

(19) 日本国特許庁(JP)

(12) 公表特許公報(A)

(11) 特許出願公表番号

特表2017-503229

(P2017-503229A)

(43) 公表日 平成29年1月26日(2017.1.26)

(51) Int.Cl.	F I	テーマコード (参考)
G06F 12/00 (2006.01)	G06F 12/00 560F	5B060
G11C 11/401 (2006.01)	G11C 11/34 371Z	5M024
G06F 7/535 (2006.01)	G06F 7/535	

審査請求 有 予備審査請求 未請求 (全 50 頁)

(21) 出願番号 特願2016-527212 (P2016-527212)
 (86) (22) 出願日 平成26年11月4日 (2014.11.4)
 (85) 翻訳文提出日 平成28年6月14日 (2016.6.14)
 (86) 国際出願番号 PCT/US2014/063831
 (87) 国際公開番号 W02015/069628
 (87) 国際公開日 平成27年5月14日 (2015.5.14)
 (31) 優先権主張番号 14/075,539
 (32) 優先日 平成25年11月8日 (2013.11.8)
 (33) 優先権主張国 米国 (US)

(71) 出願人 595168543
 マイクロン テクノロジー, インク.
 アメリカ合衆国, アイダホ州 83716
 -9632, ボイズ, サウス フェデ
 ラル ウェイ 8000
 (74) 代理人 100074099
 弁理士 大菅 義之
 (74) 代理人 100106851
 弁理士 野村 泰久
 (72) 発明者 ウィーラー, カイル ビー.
 アメリカ合衆国, アイダホ州 83646
 , メリディアン, ウェスト グラッシ
 ブランチ ドライブ 2311
 Fターム(参考) 5B060 MM20

最終頁に続く

(54) 【発明の名称】 メモリ用の除算演算

(57) 【要約】

本開示の実施例は、メモリ内で除算演算を実行するための装置及び方法を提供する。実施例の装置は、センス線及び第1の本数の選択線に接続された第1の数のメモリセルを含む第1アドレス空間を備える。第1アドレス空間は、被除数値を記憶する。第2アドレス空間は、センス線及び第2の本数の選択線に接続された第2の数のメモリセルを含む。第2アドレス空間は、除数値を記憶する。第3アドレス空間は、センス線及び第3の本数の選択線に接続された第3の数のメモリセルを含む。第3アドレス空間は、剰余値を記憶する。検出回路は、被除数値及び除数値を受け取り、被除数値を除数値で割り、剰余結果を第3の数のメモリセルに記憶させるように構成することができる。

【選択図】 図2A

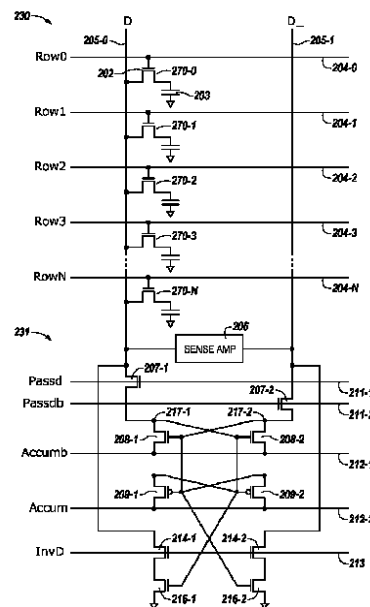


Fig. 2A

【特許請求の範囲】

【請求項 1】

除算演算の実行方法であって、

メモリアレイのセンス線に接続された複数のメモリセルの第 1 部分に記憶された被除数値の論理表現、及び前記メモリアレイの前記センス線に接続された前記複数のメモリセルの第 2 部分に記憶された除数値の論理表現を用いて除算演算を実行すること、並びに

前記除算演算の結果の商の値の論理表現を、前記メモリアレイの前記センス線に接続された前記複数のメモリセルの第 3 部分に記憶させること、を含む方法。

【請求項 2】

前記被除数値の前記論理表現を、前記センス線に接続された前記複数のメモリセルの前記第 1 部分にビットベクトルとして記憶させることをさらに含み、前記複数のメモリセルの前記第 1 部分の各メモリセルは、前記ビットベクトルの各ビットを記憶する、請求項 1 に記載の方法。

10

【請求項 3】

前記除数値の前記論理表現を、前記センス線に接続された前記複数のメモリセルの前記第 2 部分にビットベクトルとして記憶させることをさらに含み、前記複数のメモリセルの前記第 2 部分の各メモリセルは、前記ビットベクトルの各ビットを記憶する、請求項 1 に記載の方法。

【請求項 4】

前記商の値の前記論理表現を記憶させることは、前記商の値の前記論理表現を、前記センス線に接続された前記複数のメモリセルの前記第 3 部分にビットベクトルとして記憶させることを含み、前記複数のメモリセルの前記第 3 部分の各メモリセルは、前記ビットベクトルの各ビットを記憶する、請求項 1 に記載の方法。

20

【請求項 5】

前記商の値は、前記除数値で前記被除数値を完全に割った回数を表す整数である、請求項 1 に記載の方法。

【請求項 6】

前記複数のメモリセルの前記第 1 部分は N 個のメモリセルで構成されており、前記除算演算を実行することは、

前記複数のメモリセルの前記第 1 部分の N 番目のメモリセルに記憶されたビットを前記複数のメモリセルの第 4 部分の第 1 メモリセルにコピーすることであって、前記複数のメモリセルの前記第 4 部分は、前記除算演算の前記結果の剰余値の論理表現を記憶すること、及び

30

前記除数値を前記剰余値と比較することを含む、請求項 1 から請求項 5 のいずれか一項に記載の方法。

【請求項 7】

前記除数値が前記剰余値以下である場合、

前記除数値を前記剰余値から引き、得られた値の論理表現を前記剰余値の前記論理表現として前記複数のメモリセルの前記第 4 部分に記憶させること、

前記商の値の前記論理表現をシフトし、インクリメントして、前記商の値を更新すること、及び

40

前記剰余値の前記論理表現をシフトして前記剰余値を更新することをさらに含み、請求項 6 に記載の方法。

【請求項 8】

前記除数値が前記剰余値よりも大きい場合、

前記商の値の前記論理表現をシフトして前記商の値を更新すること、及び

前記剰余値の前記論理表現をシフトして前記剰余値を更新することをさらに含み、請求項 7 に記載の方法。

【請求項 9】

前記複数のメモリセルの前記第 1 部分の N - 1 番目のメモリセルに記憶されたビットを

50

前記複数のメモリセルの前記第 4 部分の前記第 1 メモリセルにコピーすること、及び前記除数値を前記剰余値と比較することをさらに含む、請求項 8 に記載の方法。

【請求項 10】

前記複数のメモリセルの前記第 1 部分の前記 N 番目のメモリセルに記憶された前記ビットを前記複数のメモリセルの前記第 4 部分の前記第 1 メモリセルにコピーする前に、前記剰余値及び前記商の値を 0 に初期化することをさらに含む、請求項 8 または請求項 9 に記載の方法。

【請求項 11】

前記比較に応じたマスクを作成することをさらに含み、前記マスクは、前記除数値が前記剰余値以下であるかどうか、または前記除数値が前記剰余値よりも大きいかどうかを示す、請求項 8 または請求項 9 に記載の方法。

10

【請求項 12】

センス線及び第 1 の本数の選択線に接続された第 1 の数のメモリセルを含むメモリアレイの第 1 アドレス空間であって、被除数値を記憶する第 1 アドレス空間と、

前記センス線及び第 2 の本数の選択線に接続された第 2 の数のメモリセルを含む前記メモリアレイの第 2 アドレス空間であって、除数値を記憶する第 2 アドレス空間と、

前記センス線及び第 3 の本数の選択線に接続された第 3 の数のメモリセルを含む前記メモリアレイの第 3 アドレス空間であって、剰余値を記憶する第 3 アドレス空間と、

検出回路であって、

前記被除数値及び前記除数値を受け取り、

前記被除数値を前記除数値で割り、

前記剰余値を前記第 3 アドレス空間に記憶させるように構成された検出回路と、

を備える装置。

20

【請求項 13】

前記センス線に接続された前記第 1 の数のメモリセルは、前記センス線に接続された N 個のメモリセルで構成されており、前記被除数値を前記除数値で割るよう構成されている前記検出回路は、

前記第 1 の数のメモリセルの N 番目のメモリセルに記憶されたビットを前記第 3 の数のメモリセルの第 1 メモリセルにコピーし、

前記除数値を前記剰余値と比較するよう構成されている前記検出回路からなる、請求項 12 に記載の装置。

30

【請求項 14】

前記剰余値を前記第 3 アドレス空間に記憶させるよう構成されている前記検出回路は、

前記除数値が前記剰余値以下である場合、

前記除数値を前記剰余値から引き、得られた値の論理表現を前記第 3 の数のメモリセルに記憶させ、

前記得られた値の前記論理表現をシフトして前記剰余値を更新するよう構成されている前記検出回路からなり、

前記被除数値を前記除数値で割るよう構成されている前記検出回路はさらに、前記メモリアレイの第 4 アドレス空間に記憶された商の値の論理表現をシフトし、インクリメントするよう構成されている前記検出回路からなる、請求項 13 に記載の装置。

40

【請求項 15】

前記被除数値を前記除数値で割るよう構成されている前記検出回路はさらに、前記除数値が前記剰余値よりも大きい場合、前記商の値の前記論理表現をシフトして前記商の値を更新するよう構成されている前記検出回路からなる、請求項 14 に記載の装置。

【請求項 16】

前記剰余値を前記第 3 アドレス空間に記憶させるよう構成されている前記検出回路は、前記除数値が前記剰余値よりも大きい場合、前記剰余値の論理表現をシフトして前記剰余値を更新するよう構成されている前記検出回路からなる、請求項 15 に記載の装置。

50

【請求項 17】

前記被除数値を前記除数値で割るよう構成されている前記検出回路はさらに、前記比較に応じたマスクを作成するよう構成されている前記検出回路からなり、前記マスクは、前記除数値が前記剰余値以下であるかどうか、または前記除数値が前記剰余値よりも大きいかどうかを示す、請求項 16 に記載の装置。

【請求項 18】

前記センス線及び第 4 の本数の選択線に接続された第 4 の数のメモリセルを含む前記メモリアレイの第 4 アドレス空間をさらに備え、前記第 4 アドレス空間及び前記検出回路は、前記第 4 の数のメモリセルに商の値を記憶させるよう構成されている、請求項 12 から請求項 17 のいずれか一項に記載の装置。

10

【請求項 19】

前記検出回路は、前記メモリアレイの前記メモリセルと間隔を置いて形成されたトランジスタを備える、請求項 12 から請求項 17 のいずれか一項に記載の装置。

【請求項 20】

前記検出回路は、センス線アドレスアクセスを介してデータを転送することなく、前記被除数値を前記除数値で割るよう構成されている、請求項 12 から請求項 17 のいずれか一項に記載の装置。

