

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3880331号
(P3880331)

(45) 発行日 平成19年2月14日(2007.2.14)

(24) 登録日 平成18年11月17日(2006.11.17)

(51) Int.Cl.

F I

G 1 1 C 11/406 (2006.01)

G 1 1 C 11/34 3 6 3 Z

G 0 6 F 12/16 (2006.01)

G 0 6 F 12/16 3 4 O S

G 1 1 C 11/403 (2006.01)

G 1 1 C 11/34 3 6 3 M

G 1 1 C 11/407 (2006.01)

G 1 1 C 11/34 3 6 2 S

請求項の数 8 (全 16 頁)

(21) 出願番号 特願2001-147028 (P2001-147028)

(22) 出願日 平成13年5月16日(2001.5.16)

(65) 公開番号 特開2002-8372 (P2002-8372A)

(43) 公開日 平成14年1月11日(2002.1.11)

審査請求日 平成13年5月16日(2001.5.16)

(31) 優先権主張番号 09/574186

(32) 優先日 平成12年5月18日(2000.5.18)

(33) 優先権主張国 米国(US)

前置審査

(73) 特許権者 390009531

インターナショナル・ビジネス・マシー
ズ・コーポレーションINTERNATIONAL BUSIN
ESS MACHINES CORPO
RATIONアメリカ合衆国10504 ニューヨーク
州 アーモンク ニュー オーチャード
ロード

(74) 代理人 100086243

弁理士 坂口 博

(74) 代理人 100091568

弁理士 市位 嘉宏

最終頁に続く

(54) 【発明の名称】 セルリフレッシュ可能なメモリ・デバイスの内容を保存する方法およびリフレッシュ制御回路

(57) 【特許請求の範囲】

【請求項1】

ホット・リセットされた際に、外部リフレッシュ信号がリイネーブルされるまで、セルフリフレッシュ可能なメモリ・デバイスの内容を保存するリフレッシュ制御回路であって、

前記リフレッシュ制御回路が設けられるデータ処理システムにバスで接続され、該システムからの信号を解釈するデコーダと、

ORゲートと、

ANDゲートと、

ラッチとを備え、

前記デコーダは、前記ORゲートの第1の入力に接続された出力と、システム・リセット信号が供給される入力とを有し、

前記ANDゲートは、前記ORゲートの出力に接続された第1の入力と、前記システム・リセット信号が供給される第2の入力と、前記データ処理システムからのクロック・イネーブル信号が供給される第3の入力とを有し、

前記ANDゲートは、前記ラッチの入力と、セルフリフレッシュ可能なメモリ・デバイスとに接続された出力を有し、

前記ラッチの出力は、前記ORゲートの第2の入力に接続されており、

システムがホット・リセットされると、CPUコアからのクロック・イネーブル信号が非活性化されることによって、前記ANDゲートの出力が非活性化されて、メモリ・デバ

イスがセルフリフレッシュ・モードにされ、次いで、前記システム・リセット信号が活性化されて、前記デコーダの、前記ORゲートの第1の入力に接続された出力が非活性化され、これによって、前記クロック・イネーブル信号が活性化された後にも、前記ANDゲートの出力が非活性に維持される、リフレッシュ制御回路。

【請求項2】

前記データ処理システムは、システム・オン・ア・チップであり、前記クロック・イネーブル信号は、前記システム・オン・ア・チップのコアによって発生される、請求項1に記載のリフレッシュ制御回路。

【請求項3】

前記システム・オン・ア・チップは、特定用途向け集積回路である、請求項2に記載のリフレッシュ制御回路。

【請求項4】

前記システム・オン・ア・チップは、プログラマブル・アレイ・ロジック・チップである請求項2に記載のリフレッシュ制御回路。

【請求項5】

前記システム・オン・ア・チップは、フィールド・プログラマブル・アレイ・ロジック・チップである請求項2に記載のリフレッシュ制御回路。

【請求項6】

前記セルフリフレッシュ可能なメモリ・デバイスは、シンクロナス・ダイナミック・ランダム・アクセス・メモリである、請求項1に記載のリフレッシュ制御回路。

【請求項7】

前記デコーダがレジスタを備え、該レジスタは単一レジスタ・ビット(CKE__Control)を有し、該ビットは前記システム・リセット信号が該レジスタのクリアに入力されることによってローにセットされて、前記ORゲートの第1の入力に接続された出力が非活性化され、及び、所定のソフトウェアによりハイにセットされると前記ORゲートの第1の入力に接続された出力が活性化される、請求項1～6のいずれか1項記載のリフレッシュ制御回路。

【請求項8】

バスに接続されたプロセッサと、
前記バスに接続されたメモリ・デバイス・コントローラと、
前記メモリ・デバイス・コントローラに接続されたセルフリフレッシュ可能なメモリ・デバイスとを備えたデータ処理システムにおいて、
前記バスと前記メモリ・デバイス・コントローラとに接続されたリフレッシュ・コントローラをさらに含み、該リフレッシュ・コントローラが請求項1～7のいずれか1項記載のリフレッシュ制御回路を有することを特徴とする、データ情報処理システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

この発明は、一般的には、コンピュータ・メモリ、具体的には、システムのリセットの間、同期ランダム・アクセス・メモリの内容を保存することに関する。

【0002】

【従来の技術】

通常、メモリ・コントローラがリセットされるとき、メモリの現在の状態を、保存する必要はなかった。しかし、いくつかの設計アプリケーションは、メモリの内容がリセットの間に保存されて、リセット後もメモリの内容が再びアクセスできるようにすることを必要とする。例えば、ハング(すなわちフローズン)したシステムの場合、オペレータは、システムをパワーオフすることなしに、リセット・ボタンを押すことを望むかもしれない。システムが再びブート・アップすると、オペレータは、前のシステム・ハングの原因を理解しようとするために、メモリの内容を照会するオプションを有することを望むかもしれ

10

20

30

40

50

ない。あるいはまた、オペレータは、システムがフリーズする前に保存されなかったデータをメモリから回復させたいかもしれない。もちろん、これらの場合、ブート・コードは、ブート・コードがシステム・メモリに書込む前に、この診断モードに入るための手段を与えなければならない。

