマンドは、シリアルリンクL0およびL1を介して第1メモリデバイス30-0をトラバースする。メモリコマンドが、第2メモリデバイス30-1からの応答を要請する場合には、その応答は、シリアルリンクL2、...、LN、およびL(N+1)を介してメモリコントローラ10に戻って第3メモリデバイス、...、第N(最後の)デバイス30-Nをトラバースする。

【 0 0 4 6 】

上で注記したように、メモリデバイス30-0、30-1、...、および30-Nのそれぞれは、複数のサポートされるデバイスタイプのうちの任意の1つとすることができる。メモリコントローラ10のコントローラ動作回路網13は、出力インターフェース11を介してデバイスタイプに固有のメモリコマンドを送り、入力インターフェース12を介してメモリ応答を要請するメモリコマンドに関するそのような応答を受け取るように動作可能である。例えば、コントローラ動作回路網13が、第2メモリデバイス30-1宛のコマンドを発行する場合に、発行されるコマンドは、第2メモリデバイス30-1のデバイスタイプに固有であり、このデバイスタイプは、他のメモリデバイスのデバイスタイプとは異なる可能性がある。メモリデバイス30-0、30-1、...、および30-Nのそれぞれは、メモリデバイスにアドレッシングされたこれらのメモリコマンドを実行し、出力インターフェース22を介して、別のメモリデバイスにアドレッシングされたメモリコマンドを転送する。メモリシステム40は、異なるデバイスタイプまたは混合されたメモリデバイスタイプに関してならびにメモリ密度拡張に関して、システムの全体的性能を犠牲にせずに適当に拡張することができる。

【 0 0 4 7 】

一般に、メモリシステム40は、2フェーズの動作すなわち、図2Cに示された35によって示される初期化フェーズおよび36によって示される通常動作フェーズを実行する。初期化フェーズ35では、メモリデバイス30-0、30-1、...、および30-Nは、デバイスアドレスを割り当てられる。割り当てられたデバイスアドレスは、メモリデバイス内に保持される。その後、通常動作フェーズ36で、ターゲットメモリデバイスすなわちアドレッシングされたメモリデバイスが、データアクセス動作を実行する。

【 0 0 4 8 】

メモリコントローラ10のコントローラ動作回路網13は、メモリデバイス30-0、30-1、...、および30-Nを制御するために出力インターフェース11を介してメモリコマンドを送る。これを達成できる多数の形がある。例のために、第1および第2の実施態様を下で説明するが、他の実施態様が可能である。

【 0 0 4 9 】

第1の実施態様では、コントローラ動作回路網13がメモリコマンドを送る。コマンドのそれぞれは、組み合わせられて複数のメモリデバイスの選択されたメモリデバイスを一意に識別する第1部分および第2部分を有する。いくつかの実施態様で、第1部分は、選択されたメモリデバイスのデバイスタイプを識別し、第2部分は、選択されたメモリデバイスの

10

20

30

40

50

デバイスアドレスを識別する。各メモリコマンドは、選択されたメモリデバイスによって実行される選択されたコマンドを識別するコマンド部分をも有する。各メモリコマンドは、適当に他の部分、例えば、さらなるアドレス情報およびデータをも含んでもよい。

【0050】

第1実施態様では、メモリデバイス、例えば第1メモリデバイス30-0が、メモリコマンドを受け取る時に、そのメモリデバイス動作回路網23が、そのメモリコマンドの第1部分および第2部分の組合せに应答して、そのメモリコマンドがそのメモリデバイス(例えば、第1メモリデバイス30-0)にアドレッシングされているかどうかを判定する。例えば、メモリデバイス動作回路網23は、先ず、デバイスタイプを示す第1部分を解釈する。第1部分によって示されるデバイスタイプが、デバイスタイプレジスタ25によって与えられる第1メモリデバイス30-0のデバイスタイプとは異なる場合に、そのデバイスのメモリデバイス動作回路網23は、受け取られたメモリコマンドのさらなる部分を調べる必要がない。したがって、受け取られたコマンドが他のメモリデバイス30-1、...、および30-Nのうちの1つにアドレッシングされていると判定される。したがって、メモリデバイス動作回路網23は、出力インターフェース22を介してメモリコマンドを転送する。しかし、第1部分によって示されるデバイスタイプが、第1メモリデバイス30-0のデバイスタイプと同一である場合には、メモリデバイス動作回路網23は、第2部分によって示されるデバイスアドレスが第1メモリデバイス30-0のデバイスアドレスと一致するかどうかを判定する。この2つのデバイスアドレスの間に一致がある場合には、メモリデバイス動作回路網23は、コマンド部分によって示される選択されたコマンドを実行する。そうでない場合には、メモリデバイス動作回路網23は、出力インターフェース22を介してメモリコマンドを転送する。

【0051】

第2実施態様では、メモリコントローラ10のコントローラ動作回路網13は、メモリコマンドを送ることによって複数のメモリデバイスを制御する。メモリコマンドのそれぞれは、複数のメモリデバイスの選択されたメモリデバイスを一意に識別する第1部分を含む。第1部分は、選択されたメモリデバイスのデバイスアドレスを識別する。第2実施態様では、メモリコマンドがデバイスタイプを含む必要がない。各メモリコマンドは、選択されたメモリデバイスによって実行される選択されたコマンドを識別するコマンド部分をも有する。前と同様に、各メモリコマンドは、適当に他の部分をも含んでもよい。第2実施態様では、メモリデバイス、例えば第1メモリデバイス30-0が、メモリコマンドを受け取る時に、そのメモリデバイス動作回路網23が、メモリコマンドの第1部分に应答して、そのメモリコマンドがそのデバイス(例えば、第1メモリデバイス30-0)にアドレッシングされているかどうかを判定する。メモリデバイス動作回路網23は、第1部分によって示されるデバイスアドレスが、第1メモリデバイス30-0のデバイスアドレスと一致するかどうかを判定する。この2つのデバイスアドレスの間に一致がある場合には、メモリデバイス動作回路網23は、コマンド部分によって示される選択されたコマンドを実行する。そうでない場合には、メモリデバイス動作回路網23は、出力インターフェース22を介してメモリコマンドを転送する。

【0052】

いくつかの実施態様で、メモリコントローラ10のコントローラ動作回路網13は、インターフェース14を介して受け取られる要求に应答して出力インターフェース11を介してメモリコマンドを送るように動作可能であり、さらに、入力インターフェース12を介して受け取られるメモリ応答を使用して要求に应答するように動作可能である。インターフェース14は、メモリシステム40を使用する別のデバイスまたはシステム(図示せず)への任意の適当なインターフェースとすることができる。

【0053】

メモリコマンドに関する多数の可能性がある。これらは、読取り動作、書込み動作、消去動作、状況読取り動作、ID読取り動作、構成レジスタ書込み動作、アドレス書込み動作、およびリセット動作のうちの1つまたは複数を含むことができる。これらを、他のメモリコマンドとすることができる。

【0054】

メモリコントローラ10がメモリコマンドを送る形は、デバイスアドレスが割り当てられる形に依存してもよい。デバイスアドレスの割当てに関する例の実施態様を、下で与える。

【0055】

デバイスアドレスをメモリデバイス30-0、30-1、...、および30-Nに割り当てる多数の形がある。いくつかの実施態様では、デバイスアドレスは、事前に決定されるかハードワイヤドである。他の実施態様では、コントローラ動作回路網13が、初期化ステージ中にデバイスアドレスを割り当てる。例のために、第1および第2の実施態様を下で説明するが、他の実施態様が可能である。

10

【0056】

第1実施態様では、コントローラ動作回路網13は、複数のサポートされるデバイスタイプのデバイスタイプごとに、出力インターフェース11を介して、そのデバイスタイプの各メモリデバイスにデバイスアドレスを割り当てるためにそれぞれの初期化メッセージを送るように動作可能である。メモリデバイス30-0、30-1、...、および30-Nのそれぞれは、この初期化メッセージを受け取り、処理する。例えば、第1メモリデバイス30-0は、第1入力インターフェース21を介して初期化メッセージを受け取る。受け取られる初期化メッセージごとに、その初期化メッセージが第1メモリデバイス30-0以外の別のデバイスタイプに関するものであることが示される場合には、メモリデバイス動作回路網23が、出力インターフェース22を介してその初期化メッセージを転送する。しかし、初期化メッセージが、第1メモリデバイス30-0のデバイスタイプに関するものであることが示される場合には、メモリデバイス動作回路網23は、初期化メッセージからデバイスアドレスを判定する。これは、初期化メッセージからデバイスアドレスを読み取ることを含む。いくつかの実施態様では、初期化メッセージから読み取られるデバイスアドレスが、第1メモリデバイス30-0のデバイスアドレスになる。メモリデバイス動作回路網23は、新しいデバイスアドレスと共に、出力インターフェース22を介して初期化メッセージを転送する。他のメモリデバイス30-1、...、および30-Nのそれぞれは、類似する初期化プロセスを実行する。送られたそれぞれの初期化メッセージごとに、コントローラ動作回路網13は、入力インターフェース12を介してそれぞれの初期応答を受け取り、そのデバイスタイプのメモリデバイスがない場合を除いて、この初期化応答から、そのデバイスタイプの各メモリデバイスのデバイスアドレスを判定することができる。

20

30

【0057】

例えば、いくつかの実施態様で、デバイスアドレスを割り当てられるメモリデバイスごとに、次のデバイスに転送される新しいデバイスアドレスは、デバイスアドレスの増分である。したがって、所与のデバイスタイプの第1メモリデバイスが、0のアドレスを割り当てられる場合に、所与のデバイスタイプの最後のメモリデバイスは、 $m-1$ のアドレスを割り当てられ、ここで、 m は、所与のデバイスタイプのメモリデバイスの個数である。所与のデバイスタイプの最後のメモリデバイスの増分されたデバイスアドレスを示す初期化応答を入力インターフェース12を介して受け取ることによって、コントローラ動作回路網13は、その所与のデバイスタイプの各メモリデバイスのデバイスアドレスを判定することができる。複数のデバイスタイプがある場合に、メモリコントローラ10が、どの物理デバイスがどのタイプであるかを知らないことに留意されたい。そうではなく、メモリコントローラ10は、各タイプのメモリデバイスが何個あるかを知る。例えば、4つのNANDデバイスおよび4つのNORデバイスがある場合がある。4つのNANDデバイスは、それぞれタイプ=NANDおよびアドレス=0、1、2、3を有し、4つのNORデバイスは、それぞれタイプ=NORおよびアドレス=0、1、2、3を有する。次に、NANDメモリデバイスおよびNORメモリデバイスの物理アドレスにかかわらず、タイプ部分およびアドレス部分を含むコマンドは、必ずターゲットデバイスを見つける。デバイス番号付けの2つの異なるセット(すなわち、デバイスタイプおよびデバイスアドレス)を実施することによって、メモリコントローラ10は、どのデバイスアドレスがメモリシステム40内でどのデバイスタイプに割り当てられているかを

40

50

考慮する必要がない。

【 0 0 5 8 】

第2実施態様では、メモリコントローラ10のコントローラ動作回路網13は、デバイスアドレスをメモリデバイス30-0、30-1、...、および30-Nに割り当てる初期化メッセージを出カインターフェース11を介して送るように動作可能である。メモリデバイス30-0、30-1、...、および30-Nは、その初期化メッセージを受け取り、処理する。例えば、第1メモリデバイス30-0は、入力インターフェース21を介して初期化メッセージを受け取り、そのメモリデバイス動作回路網23が、受け取られた初期化メッセージからデバイスアドレスを読み取る。いくつかの実施態様では、初期化メッセージから読み取られるデバイスアドレスが、第1メモリデバイス30-0のデバイスアドレスになる。メモリデバイス動作回路網23は、新しいデバイスアドレスと共に、出力インターフェース22を介して初期化メッセージを転送する。他のメモリデバイス30-1、...、および30-Nのそれぞれは、類似する初期化プロセスを実行する。最終的に、メモリコントローラ10のコントローラ動作回路網13は、入力インターフェース12を介して初期化応答を受け取り、この初期化応答から、各メモリデバイスのデバイスアドレスを判定することができる。

10

【 0 0 5 9 】

例えば、メモリデバイス30-0、30-1、...、および30-Nのそれぞれについて、次のデバイスに転送される新しいデバイスアドレスは、デバイスアドレスの増分である。したがって、第1メモリデバイス30-0が、0のアドレスを割り当てられる場合に、最後のメモリデバイス30-Nは、Nのアドレスを割り当てられ、ここで、メモリデバイスの個数は(N+1)である。最後のメモリデバイス30-Nの増分されたデバイスアドレスを示す初期化応答を入力インターフェース12を介して受け取ることによって、コントローラ動作回路網13は、各メモリデバイスのデバイスアドレスを判定することができる。

20

【 0 0 6 0 】

上で説明した第2実施態様によれば、メモリデバイス30-0、30-1、...、および30-Nのそれぞれのデバイスアドレスを割り当てたならば、コントローラ動作回路網13は、各メモリデバイスのデバイスタイプを判定する。デバイスアドレスごとに、コントローラ動作回路網13は、出力インターフェース11を介して、デバイスアドレスのメモリデバイスのデバイスタイプを判定する追加初期化メッセージを送る。メモリデバイス30-0、30-1、...、および30-Nのそれぞれは、その追加初期化メッセージを受け取り、処理する。

30

【 0 0 6 1 】

例えば、第1メモリデバイス30-0が、第1入力インターフェース21を介して追加初期化メッセージを受け取る時に、第1メモリデバイス30-0は、追加初期化メッセージで示されるデバイスアドレスに基づいて、その追加初期化メッセージが第1メモリデバイス30-0宛であるかどうかを判定する。そうである場合には、第1メモリデバイス30-0は、そのデバイスタイプの識別を用いて、出力インターフェース22を介して追加初期化メッセージに応答する。追加初期化メッセージが、他のメモリデバイス30-1、...、および30-Nのうちの1つ宛である場合には、第1メモリデバイス30-0は、追加初期化メッセージを出力インターフェース22を介して転送する。他のメモリデバイス30-1、...、および30-Nのそれぞれは、追加初期化メッセージの類似する処理を実行する。メモリコントローラ10のコントローラ動作回路網13は、メモリデバイスごとに、入力インターフェース12を介して、メモリデバイスのデバイスタイプを示す初期化応答を受け取る。

40

【 0 0 6 2 】

コントローラ動作回路網13について上で説明した第1および第2の実施態様が、例のみのために非常に特定であることを理解されたい。変形形態および修正形態が可能である。例えば、第1実施態様を、各メモリデバイスにデバイスアドレスを割り当てるために複数の初期化メッセージを用いるものとして上で説明したが、その代わりに、この目的の1つまたは複数の初期化メッセージがあってもよい。また、第2実施態様を、各メモリデバイスのデバイスタイプを判定するために複数の追加初期化メッセージを用いるものとして上で説明したが、その代わりに、この目的の1つまたは複数の追加初期化メッセージがあって

50

もよい。説明した例では、受け取られる初期化メッセージのアドレスが、デバイスアドレスとして確立され、新しいアドレスが、生成され、次のデバイスに送られる。もう1つの実施態様では、各メモリデバイスが、アドレスを受け取り、これをデバイスアドレスとして確立する前に増分する。この実施態様の詳細な例は、これによってその全体が参照によって本明細書に組み込まれる、本願と同一の譲受人に譲渡された同時係属の米国特許出願第11/529,293号、名称「Packet Based ID Generation for Serially Interconnected Devices」に記載されている。

【0063】

上で与えた例は、インターフェースに言及する。そのようなインターフェースに関する多数の可能性があることを理解されたい。特定のインターフェースが、下で説明する例で与えられる。より一般的には、任意の適当なインターフェースを実施してもよい。

10

【0064】

いくつかの実施態様で、メモリコントローラ10は、メモリデバイス30-0、30-1、...、および30-Nのそれぞれとの接続用のリセット出力(図示せず)を有する。この例は、下で説明する例で与える。より一般的に、メモリシステム40を、任意の適当なリセット実施態様を使用することによってリセットしてよい。

【0065】

いくつかの実施態様で、メモリコントローラ10は、メモリデバイス30-0、30-1、...、および30-Nのそれぞれとの接続用のシリアルクロック出力(図示せず)を有する。この例は、下で説明する例で与える。より一般的に、メモリシステム40に、任意の適当なクロック実施態様を使用することによってシリアルクロックを供給することができる。

20

【0066】

いくつかの実施態様で、メモリコントローラ10は、メモリデバイス30-0、30-1、...、および30-Nのそれぞれとの接続用のチップ選択(図示せず)を有する。この例は、下で説明する例で与える。より一般的に、メモリデバイス30-0、30-1、...、および30-Nを、任意の適当なデバイスイネーブル実施態様を使用することによってイネーブルすることができる。

【0067】

図2Cに、図2Bに示されたメモリシステムの動作を示す。図2Bおよび2Cを参照すると、メモリコントローラ10のコントローラ動作回路網13は、デバイス30-0、30-1、...、および30-Nを制御するために出力インターフェース11を介してメモリコマンドを送る。コントローラ動作回路網13は、メモリコマンドを送ることによってデバイス30-0、30-1、...、および30-Nを制御する。これを達成できる多数の形がある。例のために、いくつかの例の実施態様を下で説明するが、他の実施態様が可能である。

30

【0068】

一般に、メモリシステム40は、2フェーズの動作すなわち、35によって示される初期化フェーズおよび36によって示される通常動作フェーズを実行する。初期化フェーズ35(または初期化モード)では、デバイス30-0、30-1、...、および30-Nは、デバイスアドレスを割り当てられる。割り当てられたデバイスアドレスは、デバイス30-0、30-1、...、および30-N内に保持される。その後、通常動作フェーズ36(または通常動作モード)で、ターゲットメモリデバイスすなわちアドレスングされたメモリデバイスが、データアクセス動作を実行する。

40

【0069】

初期化フェーズ35の例で、コントローラ動作回路網13は、メモリコマンドを送る。コマンドは、デバイスに一意アドレスを割り当てるためのデバイスアドレス割当ての部分と、デバイスアドレス関連番号の部分とを有する。一実施態様で、コントローラ動作回路網13からのコマンドのデバイスアドレス関連番号は、初期値または初期番号であり、この初期番号は、デバイスのそれぞれによって増分される。増分された番号のそれぞれが、それぞれのデバイス内でそのデバイスアドレスとして保持される。

【0070】

50

通常動作フェーズ36の例では、コントローラ10のコントローラ動作回路網13は、メモリコマンドを送る。メモリコマンドは、デバイスアドレスと共に、複数のメモリデバイスのうちの選択されたメモリデバイスを一意に識別する第1部分を含む。デバイスタイプは、メモリコマンドには含まれない。各メモリコマンドは、選択されたメモリデバイスによって実行される選択されたコマンドを識別するコマンド部分をも有する。各メモリコマンドは、適宜、他の部分をも含んでよい。メモリデバイス、例えば第1デバイス30-0が、メモリコマンドを受け取る時に、そのデバイス動作回路網23は、メモリコマンドの第1部分に応答して、そのメモリコマンドがそのデバイス(すなわち、第1デバイス30-0)にアドレッシングされているかどうかを判定する。デバイス動作回路網23は、第1部分によって示されるデバイスアドレスが、第1デバイス30-0のデバイスアドレスと一致するかどうかを判定する。この2つのデバイスアドレスの間に一致がある場合には、デバイス30-0のデバイス動作回路網23は、コマンド部分によって示される選択されたコマンドを実行する。そうでない場合には、デバイス動作回路網23は、出力インターフェース22を介してメモリコマンドを次のデバイス(例えば、第2デバイス30-1)に転送する。

10

【0071】

いくつかの実施態様で、コントローラ10のコントローラ動作回路網13は、インターフェース14を介して受け取られる要求に応答して出力インターフェース11を介してメモリコマンドを送るように動作可能であり、入力インターフェース12を介して受け取られるメモリ応答を使用して要求に応答するようにさらに動作可能である。インターフェース14は、メモリシステム40を使用する別のデバイスまたはシステム(図示せず)への任意の適当なインターフェースとすることができる。

20

【0072】

メモリコマンドに関する多数の可能性がある。これは、読取り動作、書込み動作、消去動作、状況読取り動作、DA読取り動作、構成レジスタ書込み動作、アドレス書込み動作、およびリセット動作のうちの1つまたは複数を含むことができる。これらを、他のメモリコマンドとすることができる。