【請求項 21】

前記検出回路は、前記第 1 の数のメモリセル、前記第 2 の数のメモリセル、前記第 3 の数のメモリセル、及び前記第 4 の数のメモリセルと間隔を置いて設けられたトランジスタを含む計算素子を備える、請求項 12 から請求項 17 のいずれか一項に記載の装置。

20

【請求項 22】

複数の除算演算の実行方法であって、

メモリアレイの複数のセンス線に接続されたメモリセルの複数の第 1 部分に記憶された複数の被除数値を、前記メモリアレイの前記複数のセンス線に接続された前記メモリセルの複数の第 2 部分に記憶された複数の除数値で割ること、

前記複数の被除数値を前記複数の除数値で割った結果の複数の商の値を、前記メモリアレイの前記複数のセンス線に接続された前記メモリセルの複数の第 3 部分に記憶させること、及び

前記複数の被除数値を前記複数の除数値で割った前記結果の複数の剰余値を、前記メモリアレイの前記複数のセンス線に接続された前記メモリセルの複数の第 4 部分に記憶させることを含み、

30

前記複数の除算演算の実行に用いられる複数の計算は、前記複数の除算演算のいずれか 1 つの実行に用いられる複数の計算と同一である、方法。

【請求項 23】

前記メモリセルの前記複数の第 1 部分のそれぞれは、前記複数のセンス線の異なるセンス線に接続されており、前記複数の被除数値のそれぞれは、前記複数の第 1 部分の異なる第 1 部分に記憶される、請求項 22 に記載の方法。

【請求項 24】

前記メモリセルの前記複数の第 2 部分のそれぞれは、前記複数のセンス線の異なるセンス線に接続されており、前記複数の除数値のそれぞれは、前記複数の第 2 部分の異なる第 2 部分に記憶される、請求項 23 に記載の方法。

40

【請求項 25】

前記複数の除算演算のそれぞれは、前記複数の被除数値の各被除数値を前記複数の除数値の各除数値で割ることを含み、前記複数の除算演算の各除算演算の前記被除数値及び前記除数値は、前記複数のセンス線のうちの同一のセンス線に接続されたメモリセルに記憶される、請求項 24 に記載の方法。

【請求項 26】

複数のセンス線に接続されたメモリセルの前記複数の第 1 部分のそれぞれは、前記複数のセンス線のうちの 1 本に接続された N 個のメモリセルで構成されており、前記複数の被

50

除数値を前記複数の除数値で割ることは、

前記複数のメモリセルの前記第 1 部分の N 番目のメモリセルに記憶された各ビットを前記複数のメモリセルの前記第 4 部分の第 1 メモリセルのそれぞれにコピーすることを含み、

前記複数のメモリセルの前記第 1 部分の前記 N 番目のメモリセルに記憶された各ビットをコピーすることに伴う複数の計算は、前記複数の被除数値、前記複数の除数値、及び前記複数の剰余値の大きさによらず変わらない、請求項 2 5 に記載の方法。

【請求項 2 7】

前記複数の被除数値を前記複数の除数値で割ることは、前記複数の除数値を前記複数の剰余値と比較することを含む、請求項 2 6 に記載の方法。

10

【請求項 2 8】

前記複数の剰余値のうちの対応する剰余値以下である前記除数値のそれぞれについて、前記対応する剰余値から除数値を引き、得られた値の論理表現を、前記対応する剰余値の論理表現として前記複数のメモリセルの対応する第 4 部分に記憶させること、

対応する商の値の論理表現をシフトし、インクリメントして前記対応する商の値を更新すること、及び

前記対応する剰余値の前記論理表現をシフトして前記対応する剰余値を更新することをさらに含む、請求項 2 7 に記載の方法。

【請求項 2 9】

前記複数の剰余値のうちの対応する剰余値よりも大きい前記除数値のそれぞれについて

20

、前記対応する商の値の前記論理表現をシフトして前記対応する商の値を更新すること、及び

前記対応する剰余値の前記論理表現をシフトして前記対応する剰余値を更新することをさらに含む、請求項 2 8 に記載の方法。

【請求項 3 0】

前記複数のメモリセルの前記第 1 部分の N - 1 番目のメモリセルに記憶された各ビットを、前記複数のメモリセルの前記第 4 部分の前記第 1 メモリセルのそれぞれにコピーすること、及び

前記複数の除数値を前記複数の剰余値と比較することをさらに含む、請求項 2 9 に記載の方法。

30

【請求項 3 1】

前記複数のメモリセルの前記第 1 部分の前記 N 番目のメモリセルに記憶された各ビットを前記複数のメモリセルの前記第 4 部分の前記第 1 メモリセルにコピーする前に、前記複数の剰余値及び前記複数の商の値を 0 に初期化することをさらに含む、請求項 3 0 に記載の方法。

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

本開示は、一般に、半導体メモリ装置及び方法に関し、より具体的には、メモリ用の除算演算に関連した装置及び方法に関する。

40

【背景技術】

【0 0 0 2】

メモリデバイスは、通常、コンピュータまたは他の電子システムの内部に搭載された半導体集積回路として提供される。メモリには、揮発性メモリ及び不揮発性メモリを含む多くの異なる種類がある。揮発性メモリは、内部データ（例えば、ホストデータ、エラーデータなど）を維持するために電力が必要となり得る。揮発性メモリとしては、ランダムアクセスメモリ（random access memory：RAM）、ダイナミックランダムアクセスメモリ（dynamic random access memory：DRAM）、スタティックランダムアクセスメモリ（static random ac

50

cess memory : S R A M)、シンクロナスダイナミックランダムアクセスメモリ (s y n c h r o n o u s d y n a m i c r a n d o m a c c e s s m e m o r y : S D R A M)、及びサイリスタランダムアクセスメモリ (t h y r i s t o r r a n d o m a c c e s s m e m o r y : T R A M)などが挙げられる。不揮発性メモリは、電力が供給されていないときにも記憶データを保持することによって永続的なデータを提供することができる。不揮発性メモリとしては、N A N Dフラッシュメモリ、N O Rフラッシュメモリ、並びに、相変化ランダムアクセスメモリ (p h a s e c h a n g e r a n d o m a c c e s s m e m o r y : P C R A M)、抵抗変化型ランダムアクセスメモリ (r e s i s t i v e r a n d o m a c c e s s m e m o r y : R R A M)、及びスピントルクトランスファランダムアクセスメモリ (s p i n t o r q u e t r a n s f e r r a n d o m a c c e s s m e m o r y : S T T R A M)などの磁気抵抗変化型ランダムアクセスメモリ (m a g n e t o r e s i s t i v e r a n d o m a c c e s s m e m o r y : M R A M)といった抵抗可変メモリなどを挙げることができる。

【0003】

電子システムには、複数の処理リソース (例えば、1つ以上のプロセッサ)が含まれることが多い。これらの処理リソースは、命令を取得して実行し、実行された命令の結果を適切な領域に記憶させることができる。プロセッサは、算術論理ユニット (a r i t h m e t i c l o g i c u n i t : A L U)回路、浮動小数点ユニット (f l o a t i n g p o i n t u n i t : F P U)回路、及び/または組み合わせ論理ブロックなどの複数の機能ユニット (例えば、本明細書では機能ユニット回路 (f u n c t i o n a l u n i t c i r c u i t r y : F U C)と呼ばれる)を備えることができ、例えば、組み合わせ論理ブロックを用いて、A N D、O R、N O T、N A N D、N O R及びX O R論理演算などの論理演算をデータ (例えば、1つ以上のオペランド)に実行することによって命令を実行することができる。

【0004】

電子システム内の複数の構成要素は、実行用のF U Cへの命令の提供に関わっている場合がある。かかる命令は、例えば、コントローラ及び/またはホストプロセッサなどの処理リソースによって生成され得る。データ (例えば、命令の実行を受けるオペランド)は、F U Cによってアクセス可能なメモリアレイに記憶され得る。命令及び/またはデータは、F U Cがデータに対して命令を実行し始める前に、メモリアレイから取得されてもよく、さらには、順序付けられ、かつ/またはバッファされてもよい。さらに、各種演算が、F U Cを通じて1つまたは複数のクロックサイクル内で実行され得るときには、命令及び/またはデータの途中結果も、順序付けられ、かつ/またはバッファされ得る。

【0005】

多くの場合、処理リソース (例えば、プロセッサ及び/または関連F U C)は、メモリアレイの外部にあってよく、データには、(例えば、命令を実行するために)処理リソースとメモリアレイの間のバスを介してアクセスすることができる。データは、メモリアレイから当該メモリアレイの外部にあるレジスタにバスを介して移動させることができる。

【図面の簡単な説明】

【0006】

【図1】本開示のいくつかの実施形態に係るメモリデバイスを含む演算処理システムの形態をとった装置のブロック図である。

【図2A】本開示のいくつかの実施形態に係るメモリアレイの一部の概略図である。

【図2B】本開示のいくつかの実施形態に係る検出回路を用いた複数の論理演算の実行に関するタイミング図である。

【図2C-1】本開示のいくつかの実施形態に係る検出回路を用いた複数の論理演算の実行に関するタイミング図である。

【図2C-2】本開示のいくつかの実施形態に係る検出回路を用いた複数の論理演算の実行に関するタイミング図である。

10

20

30

40

50

【図 2 D - 1】本開示のいくつかの実施形態に係る検出回路を用いた複数の論理演算の実行に関するタイミング図である。

【図 2 D - 2】本開示のいくつかの実施形態に係る検出回路を用いた複数の論理演算の実行に関するタイミング図である。

【図 3 A】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

【図 3 B】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

【図 3 C】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

10

【図 3 D】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

【図 3 E】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

【図 3 F】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

【図 3 G】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

【図 3 H】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

20

【図 3 I】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

【図 3 J】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

【図 3 K】本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のレイのセルの状態を示す論理図である。

【発明を実施するための形態】

【0007】

本開示は、メモリ用の除算演算に関する装置及び方法を含む。(例えば、メモリアレイ内の)ビットベクトルに記憶可能な論理値の実施例は、センス線及び第1の本数の選択線に接続されたメモリセルを含む第1アドレス空間を備える。第1アドレス空間のメモリセルは、被除数値を記憶することができる。第2アドレス空間は、センス線及び第2の本数の選択線に接続されたメモリセルを含む。第2アドレス空間のメモリセルは、除数値を記憶することができる。除算演算の結果は、センス線及び第3の本数の選択線に接続されたメモリセルを含む第3アドレス空間に記憶される。第3アドレス空間のメモリセルは、商の値を記憶することができる。第4アドレス空間は、センス線及び第4の本数の選択線に接続されたメモリセルを含む。第4アドレス空間のメモリセルは、剰余値を記憶することができる。