【 0 0 0 3 】

シンクロナス・ダイナミック・ランダム・アクセス・メモリ (S D R A M) において、メモリ内容を保存する際に、克服すべき1つの重要な問題は、リフレッシュの維持である。S D R A M は、他の種類の S D R A M の様に、リフレッシュ・サイクルを正規の周波数で発生させることを必要とする。技術およびメモリ・サイズに依存する特定の最大時間が、リフレッシュ・サイクルとリフレッシュ・サイクルとの間の時間を超えると、メモリは破壊されるであろう。メモリ・コントローラがリセットされると、その内部レジスタのすべてが、デフォルト状態に戻る。このことは、メモリ・コントローラが、ソフトウェアによって再初期化されるまで、メモリのリフレッシュを停止することを意味している。リセットの開始とメモリ・コントローラの再初期化との間の時間は、可変であり、リフレッシュ・サイクル間の最大許容時間よりも通常長いので、この時間の間に、メモリが破壊する可能性が高い。

【 0 0 0 4 】

【 発明が解決しようとする課題 】

メモリ・コントローラが、特定種類のリセットを検知することを簡単に除外することは、選択事項ではない。システム・ハングが、多くの事象によって引き起こされるので、マシンを成功的にリブートするためには、すべてのリセットの間に、メモリ・コントローラを含むすべてのハードウェアを、デフォルト状態に復元させることが必要である。システム・メモリのデフォルト状態は、ランダムに始まるので、その内容を保存することは、リブート・プロセスを妨げない。したがって、特定種類のリセット信号の際に、新しいアーキテクチャおよび既存のアーキテクチャに、メモリをリフレッシュする機能を付加する装置および方法が望まれる。

【 0 0 0 5 】

【 課題を解決するための手段 】

この発明は、データ処理システムのリセットの期間中、セルフリフレッシュ可能なメモリ・デバイスの内容を保持する方法、システム、および装置を提供する。一実施形態では、リフレッシュ・コントローラは、データ処理システムがリセットされていることの指示を受取る。必要ならば、リフレッシュ・コントローラは、メモリ・デバイスがセルフリフレッシュ・モードに保たれるように、メモリ・コントローラからメモリ・デバイスへの信号を変更する。リフレッシュ・コントローラは、データ処理システムが外部リフレッシュ信号をリイネーブルするまで、メモリ・デバイスをセルフリフレッシュ・モードに保つ。

【 0 0 0 6 】

【 発明の実施の形態 】

図1は、この発明を実施することのできるデータ処理システムのブロック図を示す。データ処理システム100は、周辺コンポーネント相互接続 (peripheral component interconnect ; P C I) ローカルバス・アーキテクチャを採用している。図示の例は、P C I バスを採用しているが、M i c r o C h a n n e l および I S A のような他のバス・アーキテクチャを用いることもできる。プロセッサ102と、シンクロナス・ダイナミック・ランダム・アクセス・メモリ (synchronous dynamic random access memory ; S D R A M) メインメモリ・コントローラ104とが、P C I ブリッジ108を介してP C I ローカルバス106に接続されている。P C I ブリッジ108は、プロセッサ102のためのキャッシュ・メモリを有することもできる。S D R A M メイン・メモリ・コントローラ104は、また、S D R A M メイン・メモリ136およびリフレッシュ・コントローラ134に接続される。

【 0 0 0 7 】

P C I ローカルバス106への追加の接続は、直接コンポーネント相互接続またはアドイ

10

20

30

40

50

ン (a d d - i n) ボードによって行うことができる。図示の例ではローカル・エリア・ネットワーク (L A N) アダプタ 1 1 0 , S C S I ホストバス・アダプタ 1 1 2 , 拡張バス・インタフェース 1 1 4 は、直接コンポーネント接続によって、P C I ローカルバス 1 0 6 に接続される。対照的に、オーディオ・アダプタ 1 1 6 , グラフィックス・アダプタ 1 1 8 , オーディオ/ビデオ・アダプタ (A / V) 1 1 9 は、拡張スロットに挿入されたアドイン・ボードによって、P C I ローカルバス 1 0 6 に接続される。拡張バス・インタフェース 1 1 4 は、I S A バス 1 3 5 を介して、キーボードおよびマウス・アダプタ 1 2 0 , モデム 1 2 2 , 付加メモリ 1 2 4 への接続を与える。図示の例では、S C S I ホストバス・アダプタ 1 1 2 は、ディスク・ドライバ 1 2 6 , テープ・ドライバ 1 2 8 , C D - R O M ドライブ 1 3 0 , デジタル・ビデオ・ディスク・リードオンリ・メモリ・ドライブ (D V D - R O M) 1 3 2 への接続を与える。典型的な P C I ローカルバス・インプリメンテーションは、3 個または 4 個の拡張スロットまたはアドイン・コネクタを接続する。

10

【 0 0 0 8 】

オペレーティング・システムは、プロセッサ 1 0 2 上で実行され、図 1 のデータ処理システム 1 0 0 内の種々のコンポーネントの調整および制御を与えるために用いられる。オペレーティング・システムは、International Business Machines Corporationから得られる O S / 2 のような市販のオペレーティング・システムとすることができる。“ O S / 2 ” は、International Business Machines Corporationの登録商標である。J a v a (R) のようなオブジェクト指向プログラミング・システムは、オペレーティング・システムと一緒に実行されて、データ処理システム 1 0 0 上で実行される J a v a (R) プログラムまたはアプリケーションからオペレーティング・システムへの呼出しを与えることができる。オペレーティング・システムに対する命令と、オブジェクト指向オペレーティング・システムと、アプリケーションまたはプログラムとは、ハードディスク・ドライブ 1 2 6 のようなストレージ・デバイスに配置され、プロセッサ 1 0 2 により実行するために、メインメモリ 1 3 6 にロードすることができる。

20

【 0 0 0 9 】

S D R A M メインメモリ・コントローラ 1 0 4 は、データ処理システムからの特定のコマンドに応じて、S D R A M メインメモリ 1 3 6 をセルフリフレッシュ・モードにする能力を有している。この発明は、データ処理システムが、“ ボタン (b u t t o n) ” リセットのような特定種類のリセット信号に応じて、S D R A M コントローラをセルフリフレッシュ・モードにする必要とする。S D R A M メインメモリ 1 3 6 は、メモリの内容が、周期的にリフレッシュされる必要とする。S D R A M メインメモリ 1 3 6 が、長期間にわたってリフレッシュされなければ、S D R A M メインメモリ 1 3 6 内の内容は、失われるであろう。セルフリフレッシュ・モードの間、S D R A M メインメモリ 1 3 6 は、外部リフレッシュ・サイクルの実行を必要としないが、S D R A M メインメモリ 1 3 6 は、この間、アクセスできない。しかし、S D R A M メインメモリ 1 3 6 内の内容は、データ処理システムが、S D R A M メインメモリ 1 3 6 の外部リフレッシュをリイネブルするまで、保存される。