【0073】

コントローラ10がメモリコマンドを送る形は、デバイスアドレスが割り当てられる形に依存してよい。デバイスアドレスを割り当てる例の実施態様を、下記で説明する。

【0074】

30

次の説明および図面では、いくつかの符号が、信号および接続に使用される。例えば、「SCLK」は、メモリデバイスのクロック信号およびクロック入力接続を表し、「SIP」は、シリアル入力ポート信号およびシリアル入力ポート接続を表し、「SOP」は、シリアル出力ポート信号およびシリアル出力ポート接続を表し、「IPE」は、入力ポートイネーブル信号および入力ポートイネーブル接続を表し、「OPE」は、出力ポートイネーブル信号および出力ポートイネーブル接続を表し、「CS#」は、チップ選択信号およびチップ選択入力接続またはチップ選択入力ポートを表し、「RST#」は、リセット信号およびリセット入力信号またはリセット入力ポートを表す。また、同一の符号が、同一のまたは対応するブロック、接続、信号、および回路網に使用される。

【0075】

40

図3A、3B、3C、3D、3E、および3Fは、本発明の実施形態による特定の例のメモリシステムを示す。これらの図が、非常に特定であり、例としてのみ提供されることを理解されたい。

【0076】

図3Aに、例のメモリシステムの全般的構成を示す。メモリシステム41は、メモリコントローラ50および複数($n+1$)個のメモリデバイスを含み、 n は、整数である。この特定の例では、メモリコントローラ50およびメモリデバイスは、シリアルリンクを用いて接続される。シリアル相互接続構成は、第1デバイス80(「デバイス0」)、第2デバイス81(「デバイス1」)、第3デバイス82(「デバイス2」)、...、および第($n+1$)デバイス83(「デバイス n 」)を含む。

50

【 0 0 7 7 】

図3Aを参照すると、メモリコントローラ50は、メモリデバイス80、81、82、...、および83のそれぞれに接続された、リセットポート51、チップ選択ポート52、およびシリアルクロックポート53を有する。したがって、メモリデバイス80、81、82、...、および83のそれぞれは、リセットポート61、チップ選択ポート62、およびシリアルクロックポート63を有する。メモリコントローラ50は、第1メモリデバイス80に接続された、シリアル出力54、入力イネーブル55、および出力イネーブル56を含む出力インターフェースを有する。したがって、第1メモリデバイス80は、シリアル入力64、入力イネーブル65、および出力イネーブル66を含む入力インターフェースを有する。第1メモリデバイス80は、シリアル出力67、入力イネーブルエコー68、および出力イネーブルエコー69を含む出力インターフェースをも有する。他のメモリデバイス81、82、および83のそれぞれは、対応する入力インターフェース64、65、および66ならびに出力インターフェース67、68、および69を有し、その結果、メモリデバイス80、81、82、...、および83が、シリアルリンクを介して相互接続されるようになる。メモリコントローラ50は、最後のメモリデバイス83の出力インターフェース67、68、および69との接続のためのシリアル入力57、入力イネーブルエコー58、および出力イネーブルエコー59を含む入力インターフェースを有する。

10

【 0 0 7 8 】

メモリコントローラ50は、図2Bに示されたメモリコントローラのコンポーネントに類似するコンポーネント(図示せず)を有するが、これらは、図を単純にするために図示されていない。

20

【 0 0 7 9 】

メモリデバイス80、81、82、...、および83は、それぞれメモリ80A、81A、82A、...、および83Aなど、など、メモリタイプ固有コンポーネントを有する。しかし、図示の例では、これらのデバイスタイプが、指定されない。メモリデバイス80、81、82、...、および83のそれぞれは、そのインターフェースとそのメモリとの間のインターフェース回路(図示せず)を有する。メモリデバイス80、81、82、...、および83のそれぞれは、そのデバイスタイプの識別を維持するレジスタ60をも有する。他の実施態様では、メモリデバイス80、81、82、...、および83のそれぞれは、そのデバイスタイプの識別を維持する代替回路網を有する。メモリデバイス80、81、82、...、および83のそれぞれは、他のコンポーネントを有してよいが、これらは、図を単純にするために図示されていない。

30

【 0 0 8 0 】

動作中に、メモリシステム41は、上で図2Bを参照して説明したメモリシステム40に似た形で動作する。しかし、説明のために、メモリシステム41の動作のさらなる例の詳細を、追加の図面を参照して以下で説明する。

【 0 0 8 1 】

図3Bに、第1の特定の例のメモリシステム42を示す。メモリシステム42は、図3Bに示されたメモリシステム41に類似する。メモリシステム42は、図3Bの例と同一であるメモリコア84A、85A、86A、87Aを有する(n+1)個のメモリデバイス84、85、86、...、および87を含む。図3Bに示された特定の例では、第1メモリデバイス84が、NORフラッシュメモリコア84Aを有し、第2、第3、...、および第(n+1)メモリデバイス85、86、...、および87は、それぞれNANDフラッシュメモリコア85A、86A、...、および87Aを有する。図3Cの例は、タイプごとのアドレッシング方式が使用される、すなわち、前に導入した第1アドレッシング方式が使用されるという点で、図3Bの例とは異なる。タイプおよびアドレスは、1つのNORデバイスおよび「n」個のNANDデバイスがあると仮定して、「NOR-0」、「NAND-0」、「NAND-1」、...、および「NAND-(n-1)」として示される。

40

【 0 0 8 2 】

上で注記したように、メモリデバイス84、85、86、...87は、任意の適当なデバイスタイプであるものとすることができる。この点を例示するために、さまざまなまたは混合されたデバイスタイプを有する追加の例のメモリシステムを、図3D、3E、および3Fを参照して示す。

50

【 0 0 8 3 】

図3Cに、第2の特定の例のメモリシステム43を示す。メモリシステム43は、メモリシステム43が異なるメモリデバイスを有することを除いて、図3Cに示されたメモリシステム42と同一である。図3Cに示された特定の例では、メモリシステム43は、それぞれ混合されたメモリコア88A、89A、90A、...、および91Aを有する複数($n+1$)個のメモリデバイス88、89、90、...、および91を含む。第1メモリデバイス88は、SRAMメモリコア88Aを有し、第2メモリデバイス89は、NORフラッシュメモリコア89Aを有する。第3、...、および第($n+1$)メモリデバイス91は、それぞれNANDフラッシュメモリ90A、...、および91Aを有する。メモリデバイスは、タイプごとのアドレッシング方式を使用してアドレッシングされる。タイプおよびアドレスは、1つのSRAMデバイス、1つのNORデバイス、および「($n-1$)」個のNANDデバイスがあると仮定して、「SRAM-0」、「NOR-0」、「NAND-0」、...、および「NAND-($n-2$)」として示される。

10

【 0 0 8 4 】

図3Dは、複数($n+1$)個のメモリデバイスを示す第3の特定の例のメモリシステム44を示す図である。図3Dを参照すると、メモリシステム44は、メモリシステム44が異なるメモリデバイス92、93、...、94、および95を有することを除いて、図3Cに示されたメモリシステム42と同一である。図3Dに示された詳細では、メモリデバイス92、93、...、94、および95のメモリコア92A、93A、94A、...および95Aが混合されている。図示の例では、第1、第2、第3、...、および第 n のメモリデバイス92、93、...、および94が、それぞれNANDフラッシュメモリコア92A、93A、...、および94Aを有する。最後の($n+1$)番目のメモリデバイス95は、NORフラッシュメモリコア95Aを有する。メモリデバイスは、タイプごとのアドレッシング方式を使用してアドレッシングされる。タイプおよびアドレスは、「 n 」個のNANDデバイスおよび1つのNORデバイスがあると仮定して、「NAND-0」、「NAND-1」、...「NAND-($n-1$)」、および「NOR-0」として示される。メモリコントローラ50は、図3Bの例と図3Dの例との間の物理的レイアウトの差を知らないことを留意されたい。

20

【 0 0 8 5 】

図3Eに、第4の特定の例のメモリシステム45を示す。メモリシステム45は、メモリシステム45が異なるメモリデバイス96、97、98、...、および99を有することを除いて、図3Cに示されたメモリシステム42と同一である。図3Eに示された特定の例では、メモリデバイス96、97、98、...、および99のメモリコア96A、97A、98A、...および99Aが、混合される。図示の例では、第1メモリデバイス96が、NANDフラッシュメモリコア96Aを有する。第2メモリデバイス97は、NORフラッシュメモリコア97Aを有する。第3、...、および最後の第($n+1$)メモリデバイス98、...、および99は、それぞれNANDフラッシュメモリコア98A、...、および99Aを有する。メモリデバイスは、タイプごとのアドレッシング方式を使用してアドレッシングされる。タイプおよびアドレスは、「 n 」個のNANDデバイスおよび1個のNORデバイスがあると仮定して、「NAND-0」、「NOR-0」、「NAND-1」、...、および「NAND-($n-1$)」として示される。メモリコントローラ50が、図3B～3Eの例の間での物理レイアウトの相違を知らないことに留意されたい。

30

【 0 0 8 6 】

図3B～3Eの4つの例が、メモリコントローラ50が少なくともNORフラッシュデバイス、NANDフラッシュデバイス、およびSRAMデバイスと相互作用できると仮定して、同一の回路レイアウト、同一のメモリコントローラ50、およびメモリデバイスの「ロット」または「ソケット」を用いて実施できることがわかる。次に、サポートされるデバイスタイプの任意の配置を、「ロット」または「ソケット」にインストールすることができ、図3B、3C、3D、および3Eは、これのそれぞれの例である。

40

【 0 0 8 7 】

図3A～3Eに示された例のそれぞれで、リセット信号、チップ選択信号、およびシリアルクロック信号は、マルチドロップの形で供給される。図3Fに示されたもう1つの実施態様では、シリアルクロックは、出力エコークロック信号「SCLK_0」の追加を伴って、ポイントツーポイントリングタイプスキームで接続される。SCLKは、メモリコントローラ50Aな

50

らびにメモリデバイス180、181、182、...および183を同期化するシステムクロックである。各メモリデバイスからのエコークロック信号出力は、次のメモリデバイスのクロック入力SCLKに供給される。メモリコントローラ50Aならびにメモリデバイス180、181、182、...および183は、それぞれマスタデバイスおよびスレーブデバイスとして動作する。割り当てられるアドレスは、 $(n+1)$ 個のメモリデバイスがあると仮定して、「デバイス0」、「デバイス1」、「デバイス2」、...、「デバイス n 」として示されている。

【0088】

提示された例では、クロッキングは、SDR (シングルデータレート)に基づくが、他の適当なクロッキング方式を企図できることを理解されたい。他の適当なクロッキング方式は、例えば、DDR (ダブルデータレート)、QDR (クワッドデータレート)、立ち上がりエッジ SDR、または立ち下がりエッジSDRを含むことができる。企図できる他の適当なクロッキング方式もあり得る。

【0089】

図3Gは、メモリデバイスの例のSDR動作の相対タイミングシーケンスを示す。図3Gは、1ポートでの動作を示す。図3Aおよび図3Gを参照すると、動作は、デバイス80、81、82、...および83に転送される情報を、デバイスのシリアルクロックポート63に供給されるクロック信号SCLKの異なる時刻に取り込むことができるというものである。SDR実施態様の例では、デバイスのうちの1つにそのシリアル入力64で供給される情報を、クロック信号SCLKの立ち上がりエッジで取り込むことができる。SDR動作では、チップ選択信号が、同時にすべてのデバイスをイネーブルするように共通して接続され、その結果、第1デバイスの入力データが、シリアル相互接続構成を介して転送可能になる。代替案では、SDR動作で、SIP接続でデバイスに供給される情報を、クロック信号SCLKの立ち下がりエッジで取り込むことができる。

【0090】

図3Hは、メモリデバイスの例のDDR動作の相対タイミングシーケンスを示す。図3Hは、1つのポートでの動作を示す。DDR動作では、クロック信号SCLKの立ち上がりエッジと立ち下がりエッジとの両方を使用して、シリアル入力64に供給される情報を取り込むことができる。

【0091】

図4Aに、図3A～3Eに示されたメモリデバイスとして使用される例のメモリデバイスブロックを示す。図4Aを参照すると、メモリデバイス140Aは、メモリデバイスのうちの任意の1つを表し、デバイスコントローラ/プロセッサ142A、デバイスタイプ一致デターミナ143、メモリ144、デバイスタイプレジスタ146、アドレス一致デターミナ147、デバイスアドレスレジスタ148、およびアドレス増分オペレータ149を含む。デバイスコントローラ/プロセッサ142Aは、メモリデバイス140Aの動作を制御する。メモリ144は、例えば、NANDフラッシュメモリ、NORフラッシュメモリ、SRAM、DRAMなど、任意のタイプのメモリを含む。デバイスタイプレジスタ146は、図3A～3Eに示したそのデバイスタイプの識別を維持するレジスタ60を含む。デバイスアドレスレジスタ148は、そのメモリデバイス140Aのデバイスコントローラ/プロセッサ142Aによって割り当てられたデバイスアドレス(DA)を保持する。デバイスタイプレジスタ146の詳細を、図5Aに示す。デバイスタイプ一致デターミナ143およびアドレス一致デターミナ147は、デバイスコントローラ/プロセッサ142Aによる制御の下で関連する一致判定機能を実行する。アドレス増分オペレータ149は、デバイスアドレス増分(すなわち、「DA+1」)の機能を実行する。

【0092】

デバイス140Aは、メモリコントローラ(例えば、図3Aに示されたメモリコントローラ50)に結合された、リセットポート「RST#」、チップ選択ポート「CS#」、およびシリアルクロックポート「SCLK」を有する。デバイスコントローラ/プロセッサ142Aは、前のメモリデバイスまたはメモリコントローラに接続された、そのメモリデバイスのシリアル入力「SIP」、入力イネーブル「IPE」、および出力イネーブル「OPE」に接続される。また、デバイスコントローラ/プロセッサ142Aは、次のメモリデバイスに接続された、そのメモリ

デバイスのシリアル出力「SOP」、入力イネーブルエコー「IPEQ」、および出力イネーブルエコー「OPEQ」に接続される。メモリ144は、フラッシュメモリコアに対応する。デバイスタイプレジスタ146は、そのデバイスタイプの表示を維持するタイプレジスタ60に対応する。

【0093】

メモリコントローラによって発行されるメモリコマンドの例のフォーマットは、次のとおりである。

【0094】

【表1】

メモリコマンド(1)

TYPE (aah)	TDA (bbh)	CMD (cch)	DATA (ddh)
---------------	--------------	--------------	---------------

【0095】

TYPEは、特定のメモリデバイスタイプの識別のためのデバイスタイプである。TDAは、特定のメモリデバイスのアドレスの識別のためのターゲットデバイスアドレスである。CMDは、ターゲットメモリデバイスによって実行される動作コマンドである。DATAは、メモリデバイスのプロセスまたは制御に対する情報(番号または値)を含む。さまざまな動作コマンドCMDの例を、表1に示す。

【0096】

【表2】

表1

動作	コマンド(1バイト)
ページ読取り	00h
ランダムデータ読取り	05h
ページプログラム	10h
チップ消去	20h
セクタ消去	21h
コピーのためのページ読取り	35h
デバイスアドレス書込み	39h
ブロック消去	60h
状況読取り	70h
シリアルデータ入力(バッファへの書込み)	80h
ランダムデータ入力	85h
コピーのためのターゲットアドレス入力	8Fh
デバイスタイプ読取り	90h
構成レジスタ書込み	A0h
プログラム/消去一時停止	C0h
プログラム/消去再開	D0h
リセット	FFh

【0097】

もう一度図4Aを参照すると、デバイスコントローラ/プロセッサ142Aは、シリアル入力(SI)に含まれるデバイスタイプおよびデバイスアドレスに応答して、メモリコマンドがそのメモリデバイス140Aにアドレッシングされているかどうかを判定する。例えば、デバイスタイプ一致デターミナ143は、デバイスコントローラ/プロセッサ142Aによる制御の下で、SI内のデバイスタイプ(「DTs」)がデバイスタイプレジスタ146に保持されたデバイスタイプ(「DT_r」)と一致するかどうかを判定する。これらの間の一致の場合に、デバイスタイプ一致デターミナ143は、タイプ一致表示143Mを、デバイスコントローラ/プロセッサ142Aに供給する。次に、アドレス一致デターミナ147は、デバイスコントローラ/プロセッサ

142Aによる制御の下で、SIに含まれるデバイスアドレス(「DAs」)がデバイスアドレスレジスタ148に保持されたデバイスアドレス(「DAr」)と一致するかどうかを判定する。これらの間の一致の場合に、アドレス一致デターミナ147は、アドレス一致表示147Mをデバイスコントローラ/プロセッサ142Aに供給する。

【0098】

図2Cに示された初期化フェーズ35の動作で、タイプ一致表示143Mおよびアドレス一致表示147Mにตอบสนองして、デバイスコントローラ/プロセッサ142Aは、SIに含まれるデバイスアドレス(DA)を、アドレス増分動作149に供給し、アドレス増分動作149は、「+1」の計算を実行する。したがって、計算されたまたは増分されたアドレス(DA+1)が、デバイスコントローラ/プロセッサ142Aに出力される。増分されたデバイスアドレスは、SOPを介して次のデバイスに供給される。タイプ一致表示143Mおよびアドレス一致表示147Mのどちらもが供給されない場合には、デバイスコントローラ/プロセッサ142Aは、コマンドをSOPを介して次のデバイスに転送する。

【0099】

図2Cに示されたデータアクセスフェーズ36の通常動作では、タイプ一致表示143Mおよびアドレス一致表示147Mにตอบสนองして、デバイスコントローラ/プロセッサ142Aが、SIに含まれる受け取られたコマンドを実行する。タイプ一致表示143Mおよびアドレス一致表示147Mのどちらもが供給されない場合には、デバイスコントローラ/プロセッサ142Aは、SOPを介してコマンドを次のデバイスに転送する。

【0100】

図4Bに、図3Fに示されたメモリデバイスとして使用される例のメモリデバイスを示す。図4Bに示されたメモリデバイス140Bは、図3Fに示されたメモリデバイスのうちの任意の1つを表す。メモリデバイス140Bは、図4Aに示されたメモリデバイス140Aに類似する。メモリデバイス140Bのデバイスコントローラ/プロセッサ142Bは、それに供給される入力クロックと同期化される出力クロックを出力するクロックシンクロナイザ191を含む。クロックシンクロナイザ191は、前のメモリデバイスから入力されるクロック信号SCLKの出力エコークロック信号SCLK_0をもたす位相ロックループ(PLL)または遅延ロックループ(DLL)を含むことができる。デバイス140Bの他の動作は、図4Aで示したデバイス140Aの動作と同一である。

【0101】

図3A～3Fを参照して上で提示した例では、メモリデバイスのそれぞれが、レジスタ(例えば、デバイスタイプレジスタ146)を有する。フラッシュメモリ(例えば、NANDフラッシュメモリ、NORフラッシュメモリなど)は、例えば製造業者コード、メモリ密度、ページサイズ、ブロックサイズ、バンク数、入出力構成、もしくは任意のクリティカルなAC/DC特性など、有用な情報を識別するために、フラッシュセルコアアレイの余分のセクションを利用するデバイスの内部の工場でプログラムされるレジスタを含む。しかし、上で注記したように、いくつかの実施態様で、レジスタは、デバイスタイプの識別を維持するのに使用される。あるレジスタがデバイスタイプを示す多数の形がある。例を、図5Aを参照して下で与える。

【0102】

図5Aを参照すると、例のレジスタブロック120は、物理的ハードプログラマブルレジスタユニットの一種であるタイプレジスタを有する。レジスタブロック120は、eFuse (electrically programmable fuse)アレイ121およびeFuseレベル検出論理ユニット122を有する。図示の例では、eFuseアレイ121は、ビット7、6、5、...、1、および0の8ビット構成を有して図示されている。この特定の例では、最初の4つのビット7～4は、「0000」であり、第2の4つのビット3～0は「0111」である。これは、例えば、「07h」(=00000111)を表す。特定の实施態様で、この構成は、PCRAMメモリタイプを示す。異なる構成が、異なるデバイスタイプを示すことができる。図示の例では、「閉」および「開」のヒューズが、それぞれ「0」および「1」を示す。そのような「0」および「1」の論理は、eFuseレベル検出論理ユニット122によって検出され、検出されたビット状況(ビット7～0)が、図4Aに示さ

10

20

30

40

50

れたデバイスコントローラ/プロセッサ142Aに供給される。代替案では、レジスタブロック120が、従来のポリヒューズまたは金属ヒューズ、OTP (One Time Programmableメモリ)、あるいは任意の不揮発性プログラマブルコンポーネントを有することができる。