30

【0008】

本明細書で使用する場合、被除数値は、除算演算において割られる数値である。除数値は、除算演算において被除数値を割る値である。除算演算の結果は、商の値及び剰余値とすることができる。商の値は、除数値で被除数値を割ることが可能な全回数を表すことができる。剰余値は、除数値で被除数値を割ることが可能な全回数の剰余を表すことができる。例えば、被除数値が11であり、かつ除数値が5である場合、被除数値(例えば、11)を除数値(例えば、5)で割った結果は、商の値が2となり、剰余値が1となる。

40

【0009】

本開示のいくつかの実施形態は、従来手法と比較して、複数の除算演算(例えば、除算機能)の実行に伴う計算数及び時間を削減することができる。複数の除算演算を並列に(例えば、同時に)実行することが可能であるため、その計算及び時間を削減することができる。複数の除算演算を並列に実行することにより、複数の除算演算の実行に伴う計算

50

を削減することができる。複数の除算演算を並列に実行することにより、複数の計算を実行する際の消費電力も削減することができる。例えば、いくつかの実施形態は、（例えば、メモリアレイ内の複数のメモリセルにおいて2値形式で）論理的に記憶されたデータ（例えば、被除数値及び除数値）を用いた除算機能の実行を提供することができる。実施形態は、メモリアレイ及び/または検出回路からバス（例えば、データバス、アドレスバス、制御バスなど）を介してデータを転送せずとも除算機能を実行することができる。除算機能には、複数の論理演算（例えば、AND、NOT、NOR、NAND、XORなど）の実行を含めることができる。しかしながら、実施形態は、これらの実施例に限定されない。

【0010】

従来手法では、データ（例えば、被除数値及び除数値）は、メモリアレイ及び検出回路から（例えば、入出力（input/output：I/O）線を含むバスを介して）、処理リソースが使用可能な複数のレジスタに転送され得る。処理リソースは、プロセッサ、マイクロプロセッサ及び/または計算エンジンなどであり、これらの処理リソースは、ALU回路及び/または適切な論理演算を実行するように構成された他の機能ユニット回路を含み得る。しかしながら、1つの被除数値と1つの除数値のみをALU回路が用いることにより、1つの除算機能（例えば、除算演算）の実行が可能である。メモリとレジスタの間でバスを介してデータを転送すると、消費電力及び所要時間の増大が避けられない可能性がある。処理リソースがメモリアレイと同一のチップに位置しているとしても、メモリアレイから計算回路にデータを移動させる際には、例えば、センス線からI/O線にデータを転送するためにセンス線アドレスアクセスを実行すること（例えば、列デコード信号を発行すること）、データをメモリアレイの周辺部に移動させること、及び除算機能に関連したレジスタにデータを提供することが必要となり得るため、電力消費が増大する可能性がある。

【0011】

本開示の以下の詳細な説明では、本明細書の一部をなす添付図面を参照する。これらの図面では、本開示の1つ以上の実施形態がどのように実施され得るかが一例として示されている。これらの実施形態は、本開示の実施形態を当業者が実施できる程度まで十分詳しく説明されている。さらには、他の実施形態を利用することも可能であること、並びに、プロセスの変更、電気的変更、及び/または構造上の変更を本開示の範囲から逸脱せずになし得ることを理解すべきである。本明細書で使用する場合、指示子「N」は、特に図面内の参照番号については、そのように指示された複数の特定の特性を含む可能性があることを示す。本明細書で使用する場合、「複数の」特定のものは、かかるものの1つ以上を指すことができる（例えば、複数のメモリアレイは、1つ以上のメモリアレイを指すことができる）。

【0012】

本明細書の図面は、先頭の数字（1桁または複数桁）が図面番号に対応し、残りの数字が図面内の要素または構成部分を識別するという付番規則に従っている。異なる図面間で類似の要素または構成部分は、類似の数字を使用することによって識別される場合がある。例えば、204は、図2Aの要素「04」を参照する場合があり、類似の要素は、図3Aでは304として参照される場合がある。理解されるであろうが、本明細書の様々な実施形態に示した要素を追加し、交換し、かつ/または削除することができ、それによって本開示の別の実施形態がいくつか提供される。加えて、理解されるであろうが、図面内に記載された要素の比率及び相対的な大きさは、本発明のある実施形態を例示することを意図したものであり、限定的な意味として解釈されるべきではない。

【0013】

図1は、本開示のいくつかの実施形態に係るメモリデバイス160を含む演算処理システム100の形態をとった装置のブロック図である。本明細書で使用する場合、メモリデバイス160、メモリアレイ130及び/または検出回路150は、「装置」として別途考慮される場合もある。

10

20

30

40

50

【0014】

システム100は、メモリデバイス160に接続されたホスト110を備えており、メモリデバイス160にはメモリアレイ130が搭載されている。ホスト110は、様々な異なる種類のホストの中でも、パーソナルラップトップコンピュータ、デスクトップコンピュータ、デジタルカメラ、携帯電話またはメモリカードリーダーなどのホストシステムとすることができる。ホスト110は、システムマザーボード及び/またはバックプレーンを含むことができ、さらには、複数の処理リソース（例えば、1つ以上のプロセッサ、マイクロプロセッサ、または他の何らかの種類の制御回路）を含むことができる。システム100は、独立した集積回路で構成することができ、あるいは、ホスト110とメモリデバイス160の両方を同一の集積回路とすることもできる。システム100は、例えば、サーバーシステム、または高性能演算処理（high performance computing：HPC）システム、またはこれらの一部、あるいはこれらの組み合わせとすることができる。図1に示した実施例は、フォン・ノイマン・アーキテクチャを備えたシステムを示しているが、本開示の実施形態は、非フォン・ノイマン・アーキテクチャ（例えば、チューリングマシン）で実装することもできる。このアーキテクチャは、フォン・ノイマン・アーキテクチャに付随することの多い1つ以上の構成要素（例えば、CPU、ALUなど）を含まなくてもよい。

10

【0015】

分かりやすくするため、システム100は、本開示に特定の関連性を有する特徴に焦点を当てるために簡略化されている。メモリアレイ130は、例えば、DRAMアレイ、SRAMアレイ、STT RAMアレイ、PCRAMアレイ、TRAMアレイ、RRAMアレイ、NANDフラッシュアレイ及び/またはNORフラッシュアレイとすることができる。メモリアレイ130は、選択線（本明細書では、ワード線またはアクセス線と呼ぶことがある）によって接続された各行と、センス線（本明細書では、桁線またはデータ線と呼ぶことがある）によって接続された各列とに配列されたメモリセルを含むことができる。図1には単一のアレイ130を示したが、実施形態はそのように限定されない。例えば、メモリデバイス160は、複数のアレイ130（例えば、DRAMセルからなる複数のバンク）を備えてもよい。図2Aに関連して、DRAMアレイの実施例について説明する。

20

【0016】

メモリデバイス160は、I/Oバス156（例えば、データバス）を介し、I/O回路144を通じて提供されたアドレス信号をラッチするアドレス回路142を備える。メモリアレイ130にアクセスするためには、アドレス信号を受け取り、これらを行デコーダ146及び列デコーダ152によってデコードする。いくつかの実施例では、アドレス信号は、これよりも多くのまたは少ない行デコーダによってデコードすることができる。例えば、メモリデバイスは、3つの行デコーダを備えることができる。本明細書で使用する場合、行デコーダを選択デコーダと呼ぶことがある。センス線での電圧及び/または電流の変化を検出回路150を用いて検出することにより、メモリアレイ130からデータを読み出すことができる。検出回路150は、メモリアレイ130からの1ページ分（例えば、1行分）のデータを読み出してラッチすることができる。I/O回路144は、ホスト110との双方向のデータ通信をI/Oバス156を介して行うのに使用することができる。書き込み回路148は、メモリアレイ130にデータを書き込むのに使用される。

30

40

【0017】

いくつかの実施例では、デコーディングという用語には、プリデコーディング、最終デコーディング、並びに/または、行デコーダ146及び/もしくは列デコーダ152内で実施されるその他の種類のデコーディングを含めることができる。いくつかの実施例では、プリデコーディングという用語には、アドレスが別個にアドレス指定されないようにプリデコーディングプロセスを実装する回路が含まれる。プリデコーディング及びデコーディングという用語を本明細書で使用することにより、別個にアドレス指定可能な線及び/

50

または個々にアドレス指定可能な線という各用語を区別することができる。

【0018】

いくつかの実施例では、メモリアレイ130内の複数の選択線及び/またはセンス線を、個々にアドレス指定し、かつ/またはメモリアレイ130の他の選択線及び/またはセンス線から独立してデコードすることができる。本明細書で使用する場合、個別のアドレスは、特定の選択線を活性化するのにデコーディングを必要としないアドレスとすることができる。例えば、アドレス回路142は、複数の選択線に関連したアドレスを受け取ることができるが、これらの複数の選択線は、当該複数の選択線に関連したアドレスをデコードせずとも活性化することができる。いくつかの実施例では、個々にアドレス指定された行、及び/または別個にアドレス指定された行を、完全にデコードされた行と呼ぶことができる。メモリアレイ130に関連したメモリセルは、通常であれば、数あるメモリ構成の中でも、例えば、DRAMアレイ、SRAMアレイ、STT RAMアレイ、PCRAMアレイ、TRAMアレイ、RRAMアレイ、NANDフラッシュアレイ、及び/またはNORフラッシュアレイに使用されるメモリセルで構成することができる。

10

【0019】

制御回路140は、ホスト110から制御バス154によって提供された信号をデコードする。これらの信号には、チップイネーブル信号、ライトイネーブル信号及びアドレスラッチ信号を含めることが可能であり、これらの信号を使用して、メモリアレイ130で実行される動作を制御する。このような動作としては、データの読み出し動作、データの書き込み動作、及びデータの消去動作などがある。様々な実施形態では、制御回路140は、ホスト110からの命令を実行する役割を果たす。制御回路140は、ステートマシン、シーケンサ、または他の何らかの種類のコントローラとすることができる。

20

【0020】

検出回路150の実施例については、図2A及び図3A~3Kに関連して以下でさらに説明する。例えば、いくつかの実施形態では、検出回路150は、複数のセンスアンプ及び複数の計算素子を備えることができる。これらの計算素子には、累算器を含めてもよく、これらの計算素子を使用して、(例えば、相補センス線に関連したデータに)論理演算を実行することができる。いくつかの実施形態では、検出回路(例えば、150)を使用することにより、アレイ130に記憶されたデータを入力として用いて除算演算を実行し、その除算演算の結果を、センス線アドレスアクセスを介して転送することなく(例えば、列デコード信号を発行することなく)メモリアレイ130に再度記憶させることができる。メモリアレイ130内の選択線及びセンス線に接続されたメモリセルは、除算演算及び/または当該除算演算の実行に伴う計算を実行している間に、一時記憶域(例えば、レジスタ)として機能することができる。このようにして、検出回路の外部にある処理リソースによって(例えば、デバイス160に(例えば、制御回路140もしくは他の場所に)配置された、ホスト110及び/もしくはALU回路などの他の処理回路に関連したプロセッサによって)実行されること以外に、かつ/またはそれに加えて、検出回路150を用いて除算機能を実行することができる。