30

【 0 0 1 0 】

また、リフレッシュ・コントローラ 1 3 4 は、S D R A M メインメモリ・コントローラ 1 0 4 に接続されている。リフレッシュ・コントローラ 1 3 4 は、S D R A M メインメモリ 1 3 6 が、例えばデータ処理システムのリブートの間のような、割込み期間の間、セルフリフレッシュ・モードを早々と終了することを阻止する追加のコンポーネントである。S D R A M メインメモリ 1 3 6 が、セルフリフレッシュ・モードを早々と終了することを阻止することによって、リブートまたは他システム割込みの前の S D R A M メインメモリ 1 3 6 の内容は、リブート・プロセスが完了した後に、データ処理システムに利用できる。

40

【 0 0 1 1 】

リフレッシュ・コントローラ 1 3 4 の重要性および機能の理解を助けるために、以下の例を考察する。ユーザが、データ処理システム 1 0 0 上の“ ボタン ” リセットを押したとする。リセット信号は、データ処理システム 1 0 0 のリセット・ロジックに直接供給される

50

よりはむしろ、割込みに関係づけられるので、割込みが行われる。割込みサービス・ルーチンは、最初に、SDRAMメインメモリ136のすべてのSDRAMバンクをアイドルする。次に、第2のステップで、SDRAMメインメモリ・コントローラ104は、データ処理システムで実行されている割込みサービス・ルーチンの要求時に、SDRAMメインメモリ136を、セルフリフレッシュ・モードにする。最後に、第3のステップで、割込みサービス・ルーチンは、ソフトウェア（データ処理システム内で通常利用できるメカニズム）によってシステム・リセットをアクチベートする。第2のステップにおいて、SDRAMメインメモリ・コントローラ104は、SDRAMメモリをセルフリフレッシュ状態にするコマンドをSDRAMメモリに送った後、クロック・イネーブル（CKE）をロー状態にする。しかし、リフレッシュ・コントローラ134が使用されない、割込みサービス・ルーチンは、第3のステップで、CKEを、デフォルト・ハイ（イネーブルされた）状態に戻す。というのは、システム・リセットは、すべてのデバイスをデフォルト状態に戻すからである。CKEがハイに戻されると、SDRAMメインメモリ136は、セルフリフレッシュ・モードを終了して、ブート・コードが、SDRAMメインメモリ・コントローラ104を再初期化する前に、データを失うであろう。したがって、リフレッシュ・コントローラ134によって与えられる付加回路は、リセット信号がリリースされるときに、CKEをローに保持することが要求される。CKEは、ブート・コードが、SDRAMメインメモリ・コントローラ104の再初期化を終了させるまで、インアクティブに保持されなければならない。

10

【0012】

20

当業者は、図1のハードウェアが、インプリメンテーションに応じて変化し得ることがわかるであろう。例えば、光ディスクドライブなどのような他の周辺デバイスを、図1に示されたハードウェアに加えて、あるいはハードウェアに代えて、用いることができる。図示の例は、この発明に対して構造的な限定を加えることを意味するものではない。例えば、この発明のプロセスは、マルチプロセッサ・データ処理システムに適用することができる。

【0013】

図2は、この発明を実施できるシステム・オン・ア・チップ（System-On-a-Chip）のブロック図を示す。システム・オン・ア・チップ200は、プロセッサ・ローカルバス（Processor Local Bus；PLB）アーキテクチャを用いている。しかし、他のバス・アーキテクチャをさらに用いることができる。プロセッサ・ローカルバス224は、接続されるCPU214のような高性能、高帯域幅のブロックをホストする。図示の例では、CPU214は、Power-PC405（登録商標）のCPUコアである。また、プロセッサ・ローカルバス224に、ROM周辺コントローラ228と、SDRAMコントローラ226と、ダイレクト・メモリ・アクセス（DMA）コントローラ220とが接続される。DMAコントローラ220は、CPU214を用いることなしに、メモリからメモリへデータを転送する特殊回路または専用マイクロプロセッサである。PCIブリッジ230は、プロセッサ・ローカルバス224上のデバイスに対しインタフェースを与え、システム・オン・ア・チップ200外のPCIバス上のPCIデバイスと通信する。PLBアービタ222は、PCIプロセッサ・ローカルバス224に対するリクエストを受取り、次にどのデバイスが、PCIプロセッサ・ローカルバス224へのアクセスを受取るかを決定する。

30

40

【0014】

システム・オン・ア・チップ200は、また、直列ポート202～206とイーサネット（R）（登録商標）212のような低データ転送速度周辺装置をホストするオン・チップ周辺（OPB）バス210を有する。直列ポート202～206は、システム・オン・ア・チップ200と、システム・オン・ア・チップ200外のデバイスおよびコンポーネントとの間の接続を与える。イーサネット（R）212は、外部ネットワークへの接続を与える。OPBアービタ208は、OPBバス210へのリクエストを受取り、次にどのデバイスがアクセスを受取るかを決定する。OPBバス210は、OPBブリッジ218および

50

DMAコントローラ220を経て、プロセッサ・ローカルバス224に接続される。

【0015】

SDRAMコントローラ226は、SDRAMメモリ234とリフレッシュ・コントローラ236とに接続される。SDRAMコントローラ226は、データ処理システムからのリクエストに応じて、SDRAMメモリ234をセルフリフレッシュ・モードにする能力を有している。データ処理システムは、“ボタン(button)”リセットのような特定種類のリセット信号に応じて、SDRAMコントローラが、SDRAMをセルフリフレッシュ・モードにすることを要求する。SDRAMメモリ234は、メモリの内容が、周期的にリフレッシュされることを要求する。セルフリフレッシュ・モード中、SDRAMメモリ234は、外部リフレッシュ・サイクルが実行されることを要求しない。しかし、SDRAMメモリ234内の内容は、システム・オン・ア・チップ200が、SDRAMメモリ134の外部リフレッシュをイネーブルするまで、保存される。