【0103】

図5Bに、各デバイスタイプの例の符号化方式の表を示す。この表では、「RFU」が、「将来の使用に予約済み」を意味する。図5Bを参照すると、この表は、10個のメモリタイプすなわち、NANDフラッシュ、NORフラッシュ、DRAM、SRAM、PSRAM、DiNORフラッシュ、FeRAM、PCRAM、シリアルEEPROM、およびMRAMのそれぞれの符号化方式を定義する。例えば、SRAMメモリタイプは、「03h」の符号化方式を有する。NANDフラッシュタイプは、「00h」を割り当てられ、NORフラッシュタイプは、「01h」を割り当てられる。この例では、ビット構造は、MSB(最上位ビット)からLSB(最下位ビット)へである。他の実施態様では、これを順序において逆転することができ、これは、最初に、MSBではなくLSBから始まる。一部のレジスタ構成は、将来の使用のために予約済み(RFU)である。

10

【0104】

図2Cに示された初期化フェーズ35を実施する多数の形がある。上で注記したように、各メモリデバイスは、メモリタイプを有する。タイプ依存アドレッシングを用いてデバイスアドレスを割り当てるプロセスを、下で図6Aを参照して説明する。

【0105】

図6Aは、タイプ依存アドレッシングを用いてデバイスアドレスを割り当てる方法を示す。このプロセスが、例のみのために非常に固有であることを理解されたい。この方法は、複数のメモリデバイスがシリアル相互接続される任意のメモリシステム(例えば、図3Bに示されたメモリシステム)に適用可能である。

20

【0106】

図3Bおよび6Aを参照すると、パワーアップ初期化がある時(ステップ6-1)に、メモリコントローラ50は、デバイスタイプ「m」のメモリデバイスのデバイスアドレス書込み動作を実行する(ステップ6-2)。デバイスアドレス書込み動作は、すべてのメモリデバイスがパワーアップ中に当初に「00h」にセットされたデバイスアドレスを有するので、「00h」のターゲットデバイスアドレス(TDA)を有する。デバイスアドレス書込み動作が、各メモリデバイスをトラバースする時に、そのターゲットデバイスアドレスは、「00h」のままになり、その結果、各メモリデバイスは、デバイスアドレス書込み動作を処理ようになる。デバイスアドレス書込み動作は、メモリデバイスのそれぞれをトラバースする。デバイスタイプ「m」の各メモリデバイスは、デバイスアドレス書込み動作によって示されるデバイスアドレスに基づいて、そのデバイスアドレスを割り当てられる。そのデバイスアドレスを割り当てられる各メモリデバイスは、デバイスアドレス書込み動作を次のメモリデバイスに転送する前に、それによって示されるデバイスアドレスを増分する。

30

【0107】

最終的に、デバイスアドレス書込み動作は、メモリコントローラ50に戻る。デバイスアドレス書込み動作が、信号入力イネーブルエコー(IPEQ)およびシリアル入力(SIP)によって示されるように待機(ステップ6-4)を伴ってメモリコントローラ50に戻って到着する(ステップ6-3のYES)場合に、メモリコントローラ50は、デバイスタイプ「m」のメモリデバイスの個数がデバイスアドレス書込み動作によって示されるデバイスアドレス(「NA」)と等しくなると判定し、メモリコントローラ50は、「m」を増分する(ステップ6-5)。その後、メモリコントローラ50は、追加のデバイスタイプがあるかどうかを「m」の値に基づいて判定する(ステップ6-6)。追加のデバイスタイプがある場合に(ステップ6-6のYES)、動作は、ステップ6-2に戻る。追加のデバイスタイプごとに、メモリコントローラ50は、ステップ6-2から6-5までを繰り返す。メモリコントローラ50は、「m」のすべての可能な値に対応するすべての可能なデバイスタイプのメモリデバイスにデバイスアドレスを割り当てることを試みる。これは、メモリコントローラ50が、どのデバイスタイプがメモリシステムに存在するかを前もって知らない場合があるので、実行される。追加のデバイスタイプがもうない場合に(ステップ6-6のNO)、プロセスは、終了する(ステップ6-7)。

40

50

【 0 1 0 8 】

上で述べたステップ6-2の詳細を、図6Bに示す。図3B、4A、6A、および6Bを参照すると、デバイスアドレス書込み動作を受け取ったデバイスは、受け取ったデバイスタイプ「m」がデバイスタイプレジスタ146に登録されたそのデバイスタイプと一致するかどうかを判定する(ステップ6-8)。これは、デバイスコントローラ/プロセッサ142Aおよびデバイスタイプ一致デターミナ143によって実行される。デバイスタイプ一致がある(ステップ6-8のYES)場合には、デバイスは、ターゲットデバイスアドレス(TDA)がデバイスアドレスレジスタ148に登録されたデバイスアドレスと一致するかどうかをさらに判定する(ステップ6-9)。これは、デバイスコントローラ/プロセッサ142Aおよびアドレス一致デターミナ147によって実行される。デバイスアドレス一致がある(ステップ6-9のYES)場合には、受け取られたデバイスアドレスが、デバイスアドレスレジスタ148に登録され(ステップ6-10)、受け取られたデバイスアドレスが増分される(DA+1)(ステップ6-11)。そのようなデバイスアドレス増分は、デバイスコントローラ/プロセッサ142Aおよびアドレス増分オペレータ149によって実行される。デバイスタイプ一致がない(ステップ6-8のNO)場合には、デバイスアドレス割当てもデバイスアドレス増分も実行されない。また、デバイスアドレス一致がない(ステップ6-9のNO)場合には、デバイスアドレス割当てもアドレス増分も実行されない。

10

【 0 1 0 9 】

図6Aおよび6Bを参照して上で説明したプロセスのさらなる説明を提供するために、タイミング図を、図7A、7B、7C、および7Dを参照して下で説明する。

20

【 0 1 1 0 】

図7A、7B、7C、および7Dに、タイプ依存アドレッシングを用いるデバイスアドレスの割当てに関する信号のタイミングシーケンスを示す。このタイミング図は、相互接続された1つのNORタイプフラッシュデバイスおよび「n」個のNANDタイプフラッシュデバイスを有するメモリシステムから生じる可能性がある例の信号を示す。この種類のメモリシステムは、図3Dのメモリシステム42に類似する。

【 0 1 1 1 】

上で説明したように、メモリコントローラによって発行されるメモリコマンドは、フォーマットされる。例えば、このメモリシステムでは、NANDフラッシュデバイスだけが、「0」からのデバイスアドレスを割り当てられ、メモリコントローラによって発行されるメモリコマンドは、次のとおりである。

30

【 0 1 1 2 】

【表 3】

メモリコマンド(2)			
TYPE (00h)	TDA (00h)	CMD (39h)	DATA (00h)

【 0 1 1 3 】

このメモリコマンドにおいて、

TYPE (00h)は、「NANDフラッシュ」デバイスを識別する(図5Bを参照されたい)。

40

TDA (00h)は、初期化動作が実行される時にデバイスアドレス「0」を保持するデバイスを識別する。シリアル相互接続構成のすべてのメモリデバイスが、「0」にリセットされていると仮定する。

CMD (39h)は、実行される動作が「デバイスアドレス書込み」であることを識別する(表1を参照されたい)。

DATA (00h)は、デバイスアドレスの初期番号が「0」であることを識別する。

【 0 1 1 4 】

図7A~7Dを参照すると、このタイミング図の最上部には、7-1によって示される電源の信号(VDD)がある。このタイミング図は、それぞれ7-2および7-3によって示される、デバイス_0の入力イネーブル(IPE)およびシリアル入力(SIP)の信号を含む。このタイミング図

50

は、それぞれ7-4および7-5によって示される、デバイス_1の入力イネーブル(IPE_1)およびシリアル入力(SIP_1)の信号を含む。このタイミング図は、それぞれ7-6および7-7によって示される、デバイス_2の入力イネーブル(IPE_2)およびシリアル入力(SIP_2)の信号を含む。このタイミング図は、それぞれ7-8および7-9によって示される、デバイス_(n-1)の入力イネーブル(IPE_(n-1))およびシリアル入力(SIP_(n-1))の信号を含む。このタイミング図は、それぞれ7-10および7-11によって示される、デバイス_nの入力イネーブル(IPE_n)およびシリアル入力(SIP_n)の信号を含む。最後に、このタイミング図は、それぞれ7-12および7-13によって示される、メモリシステムのシリアル相互接続構成の最後のメモリデバイスであるデバイス_nの入力イネーブルエコー(IPEQ)およびシリアル出力(SOP)の信号を含む。

10

【 0 1 1 5 】

例えば、この種類のメモリシステムが、図3Dのメモリシステム42に適用される場合に、デバイス_0、デバイス_1、...、デバイス_(n-1)、およびデバイス_nは、それぞれメモリデバイス92、93、...、94、および95に対応する。

【 0 1 1 6 】

図3Dおよび7A～7Dを参照すると、メモリシステムは、VDD 7-1がハイ状態に遷移することによって示されるように、パワーオンされる。そのすぐ後に、第1のデバイスアドレス書込み動作7-14が、メモリコントローラ50によって発行される。第1のデバイスアドレス書込み動作7-14は、NANDフラッシュのデバイスタイプを示す。第1のデバイスアドレス書込み動作7-14は、各メモリデバイスをトラバースする。各NANDフラッシュメモリデバイスは、デバイスアドレス書込み動作によって示されるデバイスアドレスに基づいてそのデバイスアドレスを割り当てられる。そのデバイスアドレスを割り当てられる各メモリデバイスは、第1のデバイスアドレス書込み動作を次のメモリデバイスに転送する前に、それによって示されるデバイスアドレスを増分する。「n」個のNANDフラッシュメモリデバイスに対応する、合計「n」回の増分がある。最後のメモリデバイスによる増分はない。というのは、このメモリデバイスが、NANDフラッシュデバイスではなく、NORフラッシュデバイスであるからである。第1のデバイスアドレス書込み動作7-14は、メモリコントローラに戻る。メモリコントローラは、NANDフラッシュデバイスの個数が、第1のデバイスアドレス書込み動作によって示される「n」と等しいと判定する。

20

【 0 1 1 7 】

メモリコントローラ50は、追加のデバイスタイプごとに、追加のデバイスアドレス書込み動作を発行する。NORフラッシュのデバイスタイプを示す第2のデバイスアドレス書込み動作7-15が、発行される。NORフラッシュデバイスのデバイスアドレス割当てのためのメモリコマンドは、次のとおりである。

30

【 0 1 1 8 】

【表 4】

メモリコマンド(3)			
TYPE (01h)	TDA (00h)	CMD (39h)	DATA (00h)

40

【 0 1 1 9 】

第2のデバイスアドレス書込み動作7-15は、各メモリデバイスをトラバースする。NANDフラッシュデバイスのどれもが、第2のデバイスアドレス書込み動作によって示されるデバイスアドレスを増分しない。NORデバイスである最後のメモリデバイスは、「00h」としてそのデバイスアドレス(TDA)を割り当てられる。最後のメモリデバイスは、第2のデバイスアドレス書込み動作をメモリコントローラに転送する前に、それによって示されるデバイスアドレスを増分もする。第2のデバイスアドレス書込み動作7-15を受け取る時に、メモリコントローラは、第2のデバイスアドレス書込み動作によって示されるデバイスアドレスに基づいて、1つのNORデバイスがあると判定する。

【 0 1 2 0 】

50

追加のデバイスアドレス書込み動作を、メモリコントローラによって発行することができるが、これは、図を単純にするために図示されていない。例えば、SRAMがデバイスアドレスを割り当てられる場合に、メモリコントローラによって発行されるメモリコマンドのTYPEは、「03h」になる(図5Bを参照されたい)。

【0121】

シリアル相互接続構成のデバイスは、メモリコントローラによって発行されるコマンドに応答して動作を実行する。

【0122】

図3D、4A、および図7A~7Dを参照すると、メモリコントローラ50は、NANDフラッシュのデバイスタイプに関する第1のデバイスアドレス書込み動作を発行する。IPEがハイである間に、SIPに含まれるTYPE (NANDフラッシュ)、TDA (00h)、CMD (39h)、およびDATA (00h)が、第1デバイス92(すなわち、デバイス_0)に供給される。CMD (39h)は、デバイス92のデバイスコントローラ/プロセッサ142Aに、「デバイスアドレス書込み」動作を実行させる。SIPのTYPE (DTs)とデバイスタイプレジスタ146に保持されたデバイスタイプ(DTr)との両方が、NANDフラッシュであり、したがって、デバイスタイプ一致デターミナ143は、デバイスタイプ一致結果(すなわち、タイプ一致表示143M)をもたらす。また、TDAは、「00h」であり、(DAs)は、デバイスアドレスレジスタ148に保持されたデバイスアドレス(DAr)と一致し、したがって、アドレス一致デターミナ147は、デバイスアドレス一致結果(すなわち、アドレス一致表示147M)をもたらす。デバイスタイプ一致結果に応答して、アドレス増分オペレータ149は、DATAと1との加算を実行して、アドレス増分(「DA+1」)を達成する。デバイスアドレス一致結果に応答して、デバイスコントローラ/プロセッサ142Aは、デバイスアドレスレジスタ148に、前に保持されたアドレスを受け取られたアドレス(SIPのDATAの値または番号)に置換させ、その結果、デバイス92は、「NAND-0」としてセットされるようになる。SIPのDATAの番号は、増分されたアドレス番号によって置換される。DATAを除く命令全体(SIPの)が、バイパスされる。したがって、TYPE (NANDフラッシュ)、TDA (00h)、CMD (39h)、および増分されたDATA (01h)を含む変更されたSIP (SIP_1)が、次のデバイス93(すなわち、デバイス_1)に送られる。

【0123】

これらの動作は、ハイへのIPE遷移とハイへのIPE_1遷移との間の時間期間TP1-SI中に実行される。デバイス93は、同一の動作を実行し、「NAND-1」としてセットされる。DATAを除く命令全体(SIP_1の)が、バイパスされる。これらの動作は、ハイへのIPE_1遷移とハイへのIPE_2遷移との間の時間期間TP1-1中に実行される。受け取られたDATAは、1つだけ増分され、増分されたDATA (02h)が、デバイス93から次のデバイス94(すなわち、デバイス_2)へのSIP_2に含まれる。これらの動作は、時間期間TP1-2中に実行される。同様に、デバイス94は、同一の動作を実行し、「NAND-(n-1)」としてセットされる。DATAを除く命令全体(SIP_(n-1)の)が、バイパスされる。これらの動作は、ハイへのIPE_(n-1)遷移とハイへのIPE_n遷移との間の時間期間TP1-(n-1)中に実行される。しかし、デバイス95は、同一の動作を実行しない。SIP_nに含まれるTYPE (NANDフラッシュ)、TDA (00h)、CMD (39h)、およびDATA (nh)という入力命令に応答して、デバイス95は、デバイスタイプ一致なし(すなわち、不一致)を判定し、その命令を無視する。したがって、DATA (nh)は、DATA増分なしで、次のデバイスについて同一のままになる。デバイス95(最後のデバイス)からの命令出力は、時間期間TP1-S0中にフィードバックとしてメモリコントローラ50に転送される。メモリコントローラ50は、DATAの番号または値から、NANDタイプデバイスの総数が「n」であることを認識する。

【0124】

次に、メモリコントローラ50は、NORフラッシュのデバイスタイプに関する第2のデバイスアドレス書込み動作を発行する。やはりIPEハイ中に、SIPに含まれるTYPE (NORフラッシュ)、TDA (00h)、CMD (39h)、およびDATA (00h)が、第1デバイス92(すなわち、デバイス_0)に供給される。CMD (39h)は、デバイス92のデバイスコントローラ/プロセッサ142Aに、デバイスタイプ一致判定を実行させる。SIPのTYPE (DTs)は、NORフラッシュであり、

デバイスタイプレジスタ146に保持されたデバイスタイプ(DTr)は、NANDフラッシュである。したがって、デバイスタイプ一致デターミナ143は、デバイスタイプ一致なし結果(すなわち不一致)をもたらし、デバイス92は、受け取られた(または入力された)デバイスアドレス書込み命令を無視する。DATAは、増分なしで同一のままである。デバイス92(デバイスコントローラ/プロセッサ142A)は、TYPE (NORフラッシュ)、TDA (00h)、CMD (39h)、およびDATA (00h)を次のデバイス93に転送する。TYPE (NORフラッシュ)、TDA (00h)、CMD (39h)、および増分されていないDATA (00h)を含む変更されないSIP (SIP_1)が、次のデバイス93(すなわち、デバイス_1)に送られる。これらの動作は、ハイへのIPE遷移とハイへのIPE_1遷移との間の時間期間TP2-SI中に実行される。

【0125】

10

受け取られたSIP_1に応答して、デバイス93は、同一の動作を実行する。デバイスタイプの不一致に起因して、デバイス93(デバイスコントローラ/プロセッサ142A)は、受け取られた命令(デバイスアドレス書込み)を無視し、DATAバイトは、増分なしで同一のままになる。デバイス93は、SIP_2に含まれるTYPE (NORフラッシュ)、TDA (00h)、CMD (39h)、およびDATA (00h)を次のデバイス94(すなわち、デバイス_2)に転送する。これらの動作は、ハイへのIPE遷移とハイへのIPE_2遷移の間の時間期間TP2-1中に実行される。同様に、TYPE (NORフラッシュ)、TDA (00h)、CMD (39h)、およびDATA (00h)を含むSIP_2に응答して、デバイス94は、同一の機能を実行する。デバイスタイプの不一致に起因して、デバイス94は、デバイスアドレス書込みを無視し、DATAは、増分なしで同一のままになる。これらの動作は、時間期間TP2-(n-1)中に実行される。

20

【0126】

TYPE (NORフラッシュ)、TDA (00h)、CMD (39h)、およびDATA (00h)を含むSIP_nに응答して、デバイス95は、デバイスタイプ一致判定を実行する。SIPのTYPE (DTs)とデバイスタイプレジスタ146に保持されたデバイスタイプ(DTr)との両方が、NORフラッシュであり、したがって、デバイスタイプ一致デターミナ143は、デバイスタイプ一致結果(すなわち、タイプ一致表示143M)をもたらす。また、TDAは「00h」であり、(DAs)はデバイスアドレスレジスタ148に保持されたデバイスアドレス(DAr)と一致し、したがって、アドレス一致デターミナ147は、デバイスアドレス一致結果(すなわち、アドレス一致表示147M)をもたらす。

【0127】

30

デバイスタイプ一致結果に응答して、アドレス増分オペレータ149は、DATAと1との加算を実行して、アドレス増分(「DA+1」)を達成する。デバイスアドレス一致結果に응答して、デバイス95のデバイスコントローラ/プロセッサ142Aは、デバイスアドレスレジスタ148に、前に保持されたアドレスを受け取られたアドレス(SIPのDATAの値または番号)に置換させ、その結果、デバイス95は、「NOR-0」としてセットされるようになる。SIPのDATAの番号は、増分されたアドレス番号によって置換される。DATAを除く命令全体(SIPの)が、バイパスされる。したがって、SOPに含まれるTYPE (NORフラッシュ)、TDA (00h)、CMD (39h)、および増分されたDATA (01h)が、出力される。デバイス95は、最後のデバイスなので、出力SOPは、メモリコントローラ50に送られる。これらの動作は、ハイへのIPE_n遷移とハイへのIPEQ遷移との間の時間期間TP2-n中に実行される。デバイス95から出力される命令は、時間期間TP2-SO中にフィードバックとしてメモリコントローラ50に転送される。メモリコントローラ50は、DATAの番号または値から、NORタイプデバイスの総数が「1」であることを認識する。

40

【0128】

その後、時刻TAMEに、メモリコントローラ50が、もう1つのデバイスタイプに関するもう1つのデバイスアドレス書込み動作を発行する。初期化すべきデバイスタイプがもうない場合には、システム44は、通常動作(例えば、図2Cに示されたフェーズ2)の準備ができている。

【0129】

例の詳細を、これから、図3A~3Fを参照して上で提示した例の文脈で与える。これらの

50

詳細は、タイプ依存アドレッシングを有する実施態様に関する。これらの実施態様では、各メモリデバイスが、デバイスタイプおよびデバイスアドレスを有し、このデバイスタイプとデバイスアドレスとの両方が、アドレッシング目的に使用される。代替実施態様は、タイプ依存アドレッシングを使用するが、その詳細は、下で、異なるセクション見出しの下で与える。このセクションで与えられる詳細が、例のみのために非常に固有であることを理解されたい。