30

【0021】

従来の様々な手法では、除算演算に関連したデータを、例えば、検出回路を介してメモリから読み出し、外部のALUに提供する。外部のALU回路は、オペランドを用いて除算機能を実行し、その結果を、ローカルI/O線を介してメモリアレイに再度転送することができる。これに対して、本開示のいくつかの実施形態では、検出回路(例えば、150)は、メモリアレイ130内のメモリセルに記憶されたデータに除算演算を実行し、その結果を、検出回路に接続されたローカルI/O線をイネーブルにせずにメモリアレイ130に再度記憶させるように構成されている。

40

【0022】

このようにして、いくつかの実施形態では、除算機能を実行するために、アレイ130及び検出回路150の外部にあるレジスタ及び/またはALUを必要としなくてもよい。というのも、検出回路150は、メモリアレイ130のアドレス空間を用いて、除算機能

50

の実行に伴う適切な計算を実行することができるためである。さらに、この除算機能は、外部の処理リソースを使用せずに実行することができる。

【0023】

図2Aは、本開示のいくつかの実施形態に係るメモリアレイ230の一部の概略図である。本実施例では、メモリアレイ230は、1T1C(one transistor one capacitor)メモリセル270-0、270-1、270-2、...、270-N-1、270-N(例えば、総称してメモリセル270と呼ばれる)のDRAMアレイであり、各セルは、アクセスデバイス202(例えば、トランジスタ)及び記憶素子203(例えば、コンデンサ)で構成されている。

【0024】

いくつかの実施形態では、メモリセル270は、破壊読み出し式のメモリセルである(例えば、メモリセルに記憶されたデータを読み出すと当該データが破壊されるため、読み出してから、そのセルに当初記憶されたデータのリフレッシュが行われる)。メモリセル270は、選択線204-0(行0)、204-1(行1)、204-2(行2)、...、204-N-1(行N-1)、204-N(行N)によって接続された各行と、センス線(例えば、桁線)205-0(D)及び205-1(D_)によって接続された各列とに配列されている。いくつかの実施形態では、メモリアレイ230は、別々の回路に接続されているアドレス空間を備えることができる。

【0025】

本実施例では、セルの各列は、相補センス線205-0(D)及び205-1(D_)のペアに関連付けられている。図2Aにはメモリセル270の1つの列のみが示されているが、実施形態はそのように限定されない。例えば、ある特定のアレイは、複数列のセル及び/または複数のセンス線(例えば、4, 096, 8, 192, 16, 384など)を備えてもよい。図2Aでは、メモリセル270は、センス線205-0に接続されている。個々のセルトランジスタ202のゲートは、当該セルトランジスタに対応する選択線204-0~204-N(例えば、総称して選択線204と呼ばれる)にそれぞれ接続されており、個々のセルトランジスタにおいて、第1のソース/ドレイン領域は当該セルトランジスタに対応する選択線205-0に、第2のソース/ドレイン領域は当該セルトランジスタに対応するコンデンサ203に、それぞれ接続されている。図2Aには図示されていないが、センス線205-1も、これに接続されたメモリセルを有し得る。

【0026】

いくつかの実施例では、センス線205に接続されている各メモリセル270は、ビットを記憶することができる。これらのビットは、1つの値及び/または複数の値(例えば、被除数値、除数値、商の値及び/または剰余値)の論理表現を表すことができる。例えば、被除数値は、4つのビットベクトルによって表すことができ、これらのビットベクトルを、メモリセル270-0、メモリセル270-1、メモリセル270-2及びメモリセル270-3に記憶させることができる。いくつかの実施例では、ビットベクトルは、図2A及び図3A~3Kに用いられているものよりも多くのまたは少ないビットによって表すことができる。例えば、被除数値は、数あるビットベクトルの大きさの中でも、8ビットベクトル、16ビットベクトル及び/または32ビットベクトルによって表すことができる。いくつかの実施例では、値に対する各ビットベクトルの表現を、選択線204に沿って左右にではなく、センス線205に沿って上下に記憶させることができる。

【0027】

値に対する各ビットベクトルの論理表現には、インデックスを付けることができる。例えば、メモリセル270-0に記憶されたビットを第1インデックスに関連付けることができ、メモリセル270-1に記憶されたビットを第2インデックスに関連付けることができ、メモリセル270-2に記憶されたビットを第3インデックスに関連付けることができ、メモリセル270-3に記憶されたビットを第4インデックスに関連付けることができる。一実施例として、第1インデックスは、記憶したビットベクトルの最上位ビット(most significant bit: MSB)を示すことができ、第4インデ

10

20

30

40

50

ックスは、記憶したビットベクトルの最下位ビット (least significant bit: LSB) を示すことができる。図 2 A に関して用いられたインデックス付けは例示であって、限定的ではない。いくつかの実施形態では、他のインデックス付け方式及び/または順序を用いることができる。例えば、メモリ 270 - 3 に記憶されたビットを第 1 インデックスに関連付けることができ、メモリセル 270 - 2 に記憶されたビットを第 2 インデックスに関連付けることができ、メモリセル 270 - 1 に記憶されたビットを第 3 インデックスに関連付けることができ、メモリセル 270 - 0 に記憶されたビットを第 4 インデックスに関連付けることができる。さらに、本明細書に記載された演算は、複数の異なるレイアウト及び/またはメモリセル配置を用いて実行することができる。

【0028】

メモリアレイ 230 の各センス線に複数の値を記憶させることができる。例えば、メモリセル 270 - 0、270 - 1、270 - 2、270 - 3 は、第 1 の被除数値を表す複数のビットを記憶することができ、メモリセル 270 - 4、270 - 5、270 - 6、270 - 7 は、第 1 の除数値を表す複数のビットを記憶することができ、メモリセル 270 - 8、270 - 9、270 - 10、270 - 11 は第 1 の商の値を表す複数のビットを記憶することができ、メモリセル 270 - 12、270 - 13、270 - 14、270 - 15 は第 1 の剰余値を表す複数のビットを記憶することができる (例えば、メモリセル 270 - 3 ~ 270 - 15 は、図 2 A には示されていない)。メモリセル 270 - 0 ~ 270 - 15 は、第 1 のセンス線に接続することができる。異なるセンス線に接続されたメモリセルは、第 2 の被除数値、第 2 の除数値、第 2 の商の値及び/または第 2 の剰余値を表す複数のビットを記憶することができる。

【0029】

いくつかの実施例では、選択線 204 - 0 及び複数のセンス線 (例えば、センス線 205 - 0 及びセンス線 205 - 1) に接続されたメモリセルを同時に活性化することができる。さらに、メモリセル 270 - 0、メモリセル 270 - 1、メモリセル 270 - 2、270 - 3 も同時に活性化することができる。いくつかの実施例では、独立にアドレス指定された選択線 204 及び/またはセンス線 205 を同時に活性化して、複数のメモリセルを同時に活性化することができる。

【0030】

検出回路は、センスアンプ 206 及び計算素子 231 を備える。この検出回路は、図 1 に示した検出回路 150 とすることができる。センスアンプ 206 は、メモリセルの特定の列に対応する相補センス線 D、D_{bar} に接続されている。センスアンプ 206 は、選択されたセル (例えば、メモリセル 270) に記憶された状態 (例えば、論理データ値) を判別するように動作することができる。実施形態は、実施例のセンスアンプ 206 に限定されない。例えば、本明細書に記載されたいくつかの実施形態に係る検出回路は、電流モードセンスアンプ及び/またはシングルエンド型センスアンプ (例えば、1 本のセンス線に接続されたセンスアンプ) を含むことができる。

【0031】

いくつかの実施形態では、計算素子 (例えば、231) は、センスアンプ (例えば、206) 及び/またはメモリアレイ (例えば、230) のメモリセル 270 を構成するトランジスタと間隔を置いて形成された複数のトランジスタを含むことができる。なお、これらのメモリセルは、特定の加工寸法 (例えば、 $4F^2$ 、 $6F^2$ など) に準拠していてもよい。以下でさらに説明するように、計算素子 231 は、センスアンプ 206 と連携して、メモリアレイ 230 内のメモリセル 270 からのデータを入力として用いて除算演算を実行するように動作することができる。その結果を、センス線アドレスアクセスを介してデータを転送することなく (例えば、列デコード信号を発行し、それによってメモリアレイ及び検出回路からローカル I/O 線を介して外部回路にデータを転送することなく)、メモリアレイ 230 内のメモリセル 270 に再度記憶させることができる。このようにして、本開示のいくつかの実施形態では、除算演算及びこれに関連した計算を、従来の様々な手法よりも少ない電力を用いて実行することが可能となる。さらに、いくつかの実施形態で

10

20

30

40

50

は、計算機能を実行するためにローカルI/O線を介してデータを転送する必要がないため、従来の手法に比べて、計算素子（例えば、231）及びメモリセル270を用いた並列処理能力を高めることが可能となる。

【0032】

図2Aに示した実施例では、計算素子231に対応する回路は、センス線D及びD_{__}のそれぞれに5つのトランジスタが接続された構成をとっている。ただし、実施形態は、この実施例に限定されない。トランジスタ207-1及び207-2は、センス線D及びD_{__}にそれぞれ接続された第1のソース/ドレイン領域と、クロスカップル型ラッチに接続された（例えば、クロスカップル型NMOSトランジスタ208-1及び208-2並びにクロスカップル型PMOSトランジスタ209-1及び209-2などの、クロスカップル型トランジスタのペアの各ゲートに接続された）第2のソース/ドレイン領域とを有する。本明細書でさらに説明するように、トランジスタ208-1、208-2、209-1及び209-2を含むクロスカップル型ラッチは、2次ラッチと呼ぶことができる（センスアンプ206に対応するクロスカップル型ラッチは、本明細書では1次ラッチと呼ぶことができる）。

10

【0033】

トランジスタ207-1及び207-2は、パストラジスタと呼ぶことができる。これらのトランジスタは、各信号211-1（Passd）及び211-2（Passdb）を介してイネーブルにすることができ、それによって各センス線D及びD_{__}の電圧または電流が、トランジスタ208-1、208-2、209-1及び209-2を含むクロスカップル型ラッチの入力（例えば、2次ラッチの入力）に送られるようになっている。本実施例では、トランジスタ207-1の第2のソース/ドレイン領域は、トランジスタ208-1及び209-1の第1のソース/ドレイン領域、並びにトランジスタ208-2及び209-2の各ゲートに接続されている。同様に、トランジスタ207-2の第2のソース/ドレイン領域は、トランジスタ208-2及び209-2の第1のソース/ドレイン領域、並びにトランジスタ208-1及び209-1の各ゲートに接続されている。