10

【0016】

SDRAMコントローラ226には、また、リフレッシュ・コントローラ236が接続される。リフレッシュ・コントローラ236は、SDRAMメモリ234が、例えばシステムのリブートの間のような、割込みの期間の間に、セルフリフレッシュ・モードを早々と終了させることを阻止する追加のコンポーネントである。SDRAMメモリ234が、セルフリフレッシュ・モードを早々と終了させることを阻止することによって、リブート、または他システムの割込みの前に、SDRAMメモリ234の内容を、リブート・プロセスが終了した後に、システム・オン・ア・チップ200に対して利用できる。リフレッシュ・コントローラ236は、図1のリフレッシュ・コントローラ134と同様に動作する。

20

【0017】

システム・オン・ア・チップ200は、この発明を実施することのできる特定用途向け集積回路(application specific integrated circuit ; ASIC)のようなシステム・オン・ア・チップの一例である。システム・オン・ア・チップ200は、単に例示として与えられており、構成上の制限を課することを意味していない。システム・オン・ア・チップ200は、図2に示されない他のコンポーネントを有することができる。

【0018】

図3は、外部リフレッシュ・サイクルがイネーブルされるまで、SDRAMメモリをセルフリフレッシュ・モードに保持するこの発明のリフレッシュ制御回路を示す図である。リフレッシュ制御回路300は、図2のリフレッシュ・コントローラ236として、あるいは、以下に説明するようなわずかな変形で、図1のリフレッシュ・コントローラ134として実現することができる。読取り/書込みクロック・イネーブル制御ビット信号CKE__Control__Bitを実現するバス・ロジックは、デコーダ302内にあるが、図示していない。リフレッシュ制御回路300は、また、プログラマブル・アレイ・ロジック(programmable array logic ; PAL)のようなプログラマブル・ロジック・デバイス内に設けることができる。なお、プログラマブル・アレイ・ロジックは、プログラマブルANDゲートおよび定義済みORゲートのアレイを有するプログラマブル・ロジック・チップのタイプのもの、あるいはSDRAMコントローラを有するチップ外のフィールド・プログラマブル・アレイ(field programmable array ; FPGA)である。

30

40

【0019】

リフレッシュ制御回路300は、デコーダ302, ORゲート304, ANDゲート306, Dフリップフロップ・ラッチ308を有している。デコーダ302は、リフレッシュ制御回路300とのソフトウェア通信をデコードするシステム・バスの“デコーダ”を含んでいる。デコーダ302は、また、CKE__Controlと呼ばれる単一レジスタ・ビットを有している。このビットは、デコーダ302を介して、システムにより書込むことのできる(およびインプリメンテーションに応じて、読取ることもできる)。デコーダ302から出力されたCKE__Controlは、ORゲート304の入力に供給される。ORゲート304の出力は、ANDゲート206の入力に供給される。ANDゲート3

50

06の入力には、アクティブ・ロー・システムリセット信号 `System_Reset_N` と、コアからのクロック・イネーブル信号 `CKE_From_Core` とが供給される。ANDゲート306の出力は、Dフリップフロップ・ラッチ308の入力に供給され、チップ外に送出される。Dフリップフロップ・ラッチ308の出力は、ORゲート304の第2の入力に供給される。ANDゲート306の出力 `CKE_Off_Chip` は、また、SDRAMメモリ（図示せず）に供給されて、リセット中にSDRAMメモリの内容を保持する。

【0020】

SDRAMバス上に特定の電子コマンドを発行し、およびクロック・イネーブル出力信号 `CKE_Off_Chip` をインアクティブ・ロー状態にすることによって、SDRAMメモリは、SDRAMコントローラにより、セルフリフレッシュ・モードにされる。通常動作中、`System_Reset_N` はハイ（インアクティブ）であり、クロック・イネーブル制御信号 `CKE_Control` はハイ（アクティブ）であり、コアからのクロック・イネーブル信号 `CKE_From_Core` はハイ（アクティブ）であり、したがってクロック・イネーブル出力信号 `CKE_Off_Chip` およびフィードバック信号 `Feedback` はハイである。このモードでは、コアは、そのアクティブ・ハイ・イネーブル信号 `CKE_From_Core` をアサートおよびデアサートし、クロック・イネーブル出力信号 `CKE_Off_Chip` に、影響を与えることができる。

【0021】

ロジック306の唯一つのレベルが、コアからのクロック・イネーブル信号 `CKE_From_Core` と、クロック・イネーブル出力信号 `CKE_Off_Chip` との間に、付加された。したがって、最近の特定用途向け集積回路（ASIC）において解決すべき困難なタイミングの問題は、存在しない。

【0022】

“ボタン”リセットまたは他の種類の特殊リセットがアサートされると、リセット割込みサービス・ルーチンが、SDRAMコントローラを、セルフリフレッシュ・モードにする。このことは、コアからのクロック・イネーブル信号 `CKE_From_Core`、したがってクロック・イネーブル出力信号 `CKE_Off_Chip` を、インアクティブ（ロー）になるようにする。割込みサービス・ルーチンの終わりは、リセット信号 `System_Reset_N` をローにアクティベートする。このことは、リセット中にコアからのクロック・イネーブル信号 `CKE_From_Core` に何が生じようとも、クロック・イネーブル出力信号 `CKE_Off_Chip` をローに保持する。また、クロック・イネーブル制御ビット信号 `CKE_Control_bit` は、リセット中、デコード302によってローにセットされる。これは、リセット信号 `System_Reset_N` を、`CKE_Control` レジスタの“クリア（clear）”入力に供給することにより達成される。

【0023】

リセット期間が終わると、リセット信号 `System_Reset_N` は、ハイに戻り、コアからのクロック・イネーブル信号 `CKE_From_Core` はまたハイであるが、クロック・イネーブル出力信号 `CKE_Off_Chip` は、クロック・イネーブル制御ビット信号 `CKE_Control_bit` がローであり、フィードバック信号 `Feedback` がローであるが故に、クロック・イネーブル出力信号 `CKE_Off_Chip` は、ローに留まる。それは、クロック・イネーブル制御ビット信号 `CKE_Control_bit` がハイになるまで、ローに留まるであろう。“パワーオン（Power-on）”リセットによって、システムがリセットされたならば、SDRAMバス・トランザクションを実行する初期化シーケンスが、SDRAMメモリ・モジュールによって検知されるように、SDRAMコントローラの初期化の前に、ソフトウェアは、クロック・イネーブル制御ビット信号 `CKE_Control_bit` を“1”にセットしなければならない。“ボタン・リセット（Button-Reset）”、またはメモリ内容が保存されない他の種類のリセットによって、システムがリセットされたならば、ソフ