【0130】

図8に、タイプ依存アドレッシングを用いる例の入力の信号のタイミングシーケンスを示す。このタイミング図は、上で図3A~3Fを参照して説明した例のメモリシステムのすべてのメモリデバイスに適用可能である。タイミング図の最上部に、8-1によって示されるチップ選択(CS#)および8-2によって示されるシリアルクロック(SCLK)の信号がプロットされている。また、このタイミング図は、入力インターフェースに関する信号すなわち、8-3によって示される入力イネーブル(IPE)、8-4によって示されるシリアル入力(SIP)、および8-5によって示される出力イネーブル(OPE)を含む。さらに、このタイミング図は、出力インターフェースからの信号すなわち、8-6によって示されるシリアル出力(SOP)を含む。

【0131】

チップ選択CS# 8-1は、アクティブ「ロウ」であり、したがって、メモリシステム内で接続されたメモリデバイスのすべてをイネーブルするために、論理「ロウ」でなければならない。SCLK 8-2は、フリーランニングシリアルクロック信号である。IPE 8-3は、シリアライズされたバイトモードでの入力ストリームの始めを示す論理「ロウ」から論理「ハイ」への遷移点を有する。論理「ハイ」状態のIPE 8-3を受け取るメモリデバイスは、バイトモード定義でのSIPポートを介するデータストリーミングを処理する準備ができていなければならない。SIP 8-4の第1バイトは、「デバイスタイプ」の情報を担持する。第1バイトは、立ち上がりエッジに対して参照されるSCLK 8-2の8サイクルを、最初にMSB(最上位ビット)、最後にLSB(最下位ビット)の順で含む。第1バイトに続いて、SIP 8-4の第2バイトは、「デバイスアドレス」情報(例えば、ターゲットデバイスアドレス(TDA))を担持し続ける。「コマンド」情報を担持する第3バイトが、第2バイトに続き、「ロウ/コラムアドレス」を担持する第4、第5、ならびに/または第6およびさらなるバイトが続く。適用可能な場合に(例えば、書込み関連動作)、1つまたは複数のデータ入力バイトが続く。

【0132】

このタイミング図に示されているように、IPE 8-3に沿ったSIP 8-4の「シリアルバイト」の位置合わせは、SCLK 8-2の立ち上がりエッジを使用する一連の8クロックサイクルとして定義される。他の実施態様では、SCLK 8-2の立ち下がりエッジを使用することもできる。SCLK 8-2の立ち上がりエッジと立ち下がりエッジとの両方が使用される場合には、クロッキングの「ダブルエッジ」のゆえに、1つの「シリアルバイト」を形成するのに4クロックサイクルだけが必要になる。1つのバイトは、8つのビットを含み、1ビットは、論理「ハイ」または論理「ロウ」の状態のいずれかを表す。

【0133】

図示の例では、SOP 8-6は、論理「ドントケア」として示される。というのは、メモリデバイスが、次のメモリデバイスへのデータ出力をイネーブルしていないからである。しかし、メモリデバイスが、次のメモリデバイスへのデータ出力をイネーブルしていなければならない場合には、そのメモリデバイスは、出力イネーブル(OPE)を論理「ハイ」に駆動させ、SOP 8-6は、論理「ドントケア」にはならない。

【0134】

その代わりに、情報を担持するSIP 8-4のバイトは、LSBを先に、MSBを最後の位置にいかせるものであってもよい。

【0135】

メモリデバイスがデータを受け取っており、データを次のメモリデバイスに転送している時の例を、図9を参照して下で与える。

【0136】

図9に、2つの隣接するメモリデバイスを介する例のシグナリングのタイミングシーケンスを示す。このタイミング図は、図3C、3D、3E、および3Fを参照して上で説明した例のメモリシステムの隣接するメモリデバイスの各対に適用可能である。この例では、デバイス0という名前の第1デバイスおよびデバイス1という名前の第2デバイスが、説明のために選択される。すべての信号名の接尾辞「_D0」および「_D1」は、説明のために、それぞれ2つのデバイスすなわちデバイス0およびデバイス1を表す。このタイミング図の最上部に、9-1によって示されるシリアルクロック(SCLK)の信号がプロットされている。次に、このタイミング図は、デバイス0の入力インターフェースの信号すなわち、それぞれ9-2、9-3、および9-4によって示される入力イネーブル(IPE_D0)、シリアル入力(SIP_D0)、および出力イネーブル(OPE_D0)を含む。次に、このタイミング図は、デバイス1の出力インターフェースからの信号すなわち、それぞれ9-5、9-6、および9-7によって示される、シリアル出力(SOP_D0)、入力イネーブルエコー(IPEQ_D0)、および出力イネーブルエコー(OPEQ_D0)を含む。次に、このタイミング図は、デバイス1の入力インターフェースからの信号すなわち、それぞれ9-8、9-9、および9-10によって示される入力イネーブル(IPE_D1)、シリアル入力(SIP_D1)、および出力イネーブル(OPE_D1)を含む。第2メモリデバイスすなわちデバイス1の入力インターフェースへの信号入力が、第1メモリデバイスすなわちデバイス0の出力インターフェースからの信号出力と同一であることを留意されたい。次に、このタイミング図は、デバイス1の出力インターフェースからの信号すなわち、それぞれ9-11、9-12、および9-13によって示される、シリアル出力(SOP_D1)、入力イネーブルエコー(IPEQ_D1)、および出力イネーブルエコー(OPEQ_D1)を含む。

【0137】

このタイミング図は、説明のためのみに提供され、したがって、すべての波形は、実際の動作を示してはいない。時刻T2に、IPE_D0 9-2は、SCLK 9-1の立ち上がりエッジで論理「ハイ」に遷移し、これは、SIP_D0 9-3を介するシリアルデータストリームインの始めを意味する。次に、デバイス0は、SIP_D0 9-3の受取りを開始し、シリアルストリームイン情報に従って適当な動作を処理する。また、デバイス0は、IPE_D0 9-4の論理「ハイ」状態をIPEQ_D0 9-6にエコーし、このIPEQ_D0 9-6は、デバイス1のIPEポートに接続される。また、SIP_D0 9-3のストリームインデータは、SOP_D0 9-5にエコーされ、このSOP_D0 9-5は、デバイス1のSOPポートに接続される。この手順は、時刻T10まで継続され、時刻T10に、IPE_D0 9-2の論理「ロウ」状態が、SCLK 9-1の立ち上がりエッジで検出される。デバイス1レベルでは、IPE_D1 9-8が、デバイス0レベルでのIPEQ_D0 9-7信号と論理的に同一の信号波形を示す。というのは、IPEQ_D0 9-6が、ワイヤまたは他の相互接続方法を介して直接にIPE_D1 9-8に接続されるからである。また、SIP_D1 9-9は、デバイス0レベルのSOP_D0 9-5信号と論理的に同一の信号波形を示す。というのは、SOP_D0 9-5が、ワイヤまたは他の相互接続方法を介して直接にSIP_D1 9-9に接続されるからである。デバイス1では、デバイス0に類似する手順が発生し、SOP_D1 9-11へのSIP_D1 9-9のエコーおよびIPEQ_D1 9-12へのIPE_D1 9-8のエコーをもたらす。

【0138】

図示の例では、エコー手順に関する1クロックサイクルの待ち時間がある。しかし、より一般的に、任意の適当なクロックサイクル待ち時間を実施することができる。例えば、1/2クロックサイクル、2クロックサイクル、または3つ以上のクロックサイクルのクロックサイクル待ち時間を実施することができる。各メモリデバイスを介するクロックサイクル待ち時間は、メモリシステムの総クロック待ち時間を決定する。1クロックサイクル待ち時間およびシステム内の4つのデバイスを仮定すると、最後のデバイスのSOP_D3およびIPEQ_D3は、オリジナルのSIP_D0 9-3信号およびIPE_D0 9-2信号から4クロックサイクルの待ち時間を有する。デバイス0レベルのT13からT17までに、OPE_D0 9-4信号がアクティブであり、信号SOP_D0 9-5を介するデバイス0からのシリアル出力動作を引き起こす。時刻T13に、OPE_D0 9-4の論理「ハイ」状態が、SCLK 9-1の立ち上がりエッジに検出され、その後、デバイス0は、そのデバイスの前の状態に従うSOP_D0 9-5ポートを介するシリアルデータストリームの出力を開始する。この例では、デバイス0は、シリアルデータを出力す

るために選択され、デバイス1は、選択されず、したがって、デバイス1は、SIP_D1 9-9信号(SOP_D0 9-5と同一)をSOP_D1 9-11ポートにエコーするのみである。OPEポートと共のシリアル出力動作は、シリアル入力手順と同一のクロック待ち時間を有する。

【0139】

図3A～3Fを参照して説明した例で、最後のデバイス83、87、91、95、または99からの出力イネーブルエコー(OPEQ) 69は、それぞれのメモリコントローラ50に接続される。この形で、メモリコントローラ50は、クロック待ち時間の個数をカウントする必要がなく、クロック待ち時間の個数は、相互接続されたデバイスの個数によって決定される。メモリコントローラは、最後のデバイスからのOPEQ信号の立ち上がり点を検出でき、相互接続内のデバイスからのシリアルデータ出力ストリーミングの開始点を判断することができる。最後のデバイス83、87、91、95、または99からの出力イネーブルエコー(OPEQ) 69がメモリコントローラ10に接続されない代替実施態様では、メモリコントローラ50は、シリアルデータがシリアル入力(SIP) 57を介して受け取られる時に、クロック待ち時間の以前の知識に基づいて予測することができる。

【0140】

図8および9を参照して上で説明したタイミング図では、SIPおよび(適用可能な場合に)SOPについてプロットされた信号が、メモリ動作を含み、このメモリ動作は、所定のフォーマットに従う。メモリ動作に関する例の所定のフォーマットの表を、下で図10を参照して説明する。

【0141】

図10に、タイプ依存アドレッシングを用いるメモリ動作の例の所定のフォーマットの表を示す。この表が、例のためにのみ非常に固有であることを理解されたい。この表では、

TYPE: ターゲットデバイスタイプ
TDA: ターゲットデバイスアドレス
CMD: コマンドコード
CA: カラムアドレス
RA: ロウアドレス

である。

注 *1:TDA (ターゲットデバイスアドレス)は、パワーアップまたはハードリセットの後に第1のデバイスアドレス書込みコマンドが発行される時に「00h」である。

【0142】

図10を参照すると、この表は、異なるメモリ動作の変化するフォーマットを示す。この表では、8つメモリ動作すなわち、読取り、書込み、消去、状況読取り、ID読取り、構成レジスタ書込み、デバイスアドレス書込み、およびリセットがリストされている。他のメモリ動作があってもよいが、これらは、表を単純にするために示されていない。第1バイトは、デバイスタイプ(TYPE)を定義する。この情報を、SIPポートを介するデータのシリアル入力ストリームを処理すべきか否かを判断するために、オンチップの事前にプログラムされるデバイスタイプレジスタ値と比較することができる。デバイスタイプと共に、第2列は、ターゲットデバイスアドレス(TDA)を指定し、これは、同一のデバイスタイプのメモリデバイス間で区別するのに使用される。第3バイトは、コマンド定義(CMD)を定義する。適当な場合に(例えば、読取り動作)、第4、第5、および/またはさらなるバイトは、ロウアドレス(RA)および/またはカラムアドレス(CA)情報を定義する。適当な場合に(例えば、書込み動作)、追加のバイトが、動作によって送られるデータ(DATA)を定義する。

【0143】

デバイスタイプ、デバイスアドレス、およびコマンドは、これらがデバイスタイプに固有になる形で符号化される。例の符号化方式を、図10から13までを参照して下で説明する。これらの符号化方式が、例のみのために非常に固有であることを理解されたい。符号化方式は、製造業者によって、彼ら自身の目的のために異なる形で変更され得る。

【0144】

図11に、タイプ依存アドレッシングの例の符号化方式の表を示す。この表では、

DA[7:0]:デバイスアドレス(この例では、デバイスの最大個数= $2^8=256$ 個である)

CA[11:0]:カラムアドレス(この例では、カラムの最大個数= $2^{12}=4096$ 個である)

RA[17:0]:ロウアドレス(この例では、ロウの最大個数= $2^{18}=262144$ 個である)

【 0 1 4 5 】

図11を参照すると、この表は、デバイスアドレス(TDA)、ロウアドレス(RA)、およびカラムアドレス(CA)の符号化方式を定義する。この例では、デバイスアドレスは、合計8ビットを有し、したがって、システム内で構成できるデバイスの最大個数は、 $2^8=256$ 個である。しかし、このデバイスアドレス定義を、適宜別のシリアルバイト(1つまたは複数)を使用して拡張することができる。また、ロウアドレスバイトおよびカラムアドレスバイトは、デバイスアドレスフォーマットと類似する形で示される。図5Bの表について上で注記したように、表の定義を、その代わりに、LSBを先に有するために逆の順序とすることができる。

10

【 0 1 4 6 】

図12に、タイプ依存アドレッシングを用いるNANDフラッシュコマンドの例の符号化方式の表を示す。

【 0 1 4 7 】

この表では、

*1:ターゲットDAは、「デバイスアドレス書込み」コマンドがパワーアップまたはハードリセットの後に発行される時に00hでなければならない。

*2:ロウアドレスバイトおよびカラムアドレスバイトは、同一位置のページ読取りコマンドが前に発行された場合に、供給されない場合がある。

20

【 0 1 4 8 】

図12を参照すると、このテーブルは、13個のコマンドすなわち、ページ読取り、ランダムデータ読取り、コピーのためのページ読取り、コピーのためのターゲットアドレス入力、シリアルデータ入力、ランダムデータ入力、ページプログラム、ブロック消去、状況読取り、ID読取り、構成レジスタ書込み、デバイスアドレス書込み、およびリセットのそれぞれについて符号化方式を定義する。各コマンドは、デバイスタイプ(デバイスTYPE)を含み、このデバイスタイプは、図5Bの表によれば、NANDフラッシュメモリについて「00h」である。次に、各コマンドは、デバイスアドレス(ターゲットDA)を含み、このデバイスアドレスは、「有効」として示される。デバイスアドレスは、シリアル相互接続内の特定のデバイスを選択するために任意のデバイスアドレスを識別することができる。デバイスアドレス列は、適当な場合に、より多くのバイトに拡張することができる。次に、各コマンドは、コマンド定義を含む。図示のコマンド定義は、従来のNANDフラッシュメモリコマンド定義に類似する。この表の第4列および第5列は、それぞれ、NANDフラッシュデバイスのメモセルアレイブロック内の特定のロウ位置およびカラム位置の選択のための、ロウアドレスおよびカラムアドレスを表す。この表に示されているように、一部のコマンドは、ロウアドレスおよび/またはカラムアドレスを含まない。

30

【 0 1 4 9 】

ロウアドレス範囲およびカラムアドレス範囲ごとのバイト数は、特定の密度のメモリアレイサイズに従って変更することができる。ロウアドレスおよびカラムアドレスを、いずれかの形で切り替えることができる。したがって、カラムアドレスバイトを、その代わりに最初にすることができ、ロウアドレスバイトがカラムアドレスに続くことができる。これは、特定のメモリチップ設計ブリファレンスに依存する。最後の列は、例えば「シリアルデータ入力(80h)」、「ランダムデータ入力(85h)」、および「構成レジスタ書込み(A0h)」などの「書込み」動作コマンドの入力データカラム定義を示す。この入力データバイトは、デバイス仕様に従って、1バイトもの小ささまたはNバイトもの大きさとすることができる。「状況読取り(70h)」コマンドは、同一のシリアルリンクポートSOPを使用して各デバイスの状況をチェックするために必要であり、そうでなければ、各デバイスは、状況表示のために別々の余分なハードピンを必要とする。これを、異なる16進数定義に変更することもできる。「デバイスアドレス書込み(39h)」コマンドは、相互接続されたデバイ

40

50

すが、ハードピン構成ではなくソフト生成されるデバイス番号を使用する場合に使用される。「リセット(FFh)」コマンドは、各選択されたデバイスに対してソフトリセット機能を実行することができる。このソフトリセットは、相互接続内のすべてのデバイスに接続される「RST#」ポートを使用する「ハードリセット」から区別される。

【 0 1 5 0 】

図13に、タイプ依存アドレッシングを用いるNORフラッシュコマンドの例の符号化方式の表を示す。この表では、

注*1:ターゲットDAは、パワーアップまたはハードリセットの後に「デバイスアドレス書込み」コマンドが発行される時に「00h」でなければならない。

注*2:ロウアドレスバイトおよびカラムアドレスバイトは、同一位置の読取りコマンドが前に発行された場合に、供給されない場合がある。

【 0 1 5 1 】

図13の表は、図12の表と類似するフォーマットに従う。しかし、図13の表は、12個のコマンドすなわち、読取り、バッファへの書込み、フラッシュするためにバッファをプログラムする(確認)、チップ消去、セクタ消去、プログラム/消去一時停止、プログラム/消去再開、状況読取り、ID読取り、構成レジスタ書込み、デバイスアドレス書込み、およびリセットの異なるセットを有すると見ることができる。各コマンドは、デバイスタイプ(デバイスTYPE)を含み、このデバイスTYPEは、図5Bの表によれば、NORフラッシュメモリについて「01h」である。「DNエントリ書込み(39h)」コマンドは、相互接続されたデバイスが、ハードピン構成ではなくソフト生成されるデバイス番号を使用する場合に使用される。図12の表と同様に、「リセット(FFh)」コマンドは、各選択されたデバイスに対してソフトリセット機能を実行することができる。このソフトリセットは、相互接続内のすべてのデバイスに接続される「RST#」ポートを使用する「ハードリセット」から区別される。

【 0 1 5 2 】

図14に、タイプ依存アドレッシングを用いてメモリ動作を処理する方法を示す。このプロセスは、全般的な概念を示す。特定のコマンドまたは動作流れ図が、この例と異なる場合がある。例えば、データの読取りまたは書込みを伴わない動作は、データの転送を伴わない。また、コマンドが、ロウアドレスまたはカラムアドレスを伴わない場合に、メモリデバイスは、ロウ/カラムアドレスバイトを転送しない。シリアル信号ストリームバイトが、相互接続内の各デバイスのIPE、SIP、OPE、またはSOPを介してバイパスされる時には、バイパス回路が1クロック待ち時間を用いて設計される場合に、1クロックサイクルの待ち時間遅延がある。

【 0 1 5 3 】

図14を参照すると、メモリデバイスがメモリコマンドを受け取る時に、そのメモリデバイスは、メモリコマンドによって示されるデバイスタイプを、そのタイプレジスタによって示されるそれ自体のデバイスタイプと比較する(ステップ14-1)。メモリコマンドは、そのメモリコマンドによって示されるデバイスタイプに固有である。メモリデバイスは、メモリコマンドのデバイスタイプがそのレジスタのデバイスタイプと一致するかどうかを判定する(ステップ14-2)。この2つのデバイスタイプの間にタイプ一致がある場合に(ステップ14-2のYES)、メモリデバイスは、さらに、メモリコマンドによって示されるデバイスアドレスを、そのデバイスアドレスレジスタによって示されるそれ自体のデバイスアドレスと比較する(ステップ14-3)。メモリデバイスは、メモリコマンドのデバイスアドレスがそのレジスタのデバイスアドレスと一致するかどうかを判定する(ステップ14-4)。アドレス一致がある場合に(ステップ14-4のYES)、メモリデバイスは、そのコマンドを実行する(ステップ14-5)。コマンドに応じて、これは、メモリデバイスが、メモリコマンドによって示されるロウアドレスおよびカラムアドレスを処理することを伴う場合があり、メモリコマンドの一部として受け取られたデータを処理することを伴う場合もある。しかし、デバイスタイプの一致がない場合(ステップ14-2のNO)、またはデバイスアドレスの一致がない場合(ステップ14-4のNO)に、メモリデバイスは、メモリコマンドを次のメモリデバイスに転送すること以外に、メモリコマンドの内部処理を実行しない(ステップ14-6)。

【 0 1 5 4 】

図14を参照して上で説明したメモリ動作を処理するプロセスのさらなる説明を提供するために、図15Aおよび15Bを参照して、タイミング図を下で説明する。

【 0 1 5 5 】

図15Aおよび15Bに、タイプ依存アドレッシングを用いるメモリ動作の処理に関する信号のタイミングシーケンスを示す。このタイミング図は、図3Bのメモリシステムから生じる可能性がある例の信号を示し、図3Bのメモリシステムは、相互接続された1つのNORタイプフラッシュデバイス84および3つのNANDタイプフラッシュデバイス85、86、...、および87を有する。