20

【0034】

トランジスタ208-1及び208-2の第2のソース/ドレイン領域は、負の制御信号212-1（Accumb）に共通に接続されている。トランジスタ209-1及び209-2の第2のソース/ドレイン領域は、正の制御信号212-2（Accum）に共通に接続されている。Accum信号212-2は、電源電圧（例えば、Vcc）とすることができ、Accumb信号は、基準電圧（例えば、グラウンド）とすることができ、信号212-1及び212-2をイネーブルにすると、2次ラッチに対応する、トランジスタ208-1、208-2、209-1及び209-2を含むクロスカップル型ラッチが活性化される。活性化されたセンスアンプのペアは、共通ノード217-1と共通ノード217-2の間の差動電圧を増幅するように動作し、それにより、ノード217-1は、Accum信号電圧とAccumb信号電圧の一方の電圧に（例えば、Vccとグラウンドの一方に）駆動され、ノード217-2は、Accum信号電圧とAccumb信号電圧の他方の電圧に駆動される。以下でさらに説明するように、2次ラッチは、論理演算（例えば、除算演算）の実行に使用されている間は累算器として機能することができるため、信号212-1は「Accumb」に、信号212-2は「Accum」に、それぞれラベル付けされている。いくつかの実施形態では、累算器は、2次ラッチを形成するクロスカップル型トランジスタ208-1、208-2、209-1及び209-2、並びにパストラジスタ207-1及び208-2を含む。本明細書でさらに説明するように、いくつかの実施形態では、センスアンプに接続された累算器を含む計算素子は、論理演算を実行するように構成することができる。この論理演算には、相補センス線のペアの少なくとも一方における信号（例えば、電圧または電流）によって表されたデータ値の累算演算を実行することが含まれる。

30

40

【0035】

50

計算素子 231 は、各桁線 D 及び D_{__} に接続された第 1 のソース/ドレイン領域を有する反転トランジスタ 214 - 1 及び 214 - 2 も含む。反転トランジスタ 214 - 1 及び 214 - 2 の第 2 のソース/ドレイン領域は、トランジスタ 216 - 1 及び 216 - 2 の第 1 のソース/ドレイン領域にそれぞれ接続されている。トランジスタ 214 - 1 及び 214 - 2 の各ゲートは、信号 213 (InvD) に接続されている。トランジスタ 216 - 1 のゲートは、共通ノード 217 - 1 に接続されており、このノードに対しては、トランジスタ 208 - 2 のゲート、トランジスタ 209 - 2 のゲート、及びトランジスタ 208 - 1 の第 1 のソース/ドレイン領域も接続されている。相補的に、トランジスタ 216 - 2 のゲートは、共通ノード 217 - 2 に接続されており、このノードに対しては、トランジスタ 208 - 1 のゲート、トランジスタ 209 - 1 のゲート、及びトランジスタ 208 - 2 の第 1 のソース/ドレイン領域も接続されている。このようにして、信号 InvD をイネーブルにすると、2 次ラッチに記憶されたデータ値が反転されるようになり、この反転値がセンス線 205 - 0 及び 205 - 1 へと駆動される。

【0036】

図 2A では、計算素子 231 は、除算演算を実行するように構成されている。以下の実施例では、アレイ 230 に記憶されたデータを入力として用いて除算演算をどのように実行するか、及びその除算演算の結果を、検出回路 (例えば、センスアンプ 206 及び計算素子 231) の動作を通じて当該アレイにどのように記憶させるかについて説明する。この実施例は、選択線 204 - 0 ~ 204 - N に接続され、かつセンス線 205 - 0 に共通に接続されたメモリセル 270 に記憶されたビット (例えば、論理 1 または論理 0) を除算演算に対する各入力として用いることを含む。この除算演算の結果を、センス線 205 - 0 に接続されたメモリセルに記憶させることができる。

【0037】

一実施例として、計算素子 231 は、センス線 205 - 0 に接続されている複数のメモリセル 270 の第 1 部分及び第 2 部分に記憶された被除数値及び除数値を用いることができる。第 1 部分は、第 1 の数のメモリセルを含むことができる。第 2 部分は、第 2 の数のメモリセルを含むことができる。除算演算の途中結果は、センス線 205 - 0 に接続されているメモリセルの第 3 部分及び/または第 4 部分に保存することができる。

【0038】

除算演算は、被除数値を記憶するメモリセル 270 - 0 ~ 270 - 3 に記憶されたビットを、剰余値を記憶するメモリセル 270 - 12 ~ 270 - 15 のメモリセル 270 - 12 にコピーすることを含むことができる。メモリセル 270 - 0 ~ 270 - 3 に記憶されたビットは、連続的に (例えば、1 つずつ) メモリセル 270 - 12 にコピーすることができる。メモリセル 270 - 0 ~ 270 - 3 に記憶されたビットは、メモリセル 270 - 0 ~ 270 - 3 にこれらを記憶させたときの順序に対して逆の順序でコピーすることができる。例えば、メモリセル 270 - 3 に記憶されたビットをメモリセル 270 - 12 にコピーしてから、メモリセル 270 - 2 に記憶されたビットをメモリセル 270 - 12 にコピーすることができる。

【0039】

いくつかの実施例では、メモリセル 270 - 0 ~ 270 - 3 に記憶されたビットを、剰余ビットベクトルの第 1 インデックスに関連付けられたメモリセルにコピーすることができる。メモリセル 270 - 12 は剰余ビットベクトルの第 1 インデックスに関連付けられているが、これは、メモリセル 270 - 12 が剰余ビットベクトルの第 1 ビットを記憶することができるためである。例えば、5 に等しい剰余値は、ビットベクトル [0101] によって表すことができる。このビットベクトルは、メモリセル 270 - 12 ~ 270 - 15 に記憶させることができる。ビットベクトル [0101] の第 1 ビットは、1 に等しくすることができる。この第 1 ビットを、第 1 インデックスに関連付けられたメモリセル 270 - 12 に記憶させることができる。ビットベクトル [0101] の第 2 ビットは、0 に等しくすることができる。この第 2 ビットを、第 2 インデックスに関連付けられたメモリセル 270 - 13 に記憶させることができる。第 3 インデックスに関連付けられたメモリ

10

20

30

40

50

セル 270 - 14 は、1 に等しくすることができる。第 4 インデックスに関連付けられたメモリセル 270 - 15 は、0 に等しくすることができる。

【0040】

この除算演算を分割して、計算を反復させることができる。本明細書で使用する場合、計算は、除算演算の実行に伴う計算演算を表すことができる。例えば、数ある計算の中でも、コピー演算を計算とすることができ、かつ/または、加算演算を計算とすることができる。被除数値のビットの 1 つを剰余ビットベクトルの第 1 インデックスに関連付けられたメモリセルにコピーする度に、除算演算の反復を開始することができる。

【0041】

剰余ビットベクトルの第 1 インデックスに関連付けられたメモリセルにビットをコピーする度に、マスクを作成することができる。マスクは、除数値が、対応する剰余値以下であるかどうかを示すことができる。除数値が剰余値以下である場合、マスクの対応ビットを 1 に等しい値に設定することができ、除数値を剰余値から引くことができ、この減算の結果の論理表現（例えば、ビットベクトル）を記憶し、シフトすることができ、さらに、剰余値のビットベクトルの第 1 インデックスに関連付けられたメモリセルにコピーされたビットが、反復すべき被除数値の最終ビットではない場合には、剰余値を更新するために、商の値の論理表現（例えば、ビットベクトル）を 1 だけインクリメントすることができる。除数値が剰余値よりも大きい場合、マスクの対応ビットを 0 に設定する。その後、商の値の論理表現及び剰余値の論理表現をそれぞれシフトして、商の値及び剰余値を更新することができる。シフト操作については、図 3 A ~ 3 K に関してさらに説明する。この除算演算は、図 3 A ~ 3 K の実施例として与えられる。

10

20

【0042】

本開示の実施形態は、図 2 A に示した特定の検出回路構成に限定されない。例えば、異なる計算素子回路を使用して、本明細書に記載されたいくつかの実施形態に係る論理演算を実行することができる。図 2 A には示されていないが、いくつかの実施形態では、アレイ 230、センスアンプ 206 及び/または計算素子 231 に制御回路を接続することができる。このような制御回路は、例えば、当該アレイ及び検出回路と同一のチップに、かつ/または外部プロセッサなどの外部処理リソースに実装されてもよく、本明細書に記載された論理演算を実行するために、上記アレイ及び検出回路に対応する様々な信号のイネーブル/ディセーブルを制御することができる。さらに、図 2 A には相補センス線の 1 つのペア（205 - 0 / 205 - 1）を示しているが、アレイ 230 は、複数の相補センス線のペアを含むことができる。

30

【0043】

疑似コードの実施例を以下に挙げる。この疑似コードは、メモリにおいて除算演算を実行するための概要を提示するものである。

```

for bitN in dividend{
    copy bitN into bit0 of remainder
    compare divisor with remainder
    if (divisor <= remainder){
        subtract divisor from remainder, store in remainder
        leftshift quotient and add 1
        if (bitN is not the last bit){
            leftshift remainder
        }
    } else {
        leftshift quotient
        leftshift remainder
    }
}

```

10

20

30

40

【 0 0 4 4 】

いくつかの実施形態では、除算演算を実行するために利用可能な数ある演算の中でも、減算演算、加算演算、比較演算、シフト演算及び/またはコピー演算を用いて除算演算を実行することができる。減算演算、加算演算、比較演算、シフト演算、及び/またはコピー演算は、ブール演算及び非ブール演算を用いて実行することができる。いくつかの実施形態では、ブール演算及び非ブール演算は、論理演算（例えば、NAND、AND、NOR、ORなど）を用いて実行することができる。計算素子231と連携してNAND、AND、NOR及びOR論理演算を実行する実施例を、図2B、2C-1、2C-1、2D-1及び2D-2に挙げる。