10

20

30

40

50

トウェアは、実リアル・システムを再びリフレッシュする直前まで、クロック・イネーブル制御ビット信号 `CKE__Control__bit` を、“0”状態のままにしておく。この場合、ソフトウェアは、SDRAMがセルフリフレッシュ・モードに依然としてある間に、通常動作状態のためにSDRAMコントローラを、セットアップする。

【0024】

SDRAMコントローラの初期化の結果、実行されるSDRAMバス・サイクルは、クロック・イネーブル信号 `CKE` がインアクティブであるが故に、SDRAMによって、無視されるであろう。このことは望ましい。というのは、SDRAMメモリは、リセットの前に正確にセットアップされており、依然として正しい状態にある。SDRAMコントローラが完全にセットアップされると、ソフトウェアは、クロック・イネーブル制御ビット信号 `CKE__Control__bit` を、“1”にセットする。これは、クロック・イネーブル出力信号 `CKE__Off__Chip` を、“1”になるようにする。ソフトウェアについて、最終ステップは、実システム・リフレッシュをリイネーブルすることである。

10

【0025】

たとえ、ソフトウェアが、通常動作中に、クロック・イネーブル制御ビット信号 `CKE__Control__bit` を切替えても、フィードバック信号 `Feedback` ロジックは、クロック・イネーブル出力信号 `CKE__Off__Chip` が、ハイに留まることを許容する。この特徴は、ソフトウェアが、クロック・イネーブル出力信号 `CKE__Off__Chip` を“0”にセットすることを阻止する。これは、適切にそのようにするために、メモリ・コントローラを用いることなく、リフレッシュ・コントローラによって行われる。このステップが、メモリ・コントローラによって行われるときには、メモリ・コントローラは、コアからのクロック・イネーブル信号 `CKE__From__Core` を“0”にセットする前に、適切なセルフリフレッシュ・モード・コマンドを発行する。これらのコマンドが、コアからのクロック・イネーブル信号 `CKE__From__Core` を“0”にセットする前に、実行されないならば、SDRAMメモリは、未定義状態になり、セルフリフレッシュ・モードにならない。

20

【0026】

リフレッシュ制御回路300を、システム・オン・ア・チップ内の動作に関連して前述したが、リフレッシュ制御回路300は、例えば図1のリフレッシュ・コントローラ134のような、リフレッシュ・コントローラとして実現することもでき、および例えば図1のデータ処理システムのようなデータ処理システム内のメモリ・コントローラ・チップ外に設けられる。このような場合、リセット信号 `System__Reset__N` およびクロック信号 `Clock` は、データ処理システムから得られるであろう。クロック信号 `Clock` は、SDRAMメモリにより用いられるクロック信号と同一でなければならない。というのは、データ処理システムは、多くのクロックを有し得るからである。コアからのクロック・イネーブル信号 `CKE__From__Core` は、チップからのクロック・イネーブル信号（通常、`CKE__From__Chip` と称される）であり、システムSDRAMへのメモリ・コントローラ・チップの `CKE` 出力である。チップからのクロック・イネーブル信号は、SDRAMメモリによって受取られる前に、例えば図1のリフレッシュ・コントローラ134のような、リフレッシュ・コントローラによって、典型的にインターセプトされる。クロック・イネーブル出力信号 `CKE__Off__Chip` は、メモリへのクロック・イネーブル信号（通常、`CKE__To__Mem` と称される）によって置換えられる。メモリへのクロック・イネーブル信号は、この発明のシステムSDRAMに送られるクロック・イネーブル信号（`CKE`）である。図3の `CKE` 制御ビット302へ接続されるシステムバスは、実行される入力/出力（I/O）トランザクションを有するデータ処理システム内のバスとすることができる。システムバスは、例えば、プロセッサ・バス、PCIバス、またはISAバスとさえすることができる。ソフトウェアが、バスに対してI/Oトランザクションを実行して、クロック・イネーブル制御ビット信号 `CKE__Control__bit` を制御する限り、いかなるバスも満足できる。

30

40

【0027】

50

図4は、パワーオン・シーケンス中の、図3のこの発明のリフレッシュ制御回路300の動作を説明するタイミング図を示す。クロック・イネーブル制御ビット信号CKE__Control__bitのパワーオン状態が“0”であり、コアからのクロック・イネーブル信号CKE__From__Coreが“1”である。クロック・イネーブル制御ビット信号CKE__Control__bitが、ソフトウェアによって“1”にセットされるまで、クロック・イネーブル出力信号CKE__Off__Chipは、“0”に留まる。その後、クロック・イネーブル出力信号CKE__Off__Chipは、コアからのクロック・イネーブル信号CKE__From__Coreの変移に従う。また、次のリセットの前の、クロック・イネーブル制御ビット信号CKE__Control__bitのさらなる変化は、クロック・イネーブル出力信号CKE__Off__Chipに影響を及ぼさない。

10

【0028】

図5は、“ホット・リセット(hot-reset)”、すなわちシステムがしばらくの間、通常モードで実行された後に発生するリセットの間の、図3のこの発明のリフレッシュ制御回路300の動作を説明するタイミング図を示す。例えば、リセット・ボタンを押すオペレータにより、あるいはウォッチドッグ・タイマーの満了の結果、“ホット・リセット”が生じる。もちろん、“ホット・リセット”を発生させることのできるシステム内に、他の可能なアクションが存在する。SDRAMをセルフリフレッシュ・モードにする割込みサービス・ルーチンの結果、コアからのクロック・イネーブル信号CKE__From__Coreは、ローになる。これは、クロック・イネーブル出力信号CKE__Off__Chipを“0”にさせる。このとき、割込みサービス・ルーチンは、ソフト・リセットを行うことによって、リセット信号System__Reset__Nをアクティベートする。このことは、クロック・イネーブル制御ビット信号CKE__Control__bitを、“0”にし、最終的にコアからのクロック・イネーブル信号CKE__From__Coreを“1”にする。リセット期間中、およびリセット信号System__Reset__Nが“1”になった後は、クロック・イネーブル出力信号CKE__Off__Chipは、“0”に留まる。ソフトウェアがクロック・イネーブル制御ビット信号CKE__Control__bitをセットするとき、クロック・イネーブル出力信号CKE__Off__Chipが“1”になる。