【 0 1 5 6 】

メモリコントローラによって発行されるNANDフラッシュデバイスのページ読取りに関するメモリコマンドは、次のとおりである。

【 0 1 5 7 】

【表 5】

メモリコマンド(4)			
TYPE (00h)	TDA (01h)	CMD (00h)	Raw/Column Addresses

【 0 1 5 8 】

このメモリコマンドにおいて、

TYPE (00h)は、「NANDフラッシュ」デバイスを識別する(図5Bを参照されたい)。

TDA (01h)は、デバイスアドレス「1」を保持するデバイスを識別する。

CMD (00h)は、実行される動作が「ページ読取り」であることを識別する。

Raw/Column Addressesは、DATAではなく、メモリのロウアドレスおよびカラムアドレスを識別する。

【 0 1 5 9 】

同様に、NORフラッシュデバイスのページ読取りに関するメモリコマンドは、次のとおりである。

【 0 1 6 0 】

【表 6】

メモリコマンド(4)			
TYPE (01h)	TDA (01h)	CMD (00h)	Raw/Column Addresses

【 0 1 6 1 】

図3B、15A、および15Bを参照すると、NORタイプフラッシュデバイス84は、相互接続の最初のデバイスである(すなわち、メモリコントローラ50に最も近い)。このデバイスは、「NOR-0」として一意のタイプおよびデバイス番号(または、タイプおよびデバイス識別、あるいはタイプおよびデバイスアドレス)を有する。NANDタイプフラッシュデバイス85、86、...、および87は、NOR-0デバイス84の次に直列に接続され、「NAND-0」、「NAND-1」、および「NAND-(n-1)」として一意のデバイス番号を有する。このタイミング図の最上部に、15-1によって示されるシリアルクロック(SCLK)の信号がある。次に、このタイミング図は、それぞれ15-2、15-3、15-4、および15-5によって示される、各メモリデバイス84、85、86、...、および87のシリアル入力(SIP)の信号を含む。次に、このタイミング図は、それぞれ15-6、15-7、15-8、および15-9によって示される、各メモリデバイス84、85、86、...、および87の出力イネーブル(OPE)の信号を含む。次に、このタイミング図は、15-10によって示される、最後のメモリデバイス87の出力イネーブルエコー(OPEQ)の信号を含む。最後に、このタイミング図は、それぞれ15-11、15-12、15-13、および15-14によって示される、各メモリデバイス84、85、86、...、および87のシリアル出力(SOP)の信号を含む。単純な説明のために、IPE、CS#、RST#などの他の信号は、このタイミング図には示されて

10

20

30

40

50

いない。

【0162】

このタイミング図では、15-15によって示されるように、「NAND-1のページ読取りコマンドセット」が、まず、デバイスタイプ(TYPE=NAND)、ターゲットデバイスアドレス(DA=1)、コマンド(00h)、およびロウ/カラムアドレスを伴って発行される。入力信号のこのシリアルストリームは、シーケンス内のデバイスを介して渡され、選択されたデバイス(このケースではNAND-1)だけが、デバイスの内部で所与の「ページ読取り」コマンドを処理する。通常、NANDタイプフラッシュメモリは、NANDフラッシュセルからデータレジスタブロックにデータを転送する内部「ページ読取り動作」にはるかにより長い時間(通常は20 μ 秒)を要する。したがって、メモリコントローラは、その20 μ 秒という時間にわたって待たなければならない。しかし、メモリコントローラは、NAND-1の長いページ読取り時間にわたって待っている間に、NORタイプフラッシュデバイスすなわちNOR-0にアクセスすることができる。したがって、15-16に示されているように、「NOR-0のページ読取りコマンドセット」が、「NAND-1のページ読取りコマンドセット」の直後に発行される。NORタイプフラッシュメモリは、例えば100ナノ秒など、非常に速い読取りアクセス時間を有し、したがって、メモリコントローラは、NOR-0からの「デマンドページング」などの多数の高速動作を実行することができる。「デマンドページング」は、仮想メモリを実施する単純な方法である。

【0163】

デマンドページングを使用するシステムでは、オペレーティングシステムが、ページにアクセスする試みが行われる場合(すなわち、ページフォールトが発生する場合)に限って、そのページを物理メモリにコピーする。したがって、プロセスは、物理メモリ内にそのページのいずれをも有しないで実行を開始することになり、多数のページフォールトが、プロセスのページのワーキングセットのほとんどが物理メモリに配置されるまでに発生する。15-17によって示されるように、NOR-0からの最後の読取りデータ出力は、NAND-2のSOPポートに現れ、このSOPポートは、接続されたメモリデバイスの総数が4なので、4クロックサイクル待ち時間の後にコントローラのSOPポートに直接に接続される。長い時間にわたって待った後に、メモリコントローラは、NAND-1にアクセスすることができる。この時に、15-18によって示されるように、メモリコントローラは、ロウ/カラムアドレスバイトなしで「NAND-1のページ読取りコマンドセット」を発行し、OPE 15-6信号を論理「ロウ」から論理「ハイ」状態にし、これによって、NAND-1デバイス内の出力回路網がイネーブルされ、その後、NAND-1からの読取りデータ出力が、15-19によって示されるように直列に接続されたSIP/SOPポートを介してストリームアウトされ始める。メモリコントローラのSOPポートでの最終的なデータ出力には4クロックサイクルの待ち時間がある。

【0164】

図16Aに、図3A～3Eに示されたメモリデバイスとして使用されるもう1つの例のメモリデバイスブロックを示す。図16Aに示されたメモリデバイス140Aは、図4Aに示されたメモリデバイス140Aに類似する。図16Aを参照すると、アドレス増分オペレータ149は、デバイスコントローラ/プロセッサ142Aからの初期フェーズの要求に応答して動作を実行する。そのような特定の例で、割り当てられるデバイスアドレスは、そのデバイスによって増分されたデバイスアドレスである。各デバイスは、図6Aに示されたデバイスアドレス割当て方法を実行する。しかし、図6Bのステップ6-10および6-11は、図16Bに示されているよう逆にされる。

【0165】

図3B、4A、16A、および16Bを参照すると、デバイスアドレス書込み動作を受け取ったデバイスは、受け取られたデバイスタイプ(「DTs」)がデバイスタイプレジスタ146に登録されたそのデバイスのデバイスタイプと一致するかどうかを判定する(ステップ16-8)。デバイスタイプ一致(ステップ16-8のYES)またはデバイスタイプ一致結果がある場合に、アドレス一致デターミナ147は、ターゲットデバイスアドレス(TDA)(すなわち、「DAs」)がデバイスアドレスレジスタ148に登録されたデバイスアドレス(DAr)と一致するかどうかを判

定する(ステップ16-9)。デバイスアドレス一致(ステップ16-9のYES)またはデバイスアドレス一致結果がある場合に、デバイスアドレス一致デターミナ147は、アドレス一致表示147Mを出力する。次に、受け取られたデバイスアドレス(「DA」)が、アドレス増分オペレータ149によって増分される(ステップ16-10)。増分されたアドレス(「DA+1」)は、デバイスアドレスレジスタ148に登録され(ステップ16-11)、増分されたデバイスアドレスは、次のデバイスに送られる。デバイスタイプ一致がない(ステップ16-8のNO)または非デバイスタイプ一致結果がある場合には、デバイスアドレス割当てもデバイスアドレス増分も実行されない。また、デバイスアドレス一致がない(ステップ16-9のNO)または非デバイスアドレス一致結果がある場合には、デバイスアドレス割当てもアドレス増分も実行されない。

【 0 1 6 6 】

10

図16Cに、図16Aのデバイスによって実行されるもう1つのデバイスアドレス割当て動作を示す。図16Cのステップ16-8および16-9は、図16Bのこれらのステップと同一である。デバイスタイプ一致(ステップ16-8のYES)およびデバイスアドレス一致(ステップ16-9のYES)がある場合に、受け取られたデバイスアドレス(「DA」)が、アドレス増分オペレータ149によって増分される(ステップ16-12)。増分されたアドレス(「DA+1」)が、デバイスアドレスレジスタ148に登録され(ステップ16-13)、増分されたデバイスアドレスは、次のデバイスに送られる(ステップ16-14)。

【 0 1 6 7 】

図17に、図3Fに示されたメモリデバイスとして使用されるもう1つの例のメモリデバイスブロックを示す。図17に示されたメモリデバイス140Bは、図4Aに示されたメモリデバイス140Aに類似する。図17を参照すると、アドレス増分オペレータ149は、デバイスコントローラ/プロセッサ142Bからの初期フェーズの要求に応答して動作を実行する。デバイス140Bは、図16Bの動作に似た動作を実行する。アドレス増分オペレータ149によって与えられる増分されたデバイスアドレスは、デバイスアドレスレジスタ148に登録され、次のデバイスに送られる。

20

【 0 1 6 8 】

図18に、本発明のもう1つの実施形態による2チャネルのメモリシステムを示す。図18を参照すると、メモリコントローラ150の第1チャネルは、シリアルリンクを介して接続されたメモリデバイスの第1シリアル相互接続構成151に接続される。同様に、メモリコントローラ150の第2チャネルは、シリアルリンクを介して接続されたメモリデバイスの第2シリアル相互接続構成152に接続される。シリアル相互接続構成151のそれぞれの最後のデバイスからのシリアル出力(SOP)、入力イネーブルエコー(IPEQ)、および出力イネーブルエコー(OPEQ)は、メモリコントローラ150にフィードバックされる。

30

【 0 1 6 9 】

メモリデバイスの第1シリアル相互接続構成151の詳細を、図19Aに示す。メモリデバイスの第2シリアル相互接続構成152の詳細を、図19Bに示す。

【 0 1 7 0 】

図19Aを参照すると、メモリデバイスの第1シリアル相互接続構成151は、シリアル相互接続された(n+1)個のNORフラッシュメモリデバイス160、161、162、...、および163を含む。デバイス160、161、162、...、および163は、それぞれNORフラッシュメモリコア160A、161A、162A、...、および163Aを含む。デバイス160、161、162、...、および163のそれぞれは、そのメモリタイプ(NORフラッシュ)を保持するレジスタ60を有する。初期化の動作中に、デバイス160、161、162、...、および163は、それぞれデバイスアドレス「NOR-0」、「NOR-1」、「NOR-2」、...、および「NOR-n」を割り当てられる。割り当てられたデバイスアドレスは、デバイスのレジスタ(図示せず)に保持される。

40

【 0 1 7 1 】

図19Bを参照すると、メモリデバイスの第2シリアル相互接続構成152は、シリアル相互接続された(n+1)個のNANDフラッシュメモリデバイス170、171、172、...、および173を含む。デバイス170、171、172、...、および173は、それぞれNANDフラッシュメモリコア170A、171A、172A、...、および173Aを含む。デバイス170、171、172、...、および173のそれぞ

50

れは、そのメモリタイプ(NANDフラッシュ)を保持するレジスタ60を有する。初期化の動作中に、デバイス170、171、172、...、および173は、それぞれデバイスアドレス「NAND-0」、「NAND-1」、「NAND-2」、...、および「NAND-n」を割り当てられる。割り当てられたデバイスアドレスは、デバイスのレジスタ(図示せず)に保持される。

【0172】

代替案では、メモリデバイスの第1シリアル相互接続構成151が、混合されたタイプのデバイスを含んでもよい。また、メモリデバイスの第2シリアル相互接続構成152が、混合されたタイプのデバイスを含んでもよい。

【0173】

図20Aおよび20Bに、図18に示されたメモリシステム内で使用される他の特定の例のメモリデバイスの概略を示す。

【0174】

図20Aを参照すると、メモリデバイスの第1シリアル相互接続構成151は、図3Bの例と同一であるメモリコア210A、211A、212A、および213Aを有する(n+1)個のメモリデバイス210、211、212、...、および213を含む。図3Bに示された特定の例では、第1メモリデバイス210は、NORフラッシュメモリコア210Aを有し、第2、第3、...、および第(n+1)のメモリデバイス211、212、...、および213は、それぞれNANDフラッシュメモリコア211A、212A、...、および213Aを有する。図3Cの例は、タイプごとのアドレッシング方式が使用される、すなわち、前に導入した第1アドレッシング方式が使用されるという点で、図3Bの例とは異なる。タイプおよびアドレスは、1つのNORデバイスおよび「n」個のNANDデバイスがあると仮定して、「NOR-0」、「NAND-0」、「NAND-1」、...、および「NAND-(n-1)」として示される。

【0175】

図20Bを参照すると、メモリデバイスの第2シリアル相互接続構成152は、それぞれ混合されたメモリコア220A、221A、222A、...、および223Aを有する複数(n+1)個のメモリデバイス220、221、222、...、および223を含む。第1メモリデバイス220は、SRAMメモリコア220Aを有し、第2メモリデバイス221は、NORフラッシュメモリコア221Aを有する。第3、...、および第(n+1)メモリデバイス222、...、および223は、それぞれNANDフラッシュメモリコア222A、...、および223Aを有する。メモリデバイスは、タイプごとのアドレッシング方式を使用してアドレッシングされる。タイプおよびアドレスは、1つのSRAMデバイス、1つのNORデバイス、および「n-1」個のNANDデバイスがあると仮定して、「SRAM-0」、「NOR-0」、「NAND-0」、...、および「NAND-(n-2)」として示される。

【0176】

上で説明した実施形態では、あるメモリコマンド(例えば、SIPの「デバイスアドレス書込み」コマンド)が、別のメモリコマンド(例えば、SIP₁の「デバイスアドレス書込み」コマンド)とオーバーラップされない。もう1つの実施形態では、デバイスへのシリアル入力内のメモリコマンドを、図21に示されているようにオーバーラップさせることができる。しかし、これは、あるデバイスによるデバイスアドレス(DATA)増分が、他のデバイスがアドレス(DATA)増分を実行する前に完了しなければならないことを必要とする。

【0177】

データ、情報、または信号の送出力が、単一ビットまたは複数のビットによって実行されることは、当業者に明白であろう。例えば、シリアル入力SIPおよびシリアル出力SOPを介するデータ伝送は、図22に示されているように単一ビットまたは複数のビット(Mビット)によって実行され、Mは、1より大きい整数である。インターフェースは、単一の入出力ピンまたは複数の入出力ピンを含んでもよい。

【0178】

図23に、本発明のもう1つの実施形態によるメモリシステムを示す。図23に示されたメモリシステムは、複数のメモリデバイスのシリアル相互接続構成351およびデバイスの動作を制御するメモリコントローラ350を含む。この構成のメモリデバイスの詳細を、図24に示す。図示の例では、構成は、(n+1)個のメモリデバイスすなわち、デバイス0、デバイ

ス1、デバイス2、...、およびデバイスnを含む。メモリデバイスのそれぞれは、複数のポートを有する。図24の特定の例では、各デバイスは、2ポートデバイスである。メモリコントローラ350は、リセット信号「RST#」、チップ選択信号「CS#」、およびシリアルクロック信号「SCLK」をメモリデバイスのそれぞれのそれぞれのポートに供給する。

【0179】

図23および24を参照すると、第1メモリデバイス(デバイス0)は、複数のデータ入力ポート(SIP1、SIP2)、複数のデータ出力ポート(SOP1、SOP2)、複数の制御入力ポート(IPE1、IPE2)、および複数の制御出力ポート(OPE1、OPE2)を有する。データ信号および制御信号は、メモリコントローラ350から第1メモリデバイスに送られる。第2メモリデバイス(デバイス1)は、デバイス1が接続されるデバイス0と同一のタイプのポートを有する。例えば、デバイス1は、デバイス0からデータ信号および制御信号を受け取る。構成内の最後のメモリデバイス(デバイスn)は、データ信号および制御信号を、所定の待ち時間の値にメモリコントローラ350に戻って供給する。各メモリデバイスは、IPE1、IPE2、OPE1、およびOPE2(すなわち、制御出力ポート)のエコー(IPEQ1、IPEQ2、OPEQ1、OPEQ2)を後続デバイスに出力する。

【0180】

図25に、本発明のもう1つの実施形態によるメモリシステムを示す。図25に示されたメモリシステムは、メモリコントローラ450および複数のメモリデバイスのシリアル相互接続構成451を含む。デバイスの構成を、図26に示す。メモリデバイスのそれぞれは、複数のポートを有する。図26の特定の例では、各デバイスが、2ポートデバイスである。メモリコントローラ450は、デバイスのポートの極性に対応する信号の複数のグループを供給する。図示の例では、リセット信号「RST#1」、チップ選択信号「CS#1」、およびシリアルクロック信号「SCLK1」が、メモリデバイスのそれぞれのそれぞれのポート1に供給される。同様に、ポート2について、リセット信号「RST#2」、チップ選択信号「CS#2」、およびシリアルクロック信号「SCLK2」が、メモリデバイスのそれぞれのそれぞれのポートに供給される。

【0181】

図23～26に示されたメモリシステムおよびデバイスで、図4Aおよび16Aに示されたデバイスを、メモリデバイスのシリアル相互接続構成において使用することができる。また、図4Bおよび17に示されたデバイスを、メモリデバイスのシリアル相互接続構成で使用することができる。その場合に、クロック信号SCLKは、図3Fに示されているように送られる必要があり、各デバイスは、次のデバイスのために出力エコークロック信号「SCLK_0」を供給するクロック同期回路を有する。

【0182】

上で説明した実施形態で、デバイス要素および回路が、説明を単純にするために、図面に示されているように互いに接続される。本発明の特定の応用では、要素、回路などを、互いに直接に接続することができる。また、要素、回路などを、デバイスまたは装置の動作に必要な他の要素、回路などを介して互いに間接に接続することができる。したがって、デバイスおよび装置の実際の構成では、要素および回路が、お互いに直接にまたは間接に結合され、または接続される。

【0183】

半導体デバイスを、デバイスとして実施できることは、当業者に明白であろう。

【0184】

本発明の上で説明した実施形態は、例のみとして意図されている。添付の特許請求の範囲のみによって定義される本発明の範囲から逸脱しない代替形態、修正形態、および変形形態を、当業者によって特定の実施形態に対してもたえらうことができる。

【符号の説明】

【0185】

7-1 電源の信号(VDD)

7-2 デバイス_0の入力イネーブル(IPE)

10

20

30

40

50

7-3	デバイス_0のシリアル入力(SIP)	
7-4	デバイス_1の入カインーブル(IPE_1)	
7-5	デバイス_1のシリアル入力(SIP_1)	
7-6	デバイス_2の入カインーブル(IPE_2)	
7-7	デバイス_2のシリアル入力(SIP_2)	
7-8	デバイス_(n-1)の入カインーブル(IPE_n-1)	
7-9	デバイス_(n-1)のシリアル入力(SIP_n-1)	
7-10	デバイス_nの入カインーブル(IPE_n)	
7-11	デバイス_nのシリアル入力(SIP_n)	
7-12	デバイス_nの入カインーブルエコー(IPEQ)	10
7-13	デバイス_nのシリアル出力(SOP)	
7-14	第1のデバイスアドレス書込み動作	
7-15	第2のデバイスアドレス書込み動作	
8-1	チップ選択(CS#)	
8-2	シリアルクロック(SCLK)	
8-3	入カインーブル(IPE)	
8-4	シリアル入力(SIP)	
8-5	出カインーブル(OPE)	
8-6	シリアル出力(SOP)	
9-1	シリアルクロック(SCLK)	20
9-2	入カインーブル(IPE_D0)	
9-3	シリアル入力(SIP_D0)	
9-4	出カインーブル(OPE_D0)	
9-5	シリアル出力(SOP_D0)	
9-6	入カインーブルエコー(IPEQ_D0)	
9-7	出カインーブルエコー(OPEQ_D0)	
9-8	入カインーブル(IPE_D1)	
9-9	シリアル入力(SIP_D1)	
9-10	出カインーブル(OPE_D1)	
9-11	シリアル出力(SOP_D1)	30
9-12	入カインーブルエコー(IPEQ_D1)	
9-13	出カインーブルエコー(OPEQ_D1)	
10	メモリコントローラ	
11	出力インターフェース	
12	入力インターフェース	
13	コントローラ動作回路網	
14	インターフェース	
15-1	シリアルクロック(SCLK)	
15-2	メモリデバイス84のシリアル入力(SIP)	
15-3	メモリデバイス85のシリアル入力(SIP)	40
15-4	メモリデバイス86のシリアル入力(SIP)	
15-5	メモリデバイス87のシリアル入力(SIP)	
15-6	メモリデバイス84の出カインーブル(OPE)	
15-7	メモリデバイス85の出カインーブル(OPE)	
15-8	メモリデバイス86の出カインーブル(OPE)	
15-9	メモリデバイス87の出カインーブル(OPE)	
15-10	メモリデバイス87の出カインーブルエコー(OPEQ)	
15-11	メモリデバイス84のシリアル出力(SOP)	
15-12	メモリデバイス85のシリアル出力(SOP)	
15-13	メモリデバイス86のシリアル出力(SOP)	50