【 0 0 4 5 】

表1に、加算演算を実行するために実行可能な複数の命令の実施例を挙げる。表1に挙げた実施例は、メモリアレイ230の一部に記憶された値（例えば、加数）の論理表現（例えば、ビットベクトル）に、1（one）を加えるものである。ビットベクトル「srcdest[p]」は加数の論理表現であり、このビットベクトルを、メモリアレイ230内の第1の数のメモリセルに記憶させることができる。ビットベクトルsrcdest[p]は、1が加えられる数値の論理表現である。Pは、ビットベクトルsrcdest[p]に関連付けられているインデックスを表す。例えば、ビットベクトル[011]は値3の論理表現であり、このビットベクトルを、メモリセル270-0~270-2に記憶させることができ、それにより、ビットsrcdest[0]は1に等しく、ビットsrcdest[1]は1に等しく、ビットsrcdest[2]は0に等しくなる。ビットベクトルtmp[p]は記憶変数の論理表現であり、このビットベクトルを、メモリアレイ230内の第2の数のメモリセルに記憶させることができる。例えば、ビットベクトルtmp[p]をメモリセル270-3~270-5に記憶させることができる。

【表 1】

1の加算 :	
read srcdest[0]	
write tmp[0]	
write-inverted srcdest[0]	
残りの全Pビット用 :	
read srcdest[p]	
or tmp[0]	
write tmp[1]	
read srcdest[p]	
nand tmp[0]	
write-inverted tmp[0]	
and tmp[1]	
write srcdest[p]	

10

20

【0046】

加算演算は、2つのステージに分けることができる。「1の加算」ステージは、第1のビットベクトル `srcdest[p]` の論理表現に1を加えることができる。「1の加算」ステージにおいて、「`read srcdest[0]`」コールは、ビットベクトル `srcdest[p]` の第1インデックスを用いてビットを読み出す（例えば、ラッチする）ことができる。例えば、本明細書で説明するように、メモリセル270-0に記憶されたビットは、センスアンプ206及び/または計算素子231においてラッチすることができる。「`write tmp[0]`」コールは、メモリセル270-3（例えば、ビットベクトル `tmp[0]` の第1インデックス）内のセンスアンプ206及び/または計算素子231においてラッチされている値を、メモリセル270-3を活性化することによって記憶させることができる。「`write-inverted srcdest[0]`」コールは、センスアンプ206及び/または計算素子231に記憶されているビットを反転することができ、その反転ビットを、メモリセル270-0を活性化することによってメモリセル270-0に記憶させることができる。

30

【0047】

「残りの全Pビット用」ステージは、ビットベクトル `srcdest[p]` の第1インデックスの後の全てのインデックスに対して表1の各コールを実行することにより、第1ステージの剰余をビットベクトル `srcdest[p]` の他のビットに移すことができる。例えば、第1反復において、「`read srcdest[p]`」コールは、メモリセル270-1に記憶されたビットをセンスアンプ206及び/または計算素子231にてラッチすることにより、メモリセル270-1に記憶されたビットを読み出すことができる。「`or tmp[0]`」コールは、ラッチされたビット（例えば、メモリセル270-1から読み出されたビット）とメモリセル270-3に記憶されたビットとを用いたOR演算を実行することができる。メモリセル270-3に記憶されたビットは、「`read srcdest[p]`」コールの後にラッチすることができる。このOR演算は、後

40

50

述するように計算素子 231 において実行することができる。「write tmp[1]」コールは、メモリセル 270-4 を活性化することによって OR 演算の結果をメモリセル 270-4 に記憶させることができる。「read srcdest[p]」コールは、メモリアレイ 270-1 に記憶されたビットをセンスアンプ 206 にてラッチすることができる。「nand tmp[0]」コールは、ラッチされたビット（例えば、センスアンプ 206 及び / または計算素子 231 に記憶されたビット）とメモリセル 270-3 に記憶されたビットとをとして用いた NAND 演算を実行することができる。メモリセル 270-3 に記憶されたビットは、センスアンプ 260 にてラッチすることができる。「read srcdest[p]」コールの後の NAND 演算のとして計算素子 231 に与えることができる。「write-inverted tmp[0]」コールは、NAND 演算の結果を計算素子 231 にて反転することができる。メモリセル 270-3 を活性化することにより、この反転ビットをメモリセル 270-3 に記憶させることができる。「and tmp[1]」コールは、NAND 演算の結果とメモリセル 270-4 に記憶されたビットとをとして用いた AND 演算を実行することができる。「write srcdest[p]」コールは、AND 演算の結果をメモリセル 270-1 に記憶させることができる。この時点で、第 2 ステージの第 2 反復を、p の第 3 インデックスを用いて実行することができる。この加算演算は、当該加算演算がビットベクトル srcdest[p] の最終インデックスに達したときに終了させることができる。

10

【0048】

表 2 に、減算演算を実行するために実行可能な複数の命令の実施例を挙げる。表 2 に示した減算演算は、表 1 に用いた構文を利用して行うことができる。この減算演算は、3 つのステージを含むことができる。すなわち、「セットアップ」ステージ、「残りの全 P ビットのループ (Loop-through remaining P bits)」ステージ、及び「クリーンアップ」ステージである。表 2 に挙げた各命令は、ビットベクトル minuend[p]、ビットベクトル subtrahend[p]、ビットベクトル tmp[p] 及びビットベクトル result[p] を用いる。ビットベクトル subtrahend[p] は、減数値の論理表現であり、他方、ビットベクトル minuend[p] は、被減数値の論理表現である。減算演算では、被減数値から減数値が引かれる。ビットベクトル result[p] は、当該ビットベクトル result[p] に記憶させることができる減算演算の結果値の論理表現である。ビットベクトル tmp[p] は、減算演算の実行中に一時記憶域として用いられる値の論理表現である。これらのビットベクトル minuend[p]、subtrahend[p]、tmp[p] 及び result[p] を、メモリアレイ 230 内の複数のメモリセル 270 に記憶させることができる。例えば、ビットベクトル minuend[p] をメモリセル 270-0 ~ 270-3 に記憶させることができ、ビットベクトル subtrahend[p] をメモリセル 270-4 ~ 270-7 に記憶させることができ、ビットベクトル tmp[p] をメモリセル 270-8 ~ 270-11 に記憶させることができ、ビットベクトル result[p] をメモリセル 270-12 ~ 270-15 に記憶させることができる。この減算演算は、減数値の 2 の補数を計算すること、及びこの減数値の 2 の補数を被減数値に加えることを含むことができる。減数値の 2 の補数をビットごとに計算して、これを被減数値の対応ビットに加えるようにすることができる。

20

30

40

【表 2】

セットアップ :	
read minuend[0]	
nand subtrahend[0]	
write tmp[0]	
read minuend[0]	10
or subtrahend[0]	
and tmp[0]	
write result[0]	
and minuend[0]	
write-inverted tmp[0]	
残りの全Pビットのループ :	
read subtrahend[p]	20
write-inverted subtrahend[p]	
xor minuend[p]	
xor tmp[0]	
write result[p]	
read minuend[p]	30
xor subtrahend[p]	
and tmp[0]	
write tmp[1]	
read minuend[p]	
and subtrahend[p]	40
or tmp[1]	
write tmp[0]	
read subtrahend[p]	
write-inverted subtrahend[p]	

クリーンアップ :
read subtrahend[p]
write-inverted subtrahend[p]
xor minuend [p]
xor tmp[0]
write result [p]
read subtrahend [p]
write-inverted subtrahend[p]

10

【 0 0 4 9 】

表 3 に、コピー演算及び/またはシフト演算を実行するために実行可能な複数の命令の実施例を挙げる。コピー演算は、入力値及び出力値として、入力及び出力をそれぞれ用いることができる。ビットベクトル `input [p]` は、入力値の論理表現である。ビットベクトル `output [p]` は、出力値の論理表現である。ビットベクトル `input [p]` のインデックスごとに、センスアンプ 206 及び/または計算素子 231 にて対応ビットをラッチすることにより、ビットベクトル `input [p]` に対応するメモリセルから 1 ビットを読み出す。ラッチした対応ビットを、ビットベクトル `input [p]` と同一のインデックスを用いて、ビットベクトル `output [p]` に対応するメモリセルに記憶させることができる。左シフト演算は、ビットベクトル `output [p]` のインデックスにシフト値を加えてから、ビットベクトル `output [p]` に対応するメモリセルにビットを記憶させることを含むことができる。例えば、「`read input [p]`」コールは、ビットベクトル `input [p]` に対応するメモリアレイ 230 の第 1 部分のうちの第 1 インデックスを有するメモリセルに記憶されているビットをラッチすることができる。また、「`write output [p]`」コールは、当該ビットを、ビットベクトル `output [p]` に対応するメモリアレイ 230 の第 2 部分のうちの第 2 インデックスを有するメモリセルに記憶させることができる。

20

30

【表 3】

各Pビットについて
read input[p]
write output[p]

40

【 0 0 5 0 】

表 4 に、比較演算を実行するために実行可能な複数の命令の実施例を挙げる。比較演算は、除数値を剰余値と比較して、除数値が剰余値よりも大きいかどうかを判定することができる。ビットベクトル `divisor [p]` は、除数値の論理表現である。ビットベクトル `remainder [p]` は、剰余値の論理表現である。「`read divisor [max]`」コールに用いられている `Max` は、ビットベクトル `divisor [p]` 及び/またはビットベクトル `remainder [p]` に関連付けられている最上位インデックスを表す。ビット `difference__found__yet [0]` 及びビット `bit difference__this__time [0]` は、一時記憶域として用いられる変数の論理表現である。ビット `result [0]` は、結果変数の論理表現である。こ

50

の比較演算は、「セットアップ」ステージ、「残りの全Pビットのループ」ステージ、及び「クリーンアップ」ステージを含む。「セットアップ」ステージは、ビットベクトル `divisor[p]` 及びビットベクトル `remainder[p]` の最大インデックスを使用する。「残りの全Pビットのループ」ステージは、最大インデックスより1つ小さいインデックスから始まり、ビットベクトル `divisor[p]` 及びビットベクトル `remainder[p]` の第1インデックスに到達するまでインデックスをデクリメントすることによってループする。例えば、最大インデックスが4である場合、「残りの全Pビットのループ」ステージは、3に等しいインデックスから始まり、1に等しい第1インデックスに達するまでインデックスをデクリメントする。「クリーンアップ」ステージは、ビット `result[0]` を反転し、当該ビットをビット `result[0]` に関連したメモリセルに記憶させる。

10

【表4】

セットアップ:	
read divisor[max]	
xor remainder[max]	
write difference_found_yet[0]	
and divisor [max]	
write result[0]	
残りの全Pビットのループ :	
read divisor [p]	
xor remainder [p]	
write difference_this_time[0]	
xor difference_found_yet[0]	
and difference_this_time[0]	
write difference_this_time[0]	
or difference_found_yet[0]	
write difference_found_yet[0]	
read difference_this_time[0]	
and divisor[p]	
or result[0]	
write result[0]	
クリーンアップ	
write-inverted result[0]	