20

【0029】

図4および図5において、イベント間のクロックの数は、イベント間のクロック・サイクルの実際の数を示すことを意味していないが、イベントの相対的発生の実例として単に示されていることに留意すべきである。実際には、リセット信号System__Reset__Nは、多数のクロック・サイクルに対して、典型的にローとなり、例えば、ハイになるリセット信号System__Reset__Nと、ハイになるクロック・イネーブル制御ビット信号CKE__Control__bitとの間に、多数のクロック・サイクルが存在し得る。

30

【0030】

図6は、システム・リセットの期間中、SDRAMメモリ・デバイスに対して、セルフリフレッシュ・モードを保持するこの発明の方法を説明するフローチャートを示す。まず最初に、データ処理システムが、システムがリセットされる指示を受取る(ステップ602)。次に、データ処理システムは、リセット指示の種類を決定する(ステップ604)。例えば、リセットは、おそらくシステムがハングしたので、ユーザがコンピュータのリセット・ボタンを押した“ボタン”リセットであり、あるいはリセットは、システムのパワーオンによるリセットとすることができる。

40

【0031】

システム・リセットの種類が一旦決定されると、データ処理システムは、外部リフレッシュがリイネーブルされるまで、リセットの期間中、セルフリフレッシュ・モードを用いて、SDRAMメモリがリフレッシュされたままにすべきか否かを決定する(ステップ606)。例えば、システムがパワーダウン状態からパワーアップ状態になるために、SDRAMメモリをリフレッシュされた状態に保つ必要がない場合には、システムは、リフレッ

50

シュ・コントローラにおけるCKE__Control__bitをすぐにイネーブルして、ソフトウェアが、通常動作にSDRAMコントローラをセットアップすることを許容する必要がある(ステップ616)。SDRAMメモリをリフレッシュ状態に保つ必要がある場合には、データ処理システムは、SDRAMコントローラに、SDRAMをセルフリフレッシュ・モードにすることを通知する(ステップ608)。これは、SDRAMコントローラに、SDRAMメモリへセルフリフレッシュ・コマンドを送らせ、続いてCKE信号をSDRAMコントローラによってローにする。SDRAMは、セルフリフレッシュ・モードのままにする必要がある。というのは、例えば、システムのハング状態のために、ユーザによってシステムがリセットされ、しかも、システムがリセットされた後に、ユーザが使用のために保有したい内容がSDRAMメモリ内に存在するかもしれないからである。

10

【0032】

リフレッシュ・コントローラは、CKE信号の制御を行って、SDRAMメモリをセルフリフレッシュ・モードに保ち、他方、リフレッシュ・コントローラは、システムがリセット(リブート)を完了するのを待つ(ステップ610)。システムは、その通常のリブート・プロシージャの一部として、SDRAMコントローラをセットアップする。システムは、そのSDRAMセットアップの一部として、外部リフレッシュをリイネーブルする(ステップ612)。システムが、外部リフレッシュをリイネーブルする直前に、リフレッシュ・コントローラに、CKEラインの制御を解放させることを命令することは差し支えない。これは、CKE__Control__bitを書込み、ハイにセットすることによって、システムが行う。システムがCKE__Control__bitをハイにセットすると、SDRAMメモリへのCKE信号の制御は、メモリ・コントローラへ引渡される。このことは、システムが、SDRAMメモリの制御をリアサートすることを許容し(ステップ614)、この時点で、プロセスは終了する。

20

【0033】

この発明を、SDRAMメモリについて主に説明したが、この発明は、リフレッシュを要求し、およびセルフリフレッシュ機能を有するいかなるメモリにも適用できることに留意すべきである。

【0034】

この発明を、十分に機能するデータ処理システムについて説明したが、当業者であれば、この発明の方法を、命令のコンピュータ読取り可能な媒体の形態で、あるいは種々の形態で頒布することができ、およびこの発明は、頒布するのに実際に用いられる特定種類の信号担持媒体にかかわらず、同様に適用できることがわかることを知るのは重要である。コンピュータ読取り可能な媒体の例は、フロッピー(登録商標)ディスク、ハードディスク・ドライブ、RAM、CD-ROM、DVD-ROMのような記録可能媒体と、デジタル/アナログ通信リンク、および例えば無線周波および光波伝送のような伝送形態を用いるワイヤまたはワイヤレス通信リンクのような伝送タイプ媒体とを含んでいる。コンピュータ読取り可能な媒体は、特定のデータ処理システムにおいて実際に使用するためにデコードされるコード化フォーマットの形態をとることができる。

30

【0035】

この発明の記述は、説明のために与えられており、この発明を開示された形態に限定することを意図するものではない。当業者には、多くの変形、変更が明らかであろう。例えば、リフレッシュ回路について特定の構造を示したが、この発明のメカニズムは、異なる回路構造にも適用できる。この発明の原理および実際の応用例を最良に説明するために、および意図する特定の使用に適した種々の変形を含む種々の実施例について、当業者にこの発明を理解させることを可能にするために、実施例を選び説明した。

40

【0036】

まとめとして、本発明の構成に関して以下の事項を開示する。

(1) データ処理システムのリセットの期間中、セルフリフレッシュ可能なメモリ・デバイスの内容を保持する方法であって、

50

データ処理システムのリセット状態の指示を受取るステップと、
必要ならば、前記メモリ・デバイスがセルフリフレッシュ・モードに保たれるように、メモリ・コントローラから前記メモリ・デバイスへの信号を変更するステップと、
前記データ処理システムが外部リフレッシュ信号をリネーブルするまで、前記メモリ・デバイスをセルフリフレッシュ・モードに保つステップと、
を含む方法。

(2) 前記メモリ・デバイスは、シンクロナス・ダイナミック・ランダム・アクセス・メモリである、上記(1)に記載の方法。

(3) 前記リセット状態が、前記メモリ・デバイスの内容を保持することを必要としないという決定に応じて、前記メモリ・コントローラから前記メモリ・デバイスへの前記信号の変更を行わないステップをさらに含む、上記(1)に記載の方法。

(4) 前記リセット状態は、パワーオン状態である、上記(3)に記載の方法。

(5) セルフリフレッシュ可能なメモリ・デバイスの内容を保存するリフレッシュ制御回路であって、

前記リフレッシュ制御回路が設けられるデータ処理システムからの信号を解釈するデコーダと、

ORゲートと、

ANDゲートと、

ラッチとを備え、

前記デコーダは、前記ORゲートの第1の入力に接続された出力と、システム・リセット信号が供給される入力とを有し、

前記ANDゲートは、前記ORゲートの出力に接続された第1の入力と、前記システム・リセット信号が供給される第2の入力と、前記データ処理システムからのクロック・イネーブル信号が供給される第3の入力とを有し、