15-14	メモリデバイス87のシリアル出力(SOP)	
15-15	NAND-1のページ読取りコマンドセット	
15-16	NOR-0のページ読取りコマンドセット	
15-17	NOR-0からの最後の読取りデータ出力	
15-18	NAND-1のページ読取りコマンドセット	
21	入力インターフェース	
22	出力インターフェース	
23	メモリデバイス動作回路網	
25	レジスタ	
30-0、30-1、...、および30-N	メモリデバイス	10
32-0、32-1、...、32-N	メモリコア	
35	初期化フェーズ	
36	通常動作フェーズ	
40	メモリシステム	
41	メモリシステム	
42	メモリシステム	
43	メモリシステム	
44	メモリシステム	
45	メモリシステム	
50	メモリコントローラ	20
50A	メモリコントローラ	
51	リセットポート	
52	チップ選択ポート	
53	シリアルクロックポート	
54	シリアル出力	
55	入力イネーブル	
56	出力イネーブル	
57	シリアル入力	
58	入力イネーブルエコー	
59	出力イネーブルエコー	30
60	レジスタ	
61	リセットポート	
62	チップ選択ポート	
63	シリアルクロックポート	
64	シリアル入力	
65	入力イネーブル	
66	出力イネーブル	
67	シリアル出力	
68	入力イネーブル	
69	出力イネーブルエコー	40
80	第1デバイス(「デバイス0」)	
80A、81A、82A、...、および83A	メモリ	
81	第2デバイス(「デバイス1」)	
82	第3デバイス(「デバイス2」)	
83	第(n+1)デバイス(「デバイスn」)	
84、85、86、...、87	メモリデバイス	
84A、85A、86A、87A	メモリコア	
88、89、90、...、91	メモリデバイス	
88A、89A、90A、...、および91A	メモリコア	
92、93、...、94、および95	メモリデバイス	50

92A、93A、94A、...および95A	メモリア	
96、97、98、...、および99	メモリデバイス	
96A、97A、98A、...および99A	メモリア	
100	コントローラ	
101	アプリケーションプロセッサ	
102	NORフラッシュメモリ	
103	揮発性メモリ	
104	アプリケーションプロセッサ	
105	NORフラッシュメモリ	
106	NANDフラッシュメモリ	10
107	揮発性メモリ	
108	アプリケーションプロセッサ	
109	OPT (one-time-programmable)ストレージ	
110	NANDフラッシュメモリ	
111	揮発性メモリ	
112	アプリケーションプロセッサ	
113	ハイブリッドNANDフラッシュメモリ	
114	揮発性メモリ	
120	レジスタブロック	
121	eFuseアレイ	20
122	eFuseレベル検出論理ユニット	
130	コントローラ動作回路網	
140A	メモリデバイス	
140B	メモリデバイス	
142A	デバイスコントローラ/プロセッサ	
142B	デバイスコントローラ/プロセッサ	
143	デバイスタイプ一致デターミナ	
143M	タイプ一致表示	
144	メモリ	
146	デバイスタイプレジスタ	30
147	アドレス一致デターミナ	
147M	アドレス一致表示	
148	デバイスアドレスレジスタ	
149	アドレス増分オペレータ	
150	メモリコントローラ	
151	メモリデバイスの第1シリアル相互接続構成	
152	メモリデバイスの第2シリアル相互接続構成	
160、161、162、...、および163	NORフラッシュメモリデバイス	
160A、161A、162A、...、および163A	NORフラッシュメモリア	
170、171、172、...、および173	NORフラッシュメモリデバイス	40
170A、171A、172A、...、および173A	NANDフラッシュメモリア	
180、181、182、...および183	メモリデバイス	
191	クロックシンクロナイザ	
210、211、212、...、および213	メモリデバイス	
210A、211A、212A、および213A	メモリア	
220、221、222、...、および223	メモリデバイス	
220A、221A、222A、および223A	メモリア	
230	デバイス動作回路網	
250	レジスタ	
300-0、300-1、...、および300-N	デバイス	50

320-0、320-1、...、および320-N メモリ
350 メモリコントローラ
351 複数のメモリデバイスのシリアル相互接続構成
450 メモリコントローラ
451 複数のメモリデバイスのシリアル相互接続構成
L0、L1、L2、...、LNおよびL(N+1) シリアルリンク

【図 1 A】

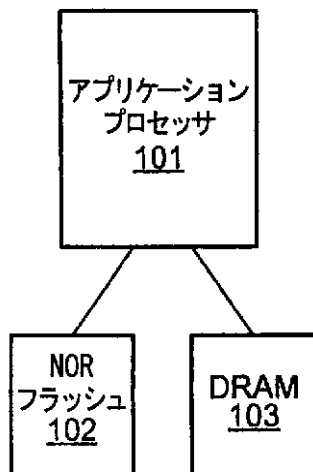


FIG. 1A

(従来技術)

【図 1 B】

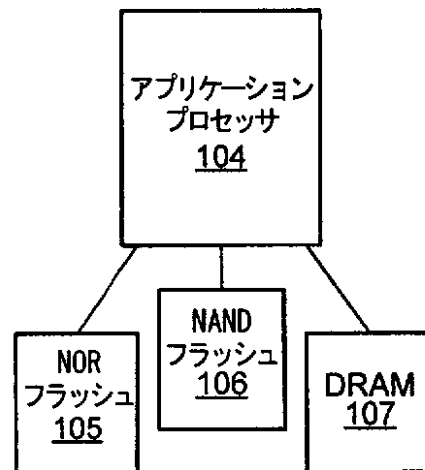


FIG. 1B

(従来技術)

【図 1 C】

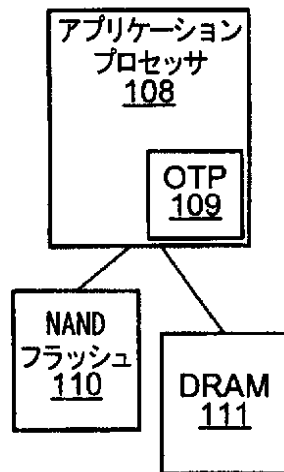


FIG. 1C

(従来技術)

【図 1 D】

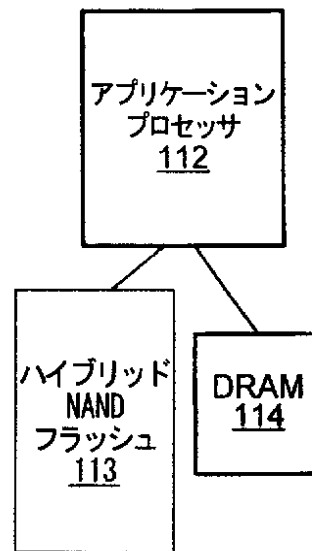


FIG. 1D

(従来技術)

【図 2 A】

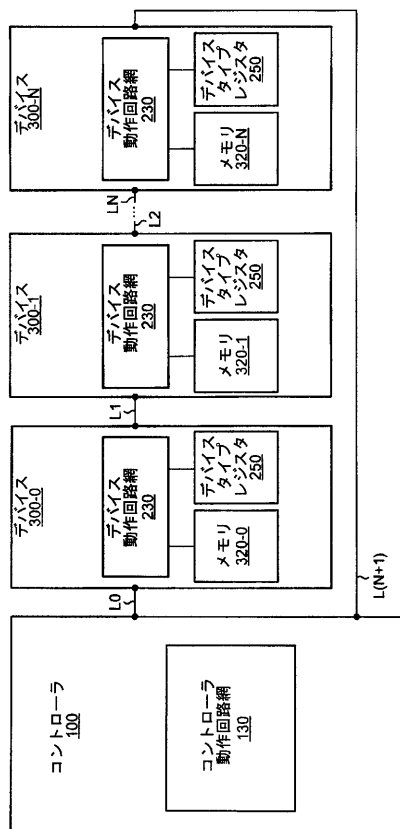


FIG. 2A

【図 2 B】

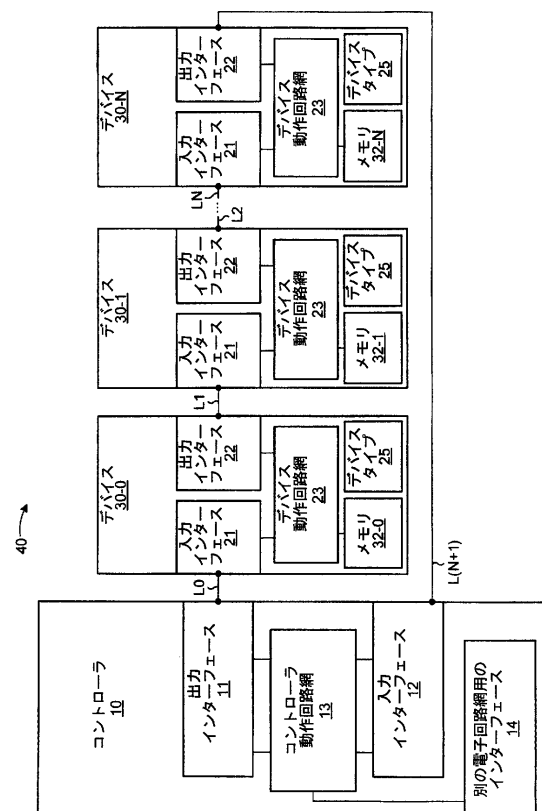


FIG. 2B

【図 2 C】

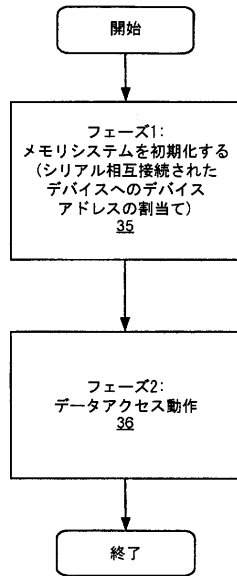


FIG. 2C

【図 3 A】

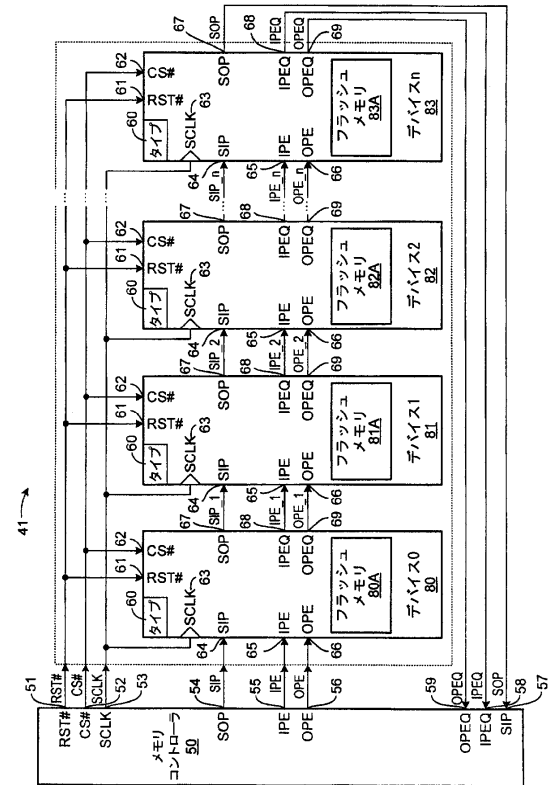


FIG. 3A

【図 3 B】

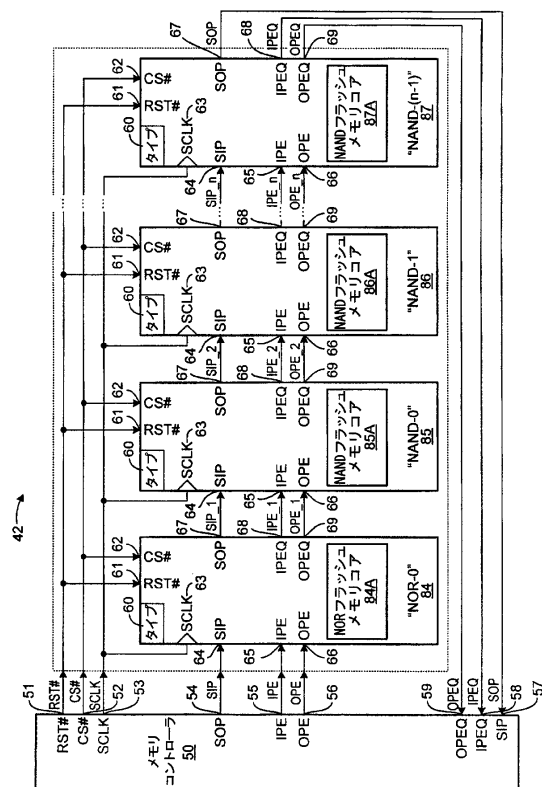


FIG. 3B

【図 3 C】

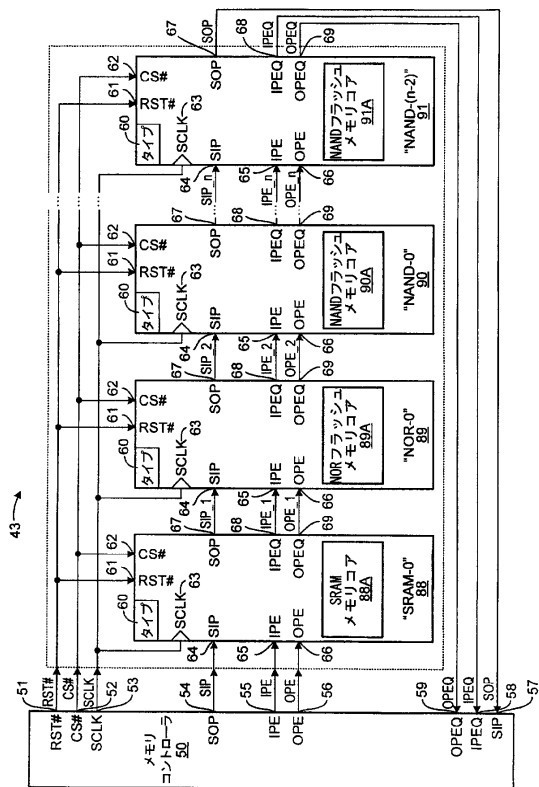


FIG. 3C

【図 3 D】

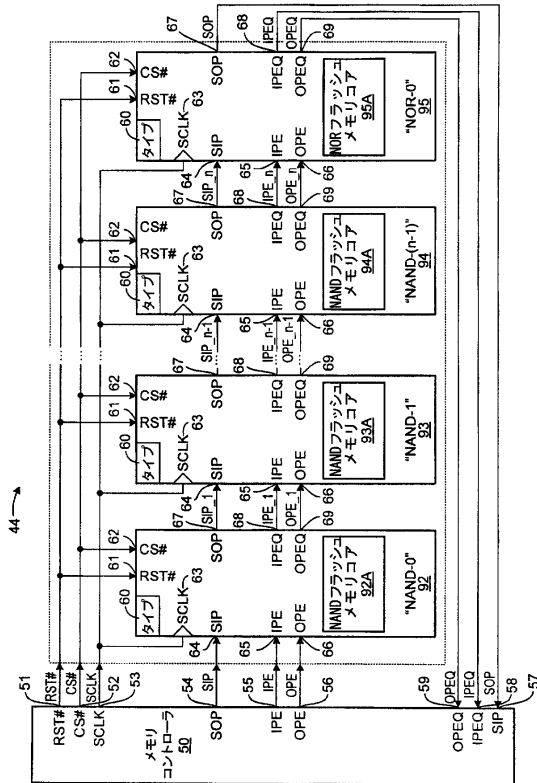


FIG. 3D

【図 3 E】

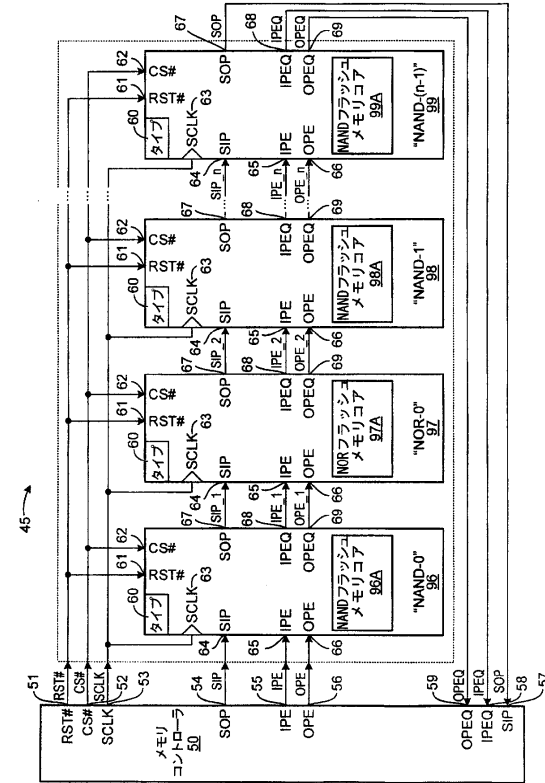


FIG. 3E

【図 3 F】

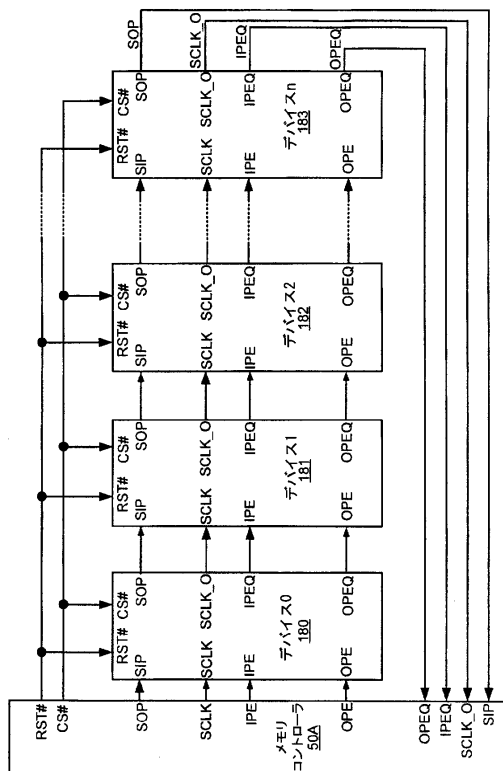


FIG. 3F

【図 3 G】

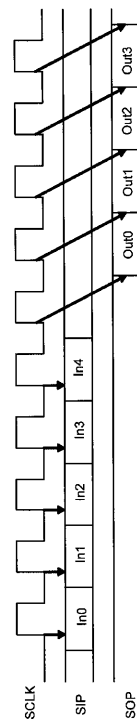
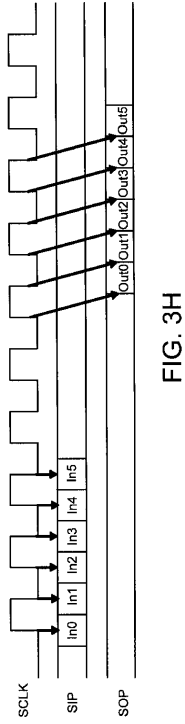
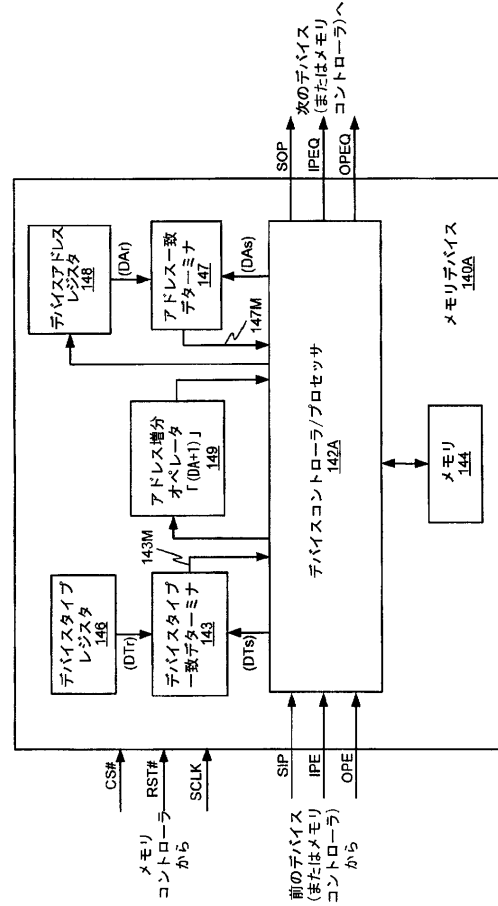