20

30

【0051】

図2Bは、本開示のいくつかの実施形態に係る検出回路を用いた複数の論理演算の実行に関するタイミング図285-1である。タイミング図285-1は、論理演算（例えば、R入力論理演算）の第1演算フェーズの実行に関する信号（例えば、電圧信号）を示す図である。図2Bに記載した第1演算フェーズは、例えば、AND、NAND、ORまたはNOR演算の第1演算フェーズとすることができる。さらに後述するように、図2Bに示した演算フェーズの実行により、従来の処理手法よりもエネルギー消費を大幅に少なく（例えば、約半分に）することができる。というのも、従来の処理手法では、計算機能を実行するために、電圧レールの間で（例えば、電源とグラウンドの間で）フルスイングさせることが必要となり得るためである。

40

【0052】

図2Bに示した実施例では、相補的な論理値（例えば、「1」及び「0」）に対応する

50

電圧レールは、電源電圧 274 (VDD) 及びグラウンド電圧 272 (Gnd) である。論理演算を実行する前に、平衡状態を生じさせることができる。例えば、相補センス線 D 及び D_{__} を共に短絡して、平衡電圧 225 (VDD/2) になるようにする。平衡状態については、図 3 に関連して以下でさらに説明する。

【0053】

時刻 t_1 では、平衡信号 226 が非活性化され、選択された行 (例えば、データ値が検出され、当該データ値が第 1 の入力として用いられるメモリセルに対応する行) が活性化される。信号 204 - 0 は、選択された行 (例えば、行 204 - 0) に印加された電圧信号を表す。行信号 204 - 0 が、選択されたセルに対応するアクセストランジスタ (例えば、202) の閾電圧 (V_t) に到達すると、当該アクセストランジスタがターンオンし、このトランジスタにより、選択されたメモリセルに (例えば、当該セルが 1T1C の DRAMセルである場合にはコンデンサ 203 に) センス線 D が接続される。これにより、(例えば、信号 205 - 0 及び 205 - 1 によってそれぞれ示されるように) 時刻 t_2 と時刻 t_3 の間において、センス線 D とセンス線 D_{__} の間に差動電圧信号が生じる。選択されたセルの電圧は、信号 203 によって表される。エネルギー保存の法則により、(例えば、センス線 D にセルを接続することによって) D と D_{__} の間に差動信号が生じてもエネルギーは消費されない。これは、行信号 204 の活性化 / 非活性化に関連したエネルギーを、当該行に接続された複数のメモリセルにわたって均すことができるためである。

10

【0054】

時刻 t_3 では、センスアンプ (例えば、206) が活性化する (例えば、正の制御信号 231 がハイになり、負の制御信号 228 がローになる)。これにより、D と D_{__} の間の差動信号が増幅され、その結果、論理 1 に対応する電圧 (例えば、VDD) または論理 0 に対応する電圧 (例えば、グラウンド) が、センス線 D に生じる (さらに、もう一方の電圧が相補センス線 D_{__} に生じる)。これにより、検出データ値が、センスアンプ 206 の 1 次ラッチに記憶されるようになる。主なエネルギー消費は、平衡電圧 VDD/2 からレール電圧 VDD までセンス線 D (205 - 0) を充電する際に生じる。

20

【0055】

時刻 t_4 では、パストランジスタ 207 - 1 及び 207 - 2 が、(例えば、制御線 211 - 1 及び 211 - 2 に印加される各制御信号 Passd 及び Passdb を通じて) イネーブルにされる。制御信号 211 - 1 及び 211 - 2 を総称して制御信号 211 と呼ぶ。本明細書で使用する場合、Passd や Passdb などの様々な制御信号は、これらの信号が印加される制御線を参照することによって参照され得る。例えば、Passd 信号を制御信号 211 - 1 と呼ぶことができる。時刻 t_5 では、累算器制御信号 Accumb 及び Accum が、各制御線 212 - 1 及び 212 - 2 を通じて活性化される。後述するように、累算器制御信号 212 - 1 及び 212 - 2 は、次の演算フェーズのために活性化された状態を保ち得る。このようにして本実施例では、制御信号 212 - 1 及び 212 - 2 を活性化することにより、計算素子 231 の 2 次ラッチ (例えば、累算器) が活性化される。センスアンプ 206 に記憶された検出データ値は、2 次ラッチに転送される (例えば、コピーされる)。

30

【0056】

時刻 t_6 では、パストランジスタ 207 - 1 及び 207 - 2 がディセーブルにされる (例えば、ターンオフされる)。しかしながら、累算器制御信号 212 - 1 及び 212 - 2 が活性化された状態を保っているため、累算結果は、2 次ラッチ (例えば、累算器) に記憶される (例えば、ラッチされる)。時刻 t_7 では、行信号 204 - 0 が非活性化され、時刻 t_8 では、メモリアレイのセンスアンプが非活性化される (例えば、センスアンプ制御信号 228 及び 231 が非活性化される)。

40

【0057】

時刻 t_9 では、センス線電圧信号 205 - 0 及び 205 - 1 がこれらの各レール値から平衡電圧 225 (VDD/2) に移ることによって示されているように、センス線 D と D_{__} が平衡状態になる (例えば、平衡信号 226 が活性化される)。この平衡状態では、エ

50

エネルギー保存の法則により、ほとんどエネルギーが消費されない。平衡状態にするには、相補センス線D及びD_{__}を共に短絡して平衡電圧にすることが必要となり得る。本実施例では、この平衡電圧はVDD/2である。平衡状態は、例えば、メモリセルの検出動作の前に生じさせることができる。

【0058】

図2C-1及び2C-2は、それぞれ、本開示のいくつかの実施形態に係る検出回路を用いた複数の論理演算の実行に関するタイミング図285-2及び285-3である。タイミング図285-2及び285-3は、論理演算（例えば、R入力論理演算）の複数の中間演算フェーズの実行に関する信号（例えば、電圧信号）を示す図である。例えば、タイミング図285-2は、R入力NAND演算またはR入力AND演算の複数の中間演算フェーズに対応し、タイミング図285-3は、R入力NOR演算またはR入力OR演算の複数の中間演算フェーズに対応する。例えば、AND演算またはNAND演算を実行することは、図2Bに記載したものなどの初期演算フェーズに続いて、図2C-1に示した演算フェーズを1回以上実行することを含むことができる。同様に、OR演算またはNOR演算を実行することは、図2Bに記載したものなどの初期演算フェーズに続いて、図2C-2に示した演算フェーズを1回以上実行することを含むことができる。

10

【0059】

タイミング図285-2及び285-3に示すように、時刻 t_1 では、平衡状態が不可にされている（例えば、平衡信号226が非活性化されている）。その後、選択された行（例えば、データ値が検出され、当該データ値が第2の入力、第3の入力などの入力として使用されるメモリセルに対応する行）が活性化される。信号204-1は、選択された行（例えば、行204-1）に印加された電圧信号を表す。行信号204-1が、選択されたセルに対応するアクセストランジスタ（例えば、202）の閾電圧（ V_t ）に到達すると、当該アクセストランジスタがターンオンし、このトランジスタにより、選択されたメモリセルに（例えば、当該セルが1T1CのDRAMセルである場合にはコンデンサ203に）センス線Dが接続される。これにより、（例えば、信号205-0及び205-1によってそれぞれ示されるように）時刻 t_2 と時刻 t_3 の間において、センス線Dとセンス線D_{__}の間に差動電圧信号が生じる。選択されたセルの電圧は、信号203によって表される。エネルギー保存の法則により、（例えば、センス線Dにセルを接続することによって）DとD_{__}の間に差動信号が生じても、エネルギーは消費されない。これは、行信号204の活性化/非活性化に関連したエネルギーを、当該行に接続された複数のメモリセルにわたって均すことができるためである。

20

30

【0060】

時刻 t_3 では、センスアンプ（例えば、206）が活性化する（例えば、正の制御信号231がハイになり、負の制御信号228がローになる）。これにより、DとD_{__}の間の差動信号が増幅され、その結果、論理1に対応する電圧（例えば、VDD）または論理0に対応する電圧（例えば、グラウンド）が、センス線Dに生じる（さらに、もう一方の電圧が相補センス線D_{__}に生じる）。これにより、検出データ値が、センスアンプ206の1次ラッチに記憶されるようになる。主なエネルギー消費は、平衡電圧VDD/2からレベル電圧VDDまでセンス線D（205-0）を充電する際に生じる。

40

【0061】

タイミング図285-2及び285-3に示すように、（例えば、選択されたセルが検出された後の）時刻 t_4 では、個々の論理演算に応じて、制御信号211-1（Passd）及び211-2（Passdb）のうち的一方のみが活性化される（例えば、パストランジスタ207-1及び207-2のうち的一方のみがイネーブルにされる）。例えば、タイミング図285-2は、NAND演算またはAND演算の中間フェーズに対応する。そのため、制御信号211-1は時間 t_4 で活性化され、制御信号211-2は非活性化された状態を保つ。逆に、タイミング図285-3は、NOR演算またはOR演算の中間フェーズに対応する。そのため、制御信号211-2は時間 t_4 で活性化され、制御信号211-1は非活性化された状態を保つ。累算器制御信号212-1（Accumb）

50

及び 212 - 2 (Accum) が、図 2 B に記載した初期演算フェーズ中に活性化されたこと、及びこれらの信号が 1 つ以上の中間演算フェーズ中も活性化された状態を保っていることを上記から想起すること。

【 0062 】

累算器を予め活性化していたため、Passd (211 - 1) のみを活性化することにより、電圧信号 205 - 0 に対応するデータ値が累算される。同様に、Passdb (211 - 2) のみを活性化することにより、電圧信号 205 - 1 に対応するデータ値が累算される。例えば、実施例の AND / NAND 演算 (例えば、タイミング図 285 - 2) では Passd (211 - 1) のみが活性化されるが、選択されたメモリセル (例えば、本実施例における行 1 のメモリセル) に記憶されたデータ値が論理 0 である場合、2 次ラッチに関連した累算値がローにアサートされ、それにより、2 次ラッチには論理 0 が記憶される。行 1 のメモリセルに記憶されたデータ値が論理 0 ではない場合、2 次ラッチは、当該ラッチに記憶された行 0 のデータ値 (例えば、論理 1 または論理 0) を保持する。このようにして、この AND / NAND 演算の実施例では、2 次ラッチは、0 (zero) の累算器として機能する。同様に、実施例の OR / NOR 演算 (例えば、タイミング図 285 - 3) では Passdb のみが活性化されるが、選択されたメモリセル (例えば、本実施例における行 1 のメモリセル) に記憶されたデータ値が論理 1 である場合、2 次ラッチに関連した累算値がハイにアサートされ、それにより、2 次ラッチには論理 1 が記憶される。行 1 のメモリセルに記憶されたデータ値が論理 1 ではない場合、2 次ラッチは、当該ラッチに記憶された行 0 のデータ値 (例えば、論理 1 または論理 0) を保持する。このようにして、この OR / NOR 演算の実施例では、2 次ラッチは、実質的に 1 (one) の累算器として機能する。これは、D_{__} の電圧信号 205 - 1 によって累算器の真のデータ値が設定されるためである。