前記ANDゲートは、前記ラッチの入力と、セルフリフレッシュ可能なメモリ・デバイスとに接続された出力を有し、

前記ラッチの出力は、前記ORゲートの第2の入力に接続されている、リフレッシュ制御回路。

(6) 前記データ処理システムは、システム・オン・ア・チップであり、前記クロック・イネーブル信号は、前記システム・オン・ア・チップのコアによって発生される、上記(5)に記載のリフレッシュ制御回路。

(7) 前記システム・オン・ア・チップは、特定用途向け集積回路である、上記(6)に記載のリフレッシュ制御回路。

(8) 前記システム・オン・ア・チップは、プログラマブル・アレイ・ロジック・チップである上記(6)に記載のリフレッシュ制御回路。

(9) 前記システム・オン・ア・チップは、フィールド・プログラマブル・アレイ・ロジック・チップである上記(6)に記載のリフレッシュ制御回路。

(10) 前記セルフリフレッシュ可能なメモリ・デバイスは、シンクロナス・ダイナミック・ランダム・アクセス・メモリである、上記(5)に記載のリフレッシュ制御回路。

(11) バスに接続されたプロセッサと、

前記バスに接続されたメモリ・デバイス・コントローラと、

前記メモリ・デバイス・コントローラに接続されたリフレッシュ・コントローラと、

前記メモリ・デバイス・コントローラに接続されたセルフリフレッシュ可能なメモリ・デバイスとを備え、

前記リフレッシュ・コントローラは、システム・リセットの期間中、前記セルフリフレッシュ可能なメモリ・デバイス内の内容を保存する、データ処理システム。

(12) 前記セルフリフレッシュ可能なメモリ・デバイスは、シンクロナス・ダイナミック・ランダム・アクセス・メモリである、上記(11)に記載のデータ処理システム。

(13) 前記リフレッシュ・コントローラは、また、前記バスに接続されている、上記(

10

20

30

40

50

１１）に記載のデータ処理システム。

（１４）前記リフレッシュ・コントローラは、ＡＮＤゲートと、ＯＲゲートと、フィードバック・ループを与えるラッチとを備え、前記ＡＮＤゲートの出力は、前記メモリ・コントローラへの信号を制御し、前記メモリ・コントローラは、システム・リセットの期間中、セルフリフレッシュ可能なメモリの内容を保持する、上記（１１）に記載のデータ処理システム。

（１５）前記データ処理システムは、システム・オン・ア・チップである、上記（１１）に記載のデータ処理システム。

（１６）前記システム・オン・ア・チップは、特定用途向け集積回路である、上記（１５）に記載のデータ処理システム。

（１７）前記システム・オン・ア・チップは、プログラマブル・アレイ・ロジック・チップである上記（１５）に記載のデータ処理システム。

（１８）前記システム・オン・ア・チップは、フィールド・プログラマブル・アレイ・ロジック・チップである上記（１５）に記載のデータ処理システム。

（１９）データ処理システムのリセットの期間中、データ処理システム内のメモリ・デバイスの内容を保存するシステムであって、

データ処理システムがいつリセット信号を受取ったかを決定する第１の手段と、

前記データ処理システムがリセット信号を受取ったことの決定に応じて、外部リフレッシュ信号が前記データ処理システムによってリネーブルされるまで、前記メモリ・デバイスの内容が保存されたままであるように、前記メモリ・デバイスへの制御信号を変更する第２の手段と、

を備えるシステム。

（２０）前記メモリは、セルフリフレッシュ可能なメモリ・デバイスである上記（１９）に記載のシステム。

（２１）前記メモリは、シンクロナス・ダイナミック・ランダム・アクセス・メモリである上記（１９）に記載のシステム。

（２２）前記第２の手段は、外部リフレッシュ信号が前記データ処理システムによってリネーブルされるまで、前記メモリ・デバイスをセルフリフレッシュ・モードに保つ、上記（１９）に記載のシステム。