FIG. 3G

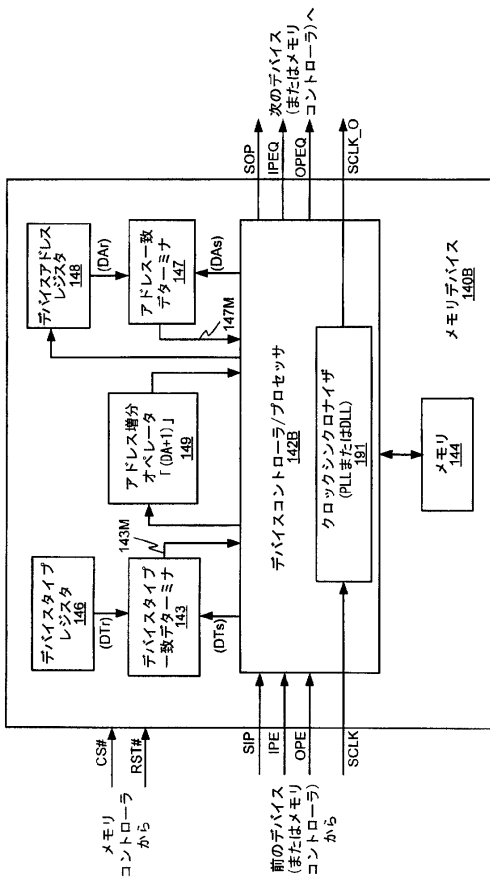
【図 3 H】



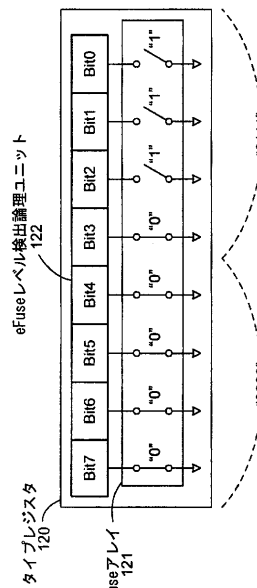
【図 4 A】



【図 4 B】



【図 5 A】



【 ㄨ 5 B 】

タイプ	16進	ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
NANDフラッシュ	00h	0	0	0	0	0	0	0	0
NORフラッシュ	01h	0	0	0	0	0	0	0	1
DRAM	02h	0	0	0	0	0	0	1	0
SRAM	03h	0	0	0	0	0	0	1	1
PSRAM	04h	0	0	0	0	0	1	0	0
DiNORフラッシュ	05h	0	0	0	0	0	1	0	1
FeRAM	06h	0	0	0	0	0	1	1	0
PCRAM	07h	0	0	0	0	0	1	1	1
シリアルEEPROM	08h	0	0	0	0	1	0	0	0
RFU									
RFU	
RFU	
MRAM	Ffh	1	1	1	1	1	1	1	1

FIG. 5B

【 図 6 A 】

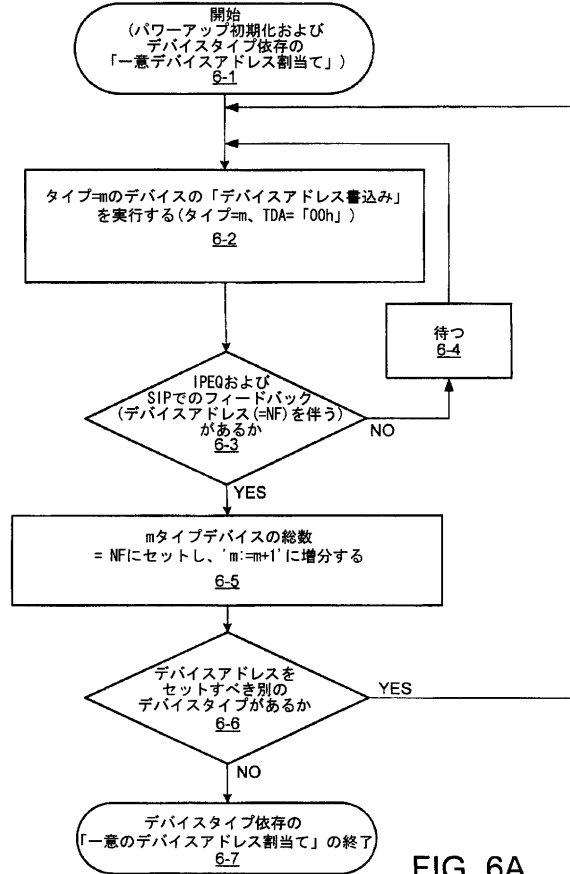


FIG. 6A

【 図 6 B 】

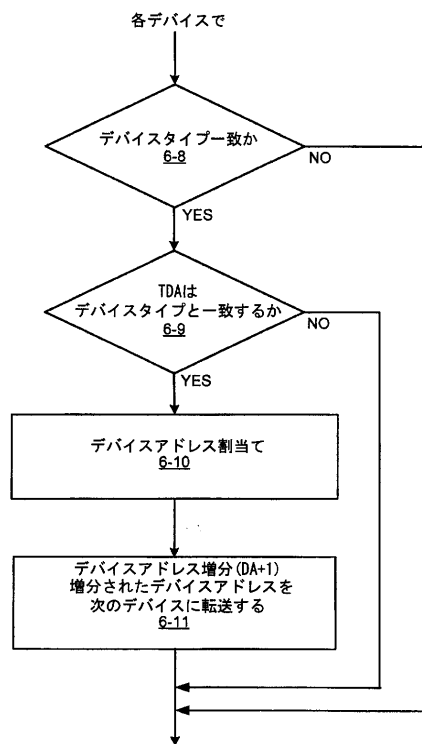


FIG. 6B

【 図 7 A 】

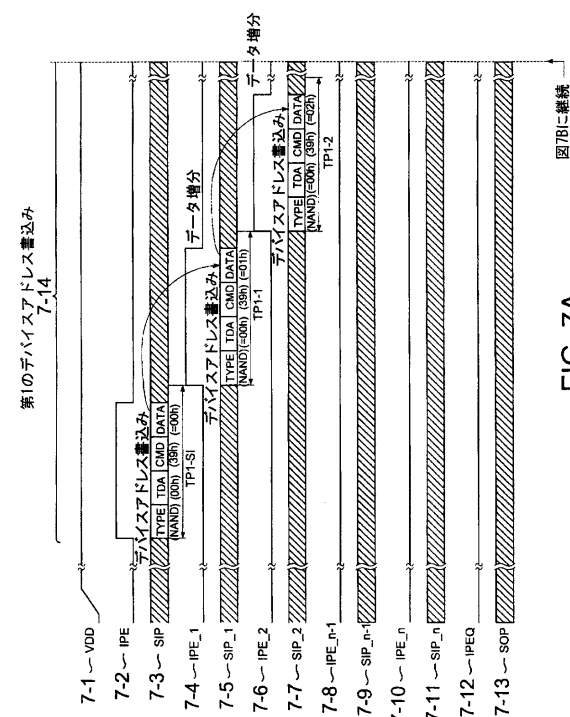


FIG. 7A

図7Bに継続

【 図 7 B 】

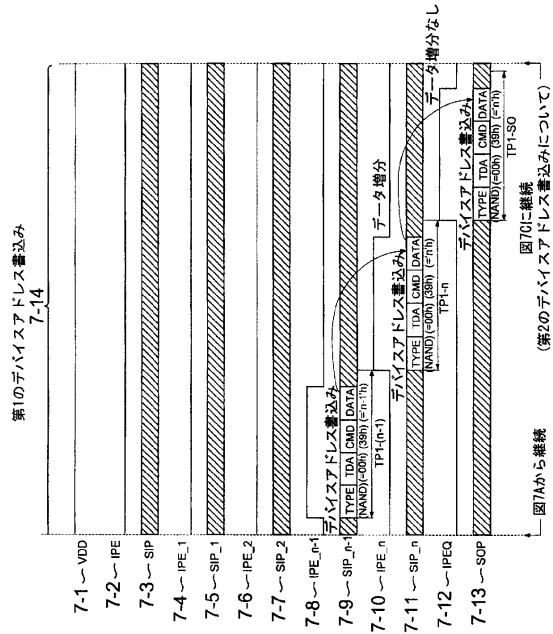


FIG. 7B

図7Cに継続
(第2のデバイスアドレス書込みについて)

【 図 7 C 】

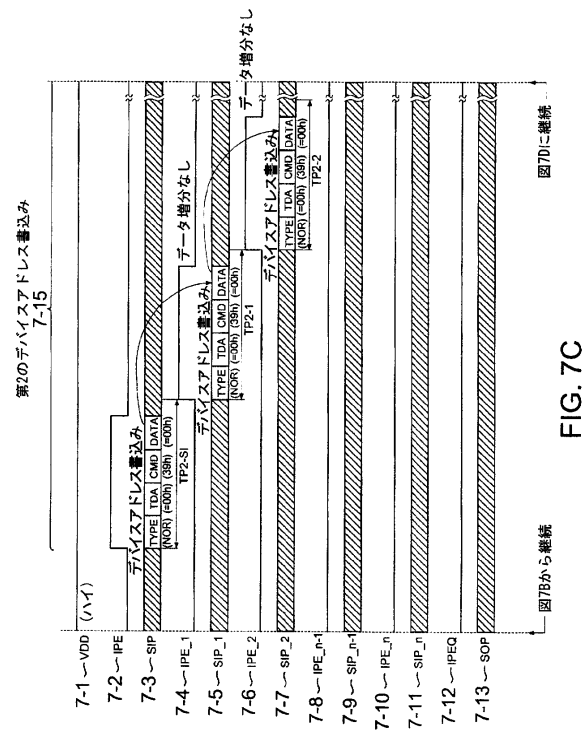


FIG. 7C

【 図 7 D 】

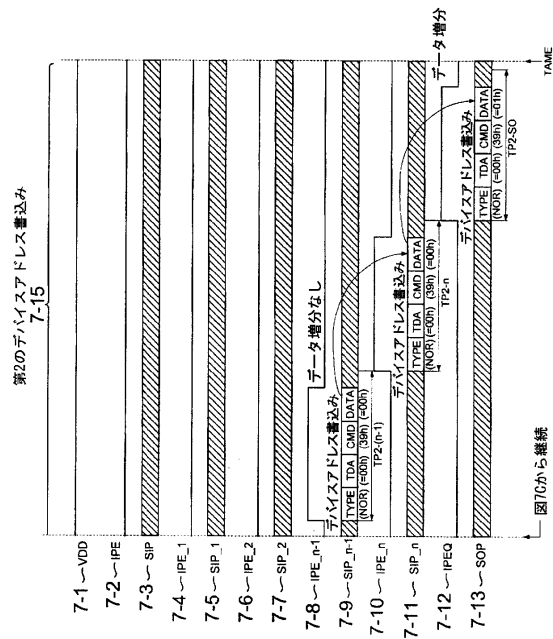


FIG. 7D

図7Cから継続

【 図 8 】

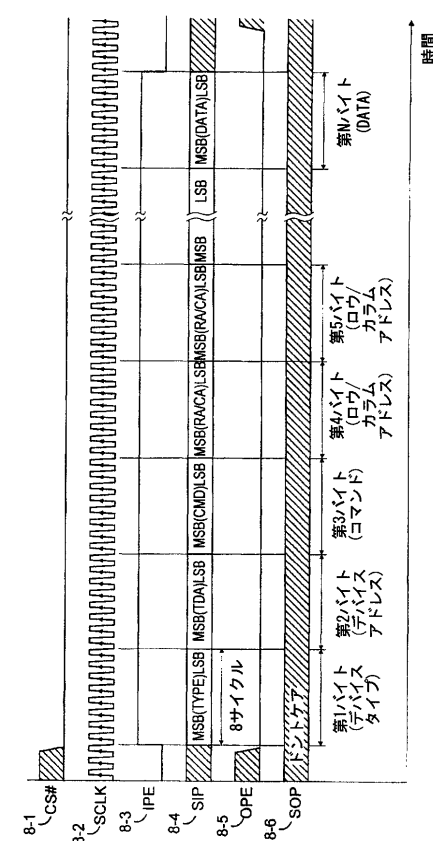


FIG. 8

時間

【図 9】

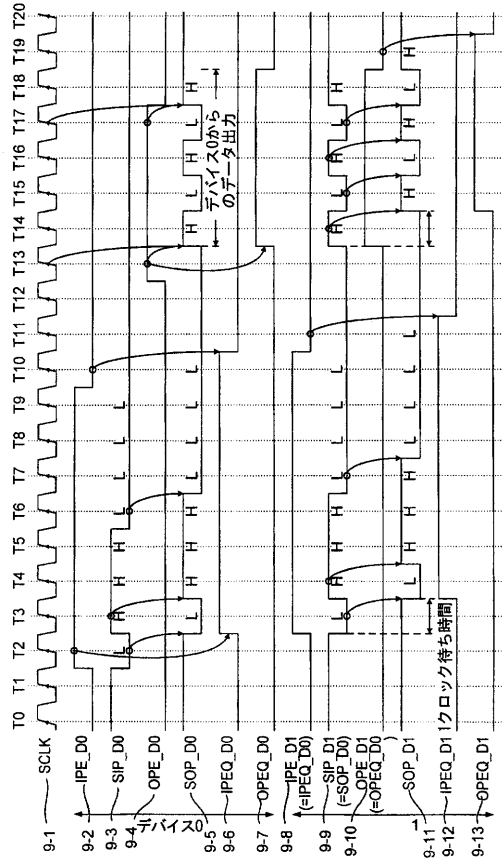


FIG. 9

【図 10】

動作	第1 バイト	第2 バイト	第3 バイト	第4 バイト	第5 バイト	第6 バイト	第7 バイト	第8 バイト	第9 バイト	...	第N バイト
読取り	TYPE	TDA	CMD	RA	RA	RA	CA	CA	-	-	-
書込み	TYPE	TDA	CMD	RA	RA	RA	CA	DATA	DATA	...	DATA
消去	TYPE	TDA	CMD	RA	RA	RA	-	-	-	-	-
状況読取り	TYPE	TDA	CMD	-	-	-	-	-	-	-	-
デバイスタイプ読取り	TYPE	TDA	CMD	-	-	-	-	-	-	-	-
構成レジスタ書込み	TYPE	TDA	CMD	DATA	-	-	-	-	-	-	-
デバイスアドレス書込み	TYPE	TDA ¹	CMD	-	-	-	-	-	-	-	-
リセット	TYPE	TDA	CMD	-	-	-	-	-	-	-	-

FIG. 10

【図 11】

ビット7	ビット6	ビット5	ビット4	ビット3	ビット2	ビット1	ビット0
TDA(ターゲットデバイスアドレス)							
第1バイト DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
RA(ロウアドレス)							
第1バイト ロウ RA[15]	RA[14]	RA[13]	ロウ	ロウ	ロウ	ロウ	RA[16]
第2バイト RA[15]	RA[14]	RA[13]	RA[12]	RA[11]	RA[10]	RA[9]	RA[8]
第3バイト RA[7]	RA[6]	RA[5]	RA[4]	RA[3]	RA[2]	RA[1]	RA[0]
CA(コラムアドレス)							
第1バイト ロウ CA[7]	CA[6]	CA[5]	ロウ	CA[11]	CA[10]	CA[9]	CA[8]
第2バイト CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	CA[0]

FIG. 11

【図 12】

動作	デバイス TYPE (1バイト)	デバイスDA (1バイト)	コマンド (1バイト)	ロウアドレス (3バイト)	コラムアドレス (2バイト)	入力データ (第1バイトから 第Nバイトまで)
ページ読取り	00h	有効	00h	有効*2	有効*2	-
ランダムデータ読取り	00h	有効	05h	-	有効	-
コピーのためのページ読取り	00h	有効	35h	有効	-	-
ページのためのターゲットアドレス入力	00h	有効	8Fh	有効	-	-
シリアルデータ入力	00h	有効	80h	有効	有効	有効
ランダムデータ入力	00h	有効	85h	-	有効	有効
ページプログラム	00h	有効	10h	-	-	-
ブロック消去	00h	有効	60h	有効	-	-
状況読取り	00h	有効	70h	-	-	-
デバイスタイプ読取り	00h	有効	90h	-	-	-
構成レジスタ書込み	00h	有効	A0h	-	-	有効(1バイト)
デバイスアドレス書込み	00h	有効*1	39h	-	-	有効(1バイト)
リセット	00h	有効	FFh	-	-	-

FIG. 12

【図 13】

動作	デバイス TYPE (1バイト)	デバイスDA (1バイト)	コマンド (1バイト)	ロウアドレス (3バイト)	コラムアドレス (2バイト)	入力データ (第1バイトから 第Nバイトまで)
読取り	01h	有効	00h	有効*2	有効*2	-
バッファへの書き込み	01h	有効	80h	有効	有効	有効
フラッシュするためのバッファを プログラムする(強制)	01h	有効	10h	-	-	-
チップ消去	01h	有効	20h	-	-	-
セクタ消去	01h	有効	21h	-	-	-
プログラム/消去一時停止	01h	有効	C0h	-	-	-
プログラム/消去再開	01h	有効	D0h	-	-	-
状態読取り	01h	有効	70h	-	-	-
デバイスタイプ読取り	01h	有効	90h	-	-	-
構成レジスタ書き込み	01h	有効	A0h	-	-	有効(1バイト)
デバイスアドレス書き込み	01h	有効*1	39h	-	-	有効(1バイト)
リセット	01h	有効	FFh	-	-	-