【図面の簡単な説明】

【図１】この発明を実施することのできるデータ処理システムのブロック図を示す。

【図２】この発明を実施できるシステム・オン・ア・チップのブロック図を示す。

【図３】外部リフレッシュ・サイクルをリネーブルするこの発明のリフレッシュ制御回路を示す。

【図４】パワーオン・シーケンス中の、図３のこの発明のリフレッシュ制御回路３００の動作を説明するタイミング図を示す。

【図５】“ホット・リセット”中の図３のこの発明のリフレッシュ制御回路３００の動作を説明するタイミング図を示す。

【図６】システム・リセットの期間中、ＳＤＲＡＭメモリ・デバイスに対して、セルフ・リフレッシュ・モードを保持するこの発明の方法を説明するフローチャートを示す。

【符号の説明】

１０２ プロセッサ

１０４ ＳＤＲＡＭメインメモリ・コントローラ

１０６ ＰＣＩローカルバス

１０８ ＰＣＩブリッジ

１１０ ＬＡＮアダプタ

１１２ ＳＣＳＩホストバス・アダプタ

１１４ 拡張バス・インタフェース

１１６ オーディオ・アダプタ

１１８ グラフィックス・アダプタ

10

20

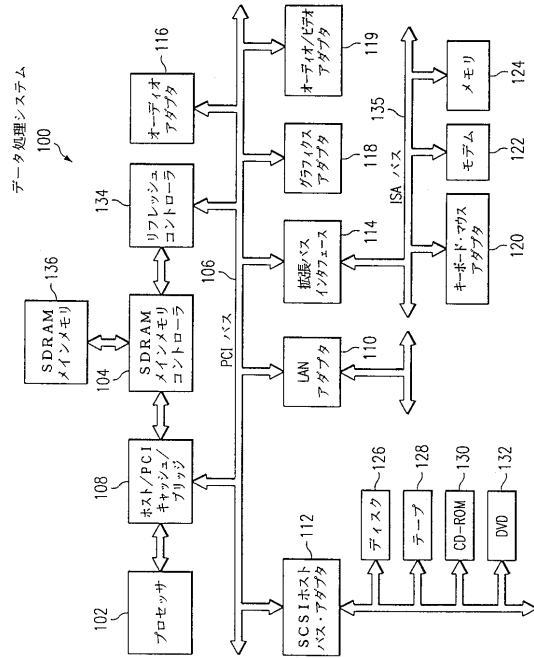
30

40

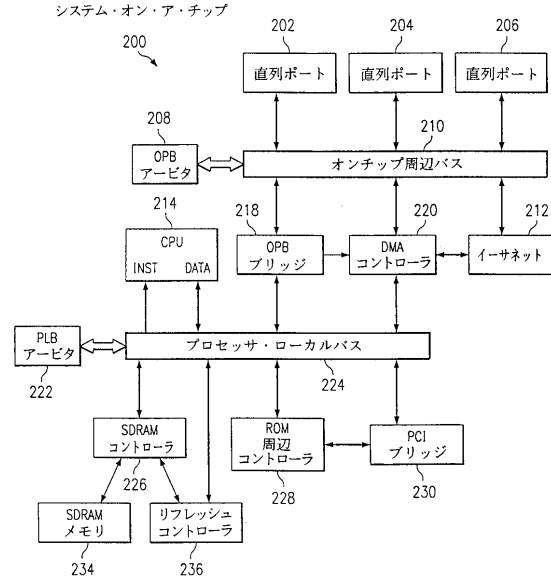
50

1 1 9	オーディオ／ビデオ・アダプタ	
1 2 0	キーボードおよびマウス・アダプタ	
1 2 2	モデム	
1 2 4	付加メモリ	
1 2 6	ディスク・ドライバ	
1 2 8	テープ・ドライバ	
1 3 0	C D - R O Mドライブ	
1 3 2	D V D - R O M	
1 3 4	リフレッシュ・コントローラ	
1 3 5	I S Aバス	10
1 3 6	S D R A Mメイン・メモリ	
2 0 0	システム・オン・ア・チップ	
2 0 2 ~ 2 0 6	直列ポート	
2 0 8	O P Bアービタ	
2 1 0	O P Bバス	
2 1 2	イーサネット(R) (登録商標)	
2 1 4	C P U	
2 2 0	D M Aコントローラ	
2 2 2	P L Bアービタ	
2 2 4	プロセッサ・ローカルバス	20
2 2 6	S D R A Mコントローラ	
2 2 8	R O M周辺コントローラ	
2 3 4	S D R A Mメモリ	
2 3 6	リフレッシュ・コントローラ	
3 0 0	リフレッシュ制御回路	
3 0 2	デコーダ	
3 0 4	O Rゲート	
3 0 6	A N Dゲート	
3 0 8	Dフリップフロップ・ラッチ	

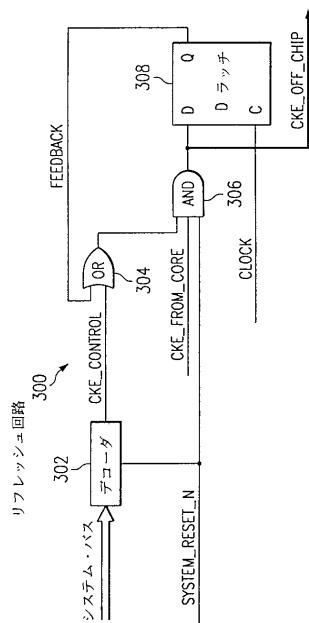
【図 1】



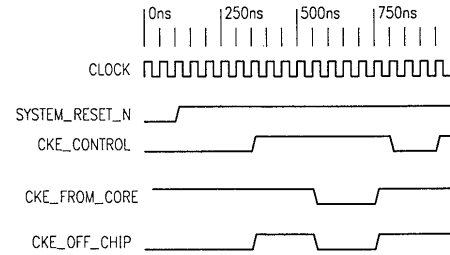
【図 2】



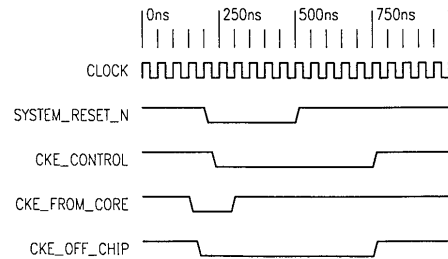
【図 3】



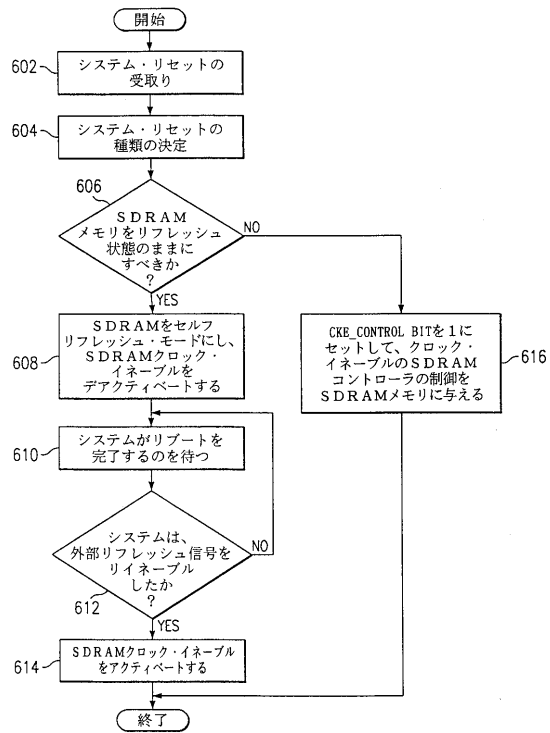
【図 4】



【図 5】



【図 6】



フロントページの続き

- (72)発明者 デヴィッド・ローレンス・アッタウェイ
アメリカ合衆国 78729 テキサス州 オースティン タイチェスター コート 13303
- (72)発明者 レオナルド・エフ・チェッティ
アメリカ合衆国 78613 テキサス州 セダー パーク グランドリッジ トレイル 2404
- (72)発明者 リチャード・ニコラス・イアチェッタ、ジュニア
アメリカ合衆国 78660 テキサス州 ピーフルーガー ヴィル ダンスワース ドライブ 17907
- (72)発明者 サクソン・ヨンゲ
アメリカ合衆国 78729 テキサス州 オースティン ヘンドリックス ドライブ 8012

審査官 石川 正二

- (56)参考文献 特開平05-028757(JP,A)
特開平07-320481(JP,A)

(58)調査した分野(Int.Cl., DB名)

G11C 11/406
G06F 12/16
G11C 11/403
G11C 11/407