FIG. 13

【図 14】

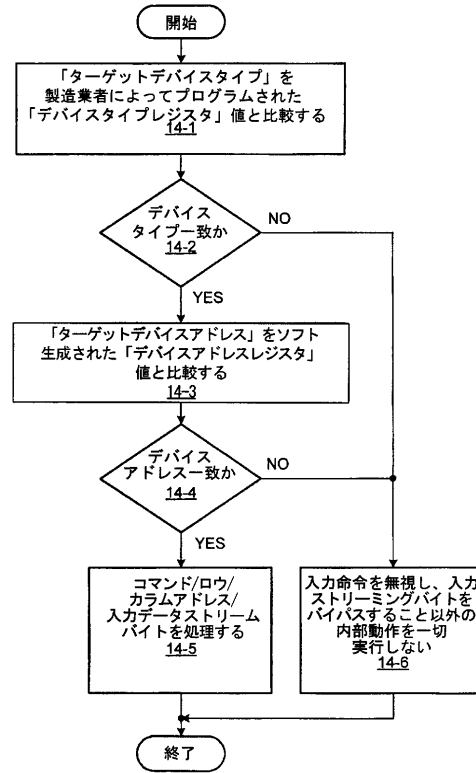


FIG. 14

【図 15 A】

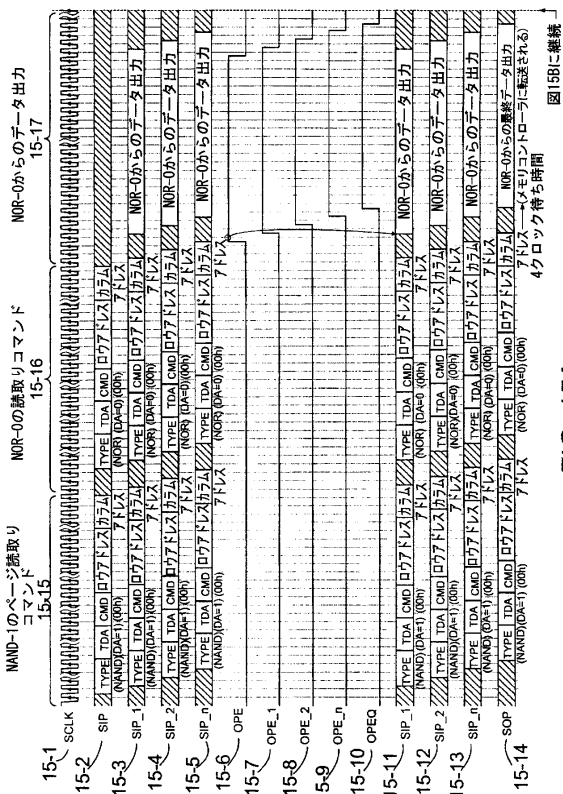


FIG. 15A

【図 15 B】

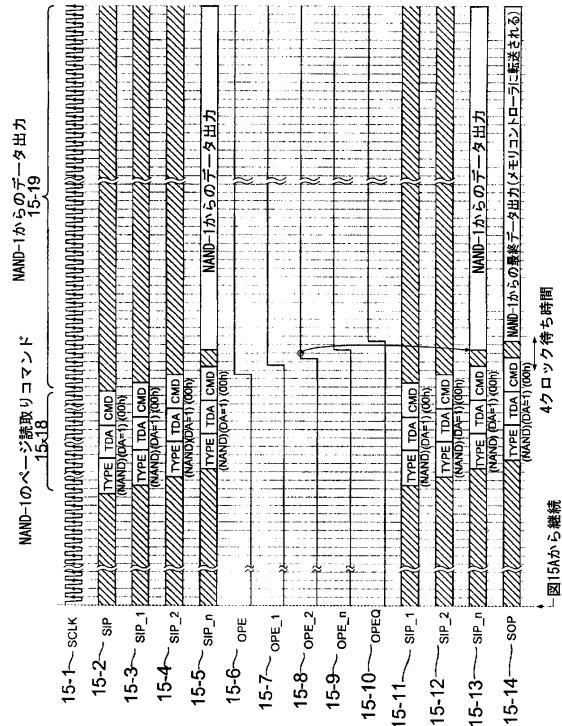


FIG. 15B

【図 16 A】

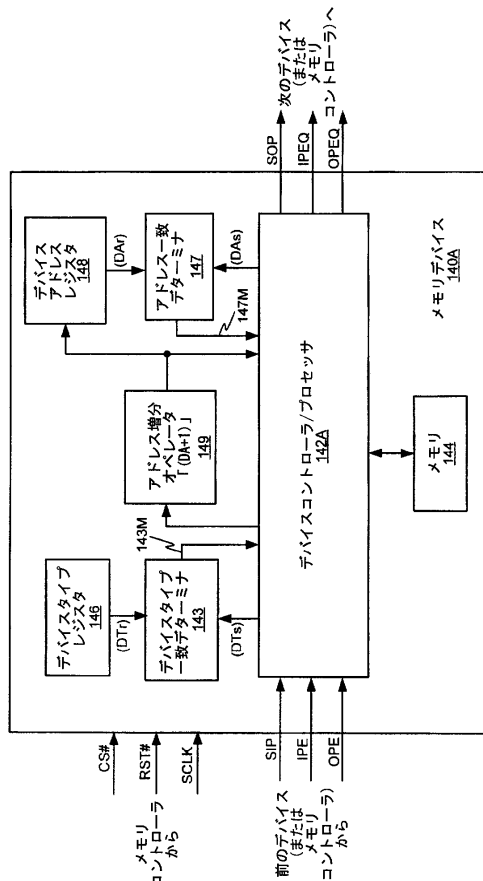


FIG. 16A

【図 16 B】

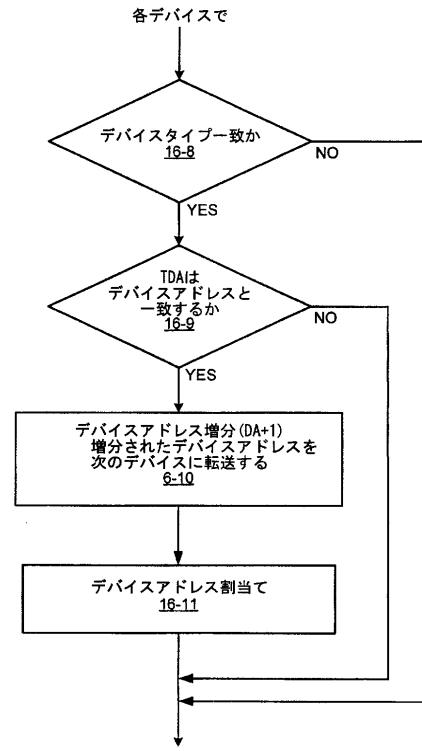


FIG. 16B

【図 16 C】

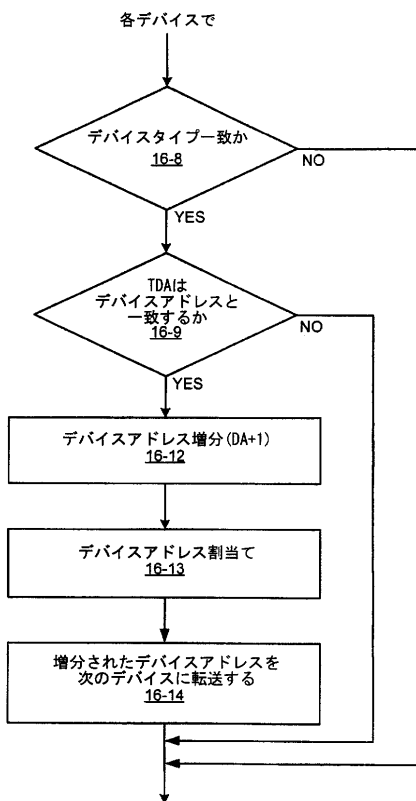


FIG. 16C

【図 17】

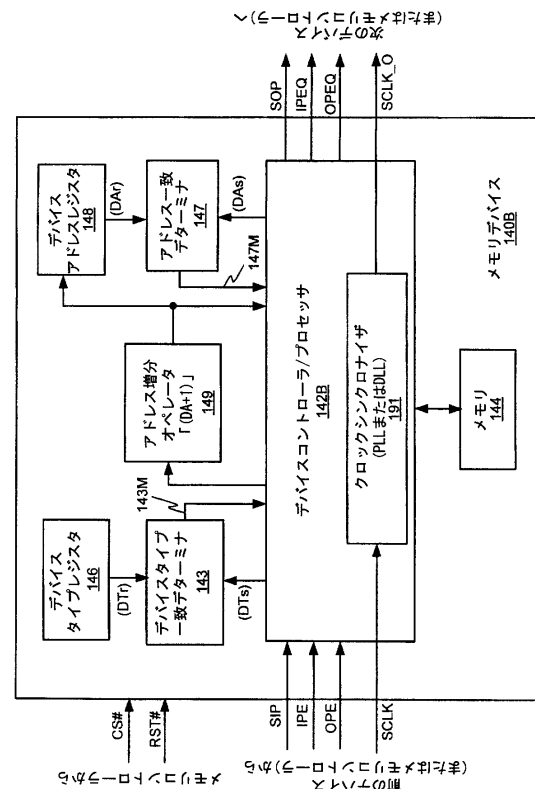


FIG. 17

【図 18】

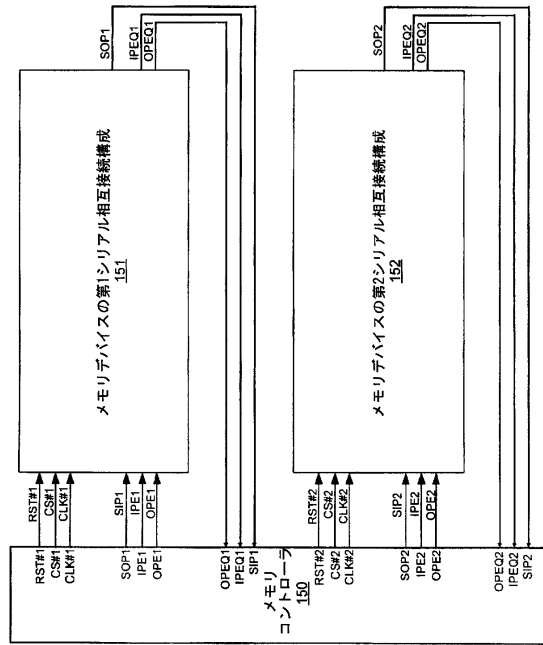


FIG. 18

【図 19A】

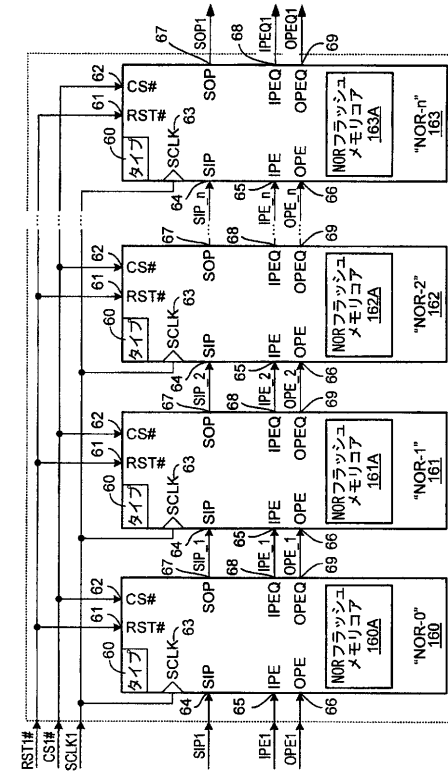


FIG. 19A

【図 19B】

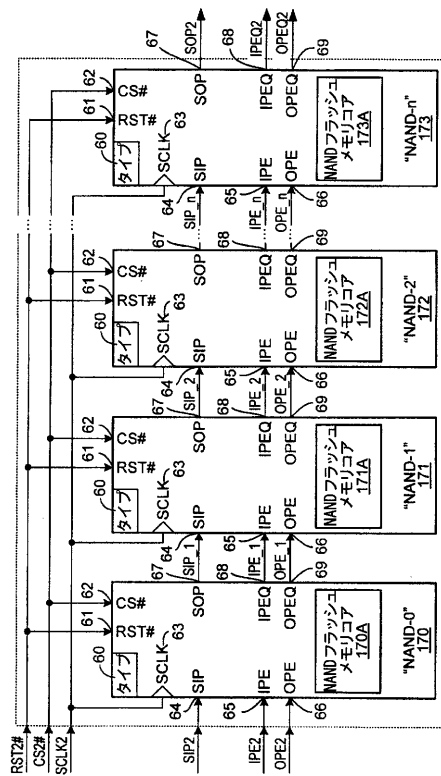


FIG. 19B

【図 20A】

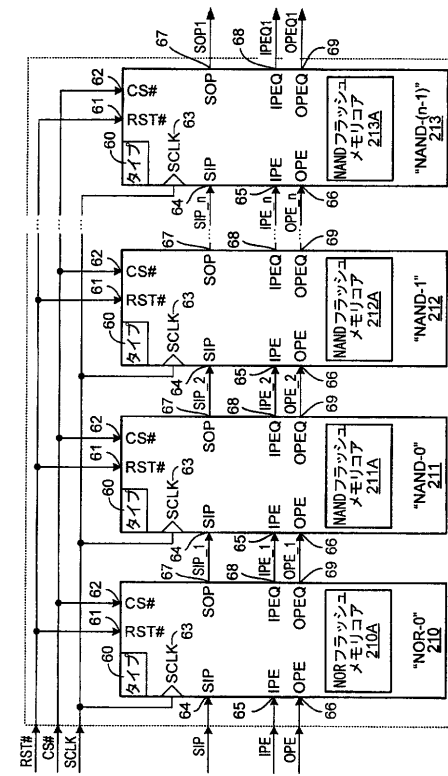


FIG. 20A

【図 20 B】

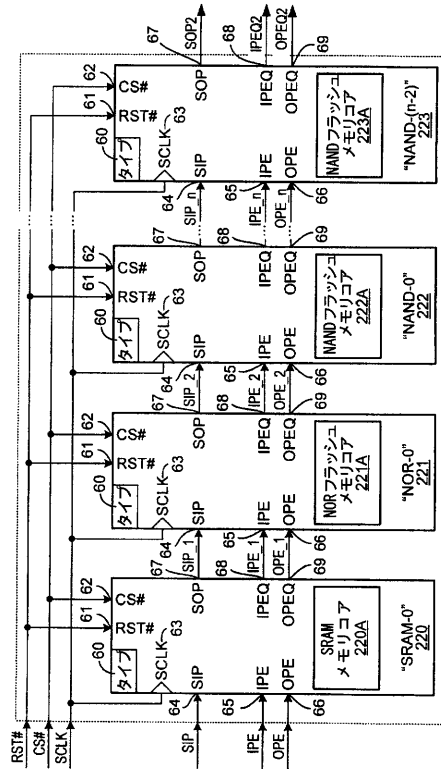


FIG. 20B

【図 21】

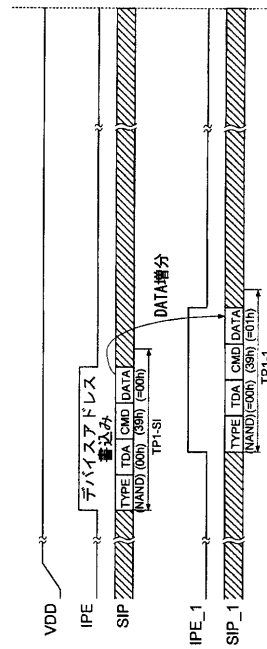


FIG. 21

【図 22】

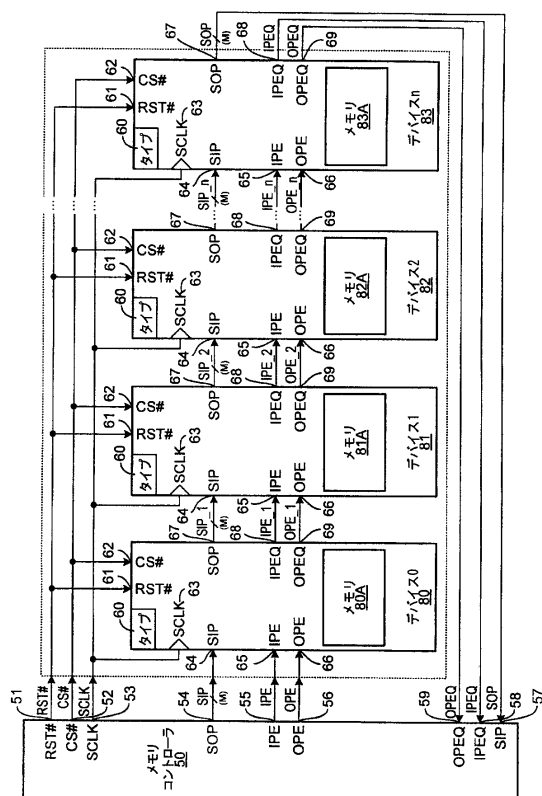


FIG. 22

【図 23】

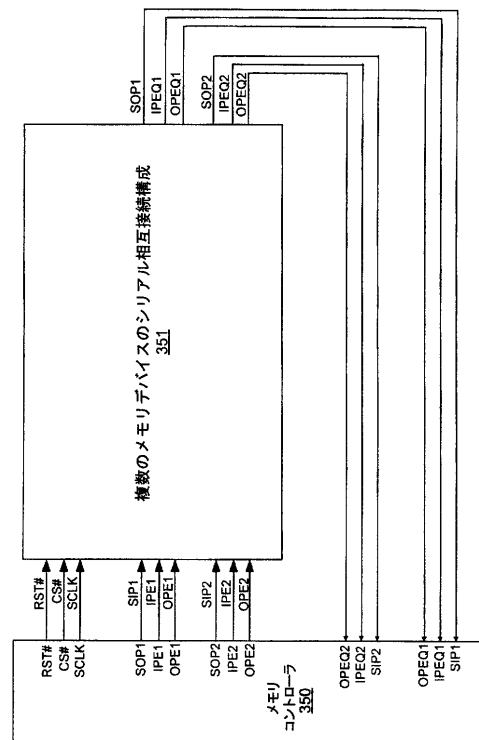


FIG. 23

【図 24】

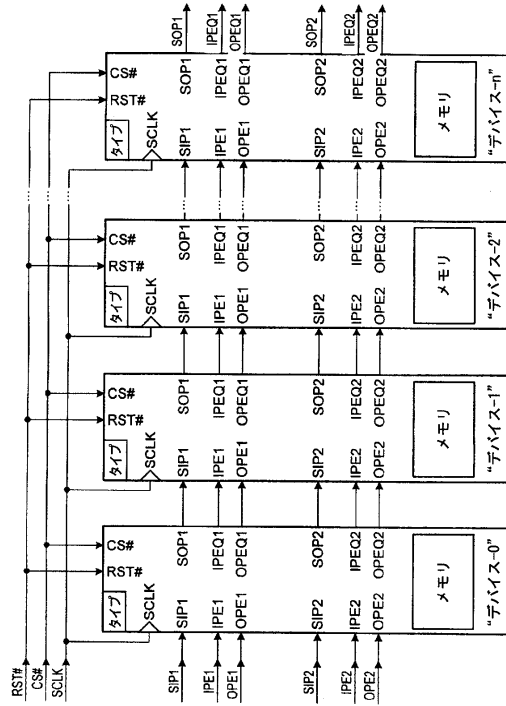


FIG. 24

【図 25】

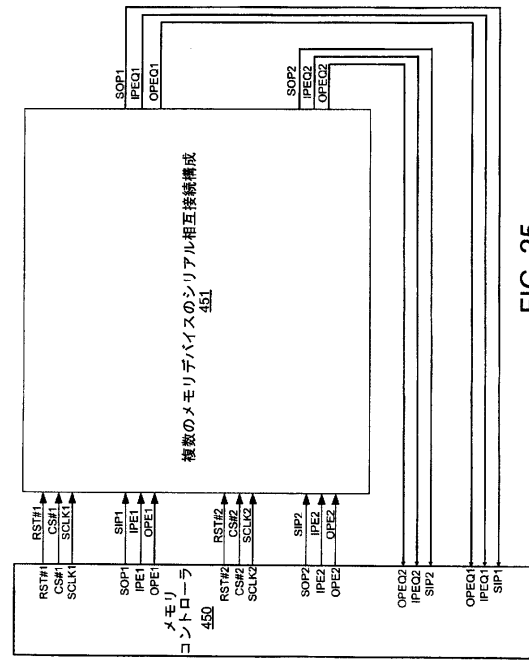


FIG. 25

【図 26】

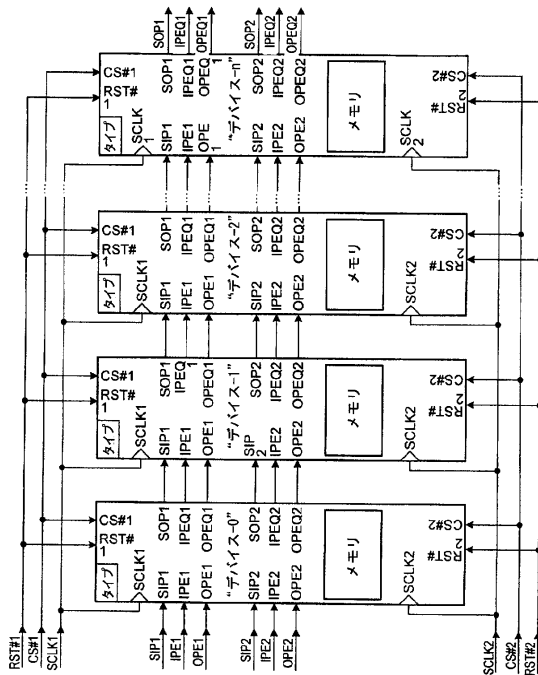


FIG. 26

フロントページの続き

(31)優先権主張番号 11/622,828

(32)優先日 平成19年1月12日(2007.1.12)

(33)優先権主張国 米国(US)

(31)優先権主張番号 11/771,241

(32)優先日 平成19年6月29日(2007.6.29)

(33)優先権主張国 米国(US)

(74)代理人 110000246

特許業務法人O F H特許事務所

(72)発明者 ハクジュン・オ

カナダ・オンタリオ・K 2 T・1 J 3・カナタ・カンピオール・クレセント・2 1

(72)発明者 ホン・ボム・ピョン

カナダ・オンタリオ・K 2 M・2 E 1・カナタ・リヴァーグリーン・クレセント・1 6

(72)発明者 ジン・キ・キム

カナダ・オンタリオ・K 2 K・3 H 6・カナタ・アイアンサイド・コート・4 6

合議体

審判長 小曳 満昭

審判官 和田 志郎

審判官 千葉 輝久

(56)参考文献 米国特許出願公開第2 0 0 4 / 0 1 4 8 4 8 2 (U S , A 1)

特開平7 - 2 0 0 4 5 8 (J P , A)

米国特許第4 3 6 0 8 7 0 (U S , A)

(58)調査した分野(Int.Cl. , D B 名)

G06F 12/00 - 12/06

G06F 13/16 - 13/18