【 0063 】

図 2 C - 1 及び 2 C - 2 に示したものなどの中間演算フェーズの終了時、Passd 信号 (例えば、AND / NAND の場合) または Passdb 信号 (例えば、OR / NOR の場合) が、(例えば、時刻 t5 で) 非活性化され、(例えば、時刻 t6 で) 選択された行が非活性化され、(例えば、時刻 t7 で) センスアンプが非活性化され、(例えば、時刻 t8 で) 平衡状態が生じる。図 2 C - 1 または 2 C - 2 に示したものなどの中間演算フェーズを繰り返して、複数の別の行からの結果を累算することができる。一実施例として、タイミング図 285 - 2 または 285 - 3 のシーケンスを、行 2 のメモリセルについては次の (例えば、第 2 の) 時間に、行 3 のメモリセルについてはその次の (例えば、第 3 の) 時間などに実行することができる。例えば、10 入力 NOR 演算の場合、図 2 C - 2 に示した中間フェーズを 9 回行って 10 入力論理演算のうちの 9 つの入力を与えると共に、(例えば、図 2 B に記載したような) 初期演算フェーズ中に 10 番目の入力を判別することができる。

【 0064 】

図 2 D - 1 及び 2 D - 2 は、それぞれ、本開示のいくつかの実施形態に係る検出回路を用いた複数の論理演算の実行に関するタイミング図 285 - 4 及び 285 - 5 である。タイミング図 285 - 4 及び 285 - 5 は、論理演算 (例えば、R 入力論理演算) の最終演算フェーズの実行に関する信号 (例えば、電圧信号) を示す図である。例えば、タイミング図 285 - 4 は、R 入力 NAND 演算または R 入力 NOR 演算の最終演算フェーズに対応し、タイミング図 285 - 5 は、R 入力 AND 演算または R 入力 OR 演算の最終演算フェーズに対応する。例えば、NAND 演算を実行することは、図 2 C - 1 に関連して記載された中間演算フェーズを複数回反復してから図 2 D - 1 に示した演算フェーズを実行することを含むことができ、NOR 演算を実行することは、図 2 C - 2 に関連して記載された中間演算フェーズを複数回反復してから図 2 D - 1 に示した演算フェーズを実行することを含むことができ、AND 演算を実行することは、図 2 C - 1 に関連して記載された中間演算フェーズを複数回反復してから図 2 D - 2 に示した演算フェーズを実行することを含むことができ、OR 演算を実行することは、図 2 C - 2 に関連して記載された中間演算

10

20

30

40

50

フェーズを複数回反復してから図2D-2に示した演算フェーズを実行することを含むことができる。以下に示した表5は、本明細書に記載された複数の実施形態に係る複数のR入力論理演算の実行に関連した演算フェーズのシーケンスに対応する図を示す。

【表5】

演算	図2B	図2C-1	図2C-2	図2D-1	図2D-2
AND	第1フェーズ	R-1 反復			最終フェーズ
NAND	第1フェーズ	R-1 反復		最終フェーズ	
OR	第1フェーズ		R-1 反復		最終フェーズ
NOR	第1フェーズ		R-1 反復	最終フェーズ	

10

20

【0065】

図2D-1及び2D-2の最終演算フェーズは、R入力論理演算の結果をメモリアレイ（例えば、アレイ230）の行に記憶させることに関連して記載されている。しかしながら、前述のように、いくつかの実施形態では、この結果を、当該アレイに再度記憶させる以外にも、適切な場所に（例えば、I/O線を介して、コントローラ及び/またはホストプロセッサに関連した外部レジスタや、異なるメモリデバイスのメモリアレイに）記憶させることができる。

【0066】

タイミング図285-4及び285-5に示すように、時刻 t_1 では、平衡状態が不可とされている（例えば、平衡信号226が非活性化されている）ため、センス線D及びD_{__}がフローティングになっている。時間 t_2 では、どの論理演算が実行されているかに応じて、InvD信号213またはPassd及びPassdb信号211が活性化される。本実施例では、NAND演算またはNOR演算（図2D-1を参照）の場合にはInvD信号213が活性化され、AND演算またはOR演算（図2D-2を参照）の場合にはPassd及びPassdb信号211が活性化される。

30

【0067】

（例えば、NAND演算またはNOR演算に関連して）時刻 t_2 でInvD信号213を活性化することにより、トランジスタ214-1/214-2がイネーブルになり、2次ラッチに記憶されたデータ値が反転する。このとき、センス線Dまたはセンス線D_{__}がローに引かれる。このようにして、信号213を活性化することにより、累算出力が反転する。従って、NAND演算の場合、先の演算フェーズ（例えば、初期演算フェーズ及び1つ以上の中間演算フェーズ）で検出されたメモリセルのいずれかが論理0を記憶していた場合（例えば、NAND演算のR入力のいずれかが論理0であった場合）、センス線D_{__}は、論理0に対応する電圧（例えば、グラウンド電圧）を伝達し、センス線Dは、論理1に対応する電圧（例えば、VDDなどの電源電圧）を伝達する。このNANDの実施例の場合、従来の演算フェーズで検出されたメモリセルの全てが論理1を記憶していた（例えば、NAND演算のR入力の全てが論理1であった）場合、センス線D_{__}は、論理1に対応する電圧を伝達し、センス線Dは、論理0に対応する電圧を伝達する。次に、時刻 t_3 では、センスアンプ206の1次ラッチが活性化される（例えば、センスアンプを作動させる）ことにより、D及びD_{__}が、適切なレールに駆動される。この時、センス線Dは

40

50

、先の演算フェーズ中に検出されたメモリセルから判別された各入力データ値をNANDした結果を伝達する。このようにして、センス線Dは、入力データ値のいずれかが論理0である場合にはVDDになり、入力データ値の全てが論理1である場合にはグラウンドになる。

【0068】

NOR演算の場合、先の演算フェーズ（例えば、初期演算フェーズ及び1つ以上の中間演算フェーズ）で検出されたメモリセルのいずれかが論理1を記憶していた場合（例えば、NOR演算のR入力のいずれかが論理1であった場合）、センス線D_{__}は、論理1に対応する電圧（例えば、VDD）を伝達し、センス線Dは、論理0に対応する電圧（例えば、グラウンド）を伝達する。このNORの実施例の場合、先の演算フェーズで検出されたメモリセルの全てが論理0を記憶していた（例えば、NOR演算のR入力の全てが論理0であった）場合、センス線D_{__}は、論理0に対応する電圧を伝達し、センス線Dは、論理1に対応する電圧を伝達する。次に、時刻t3では、センスアンプ206の1次ラッチが活性化される。この時、センス線Dは、先の演算フェーズ中に検出されたメモリセルから判別された各入力データ値をNORした結果を保持する。このようにして、センス線Dは、入力データ値のいずれかが論理1である場合にはグラウンドになり、入力データ値の全てが論理0である場合にはVDDになる。

10

【0069】

図2D-2を参照すると、（例えば、ANDまたはOR演算に関連して）Passd及びPassdb信号211を活性化することにより、計算素子231の2次ラッチに記憶された累算出力がセンスアンプ206の1次ラッチに転送される。例えば、AND演算の場合、先の演算フェーズ（例えば、図2Bの第1演算フェーズ、及び図2C-1の中間演算フェーズの1つ以上の反復）で検出されたメモリセルのいずれかが論理0を記憶していた場合（例えば、AND演算のR入力のいずれかが論理0であった場合）、センス線D_{__}は、論理1に対応する電圧（例えば、VDD）を伝達し、センス線Dは、論理0に対応する電圧（例えば、グラウンド）を伝達する。このANDの実施例の場合、先の演算フェーズで検出されたメモリセルの全てが論理1を記憶していた（例えば、AND演算のR入力の全てが論理1であった）場合、センス線D_{__}は、論理0に対応する電圧を伝達し、センス線Dは、論理1に対応する電圧を伝達する。次に、時刻t3では、センスアンプ206の1次ラッチが活性化される。この時、センス線Dは、先の演算フェーズ中に検出されたメモリセルから判別された各入力データ値をANDした結果を伝達する。このようにして、センス線Dは、入力データ値のいずれかが論理0である場合にはグラウンドになり、入力データ値の全てが論理1である場合にはVDDになる。

20

30

【0070】

OR演算の場合、先の演算フェーズ（例えば、図2Bの第1演算フェーズ、及び図2C-2に示した中間演算フェーズの1つ以上の反復）で検出されたメモリセルのいずれかが論理1を記憶していた場合（例えば、OR演算のR入力のいずれかが論理1であった場合）、センス線D_{__}は、論理0に対応する電圧（例えば、グラウンド）を伝達し、センス線Dは、論理1に対応する電圧（例えば、VDD）を伝達する。このORの実施例の場合、先の演算フェーズで検出されたメモリセルの全てが論理0を記憶していた（例えば、OR演算のR入力の全てが論理0であった）場合、センス線D_{__}は、論理0に対応する電圧を伝達し、センス線Dは、論理1に対応する電圧を伝達する。次に、時刻t3では、センスアンプ206の1次ラッチが活性化される。この時、センス線Dは、先の演算フェーズ中に検出されたメモリセルから判別された各入力データ値をORした結果を伝達する。このようにして、センス線Dは、入力データ値のいずれかが論理1である場合にはVDDになり、入力データ値の全てが論理0である場合にはグラウンドになる。

40

【0071】

次に、R入力AND、OR、NAND及びNOR演算の結果を、アレイ230のメモリセルに再度記憶させることができる。図2D-1及び2D-2に示した実施例では、R入力論理演算の結果を、行R（例えば、204-R）に接続されたメモリセルに記憶させる

50

。論理演算の結果を行 R のメモリセルに記憶させるためには、行 R を活性化することによって行 R のアクセストラジスタ 202 をイネーブルにすることが必要となるにすぎない。行 R のメモリセルのコンデンサ 203 は、センス線 D のデータ値（例えば、論理 1 または論理 0）に対応する電圧に駆動される。これにより、それまで行 R のメモリセルに記憶されていたどのようなデータ値も実質的に上書きされる。なお、行 R のメモリセルは、論理演算用の入力として使用されたデータ値を記憶していたメモリセルと同一にすることができることに注意されたい。例えば、論理演算の結果を、行 0 のメモリセルまたは行 1 のメモリセルに再度記憶させることができる。

【0072】

タイミング図 285 - 4 及び 285 - 5 は、時刻 t3 にて、正の制御信号 231 及び負の制御信号 228 が非活性化されて（例えば、信号 231 がハイになり、信号 228 がローになる）、センスアンプ 206 が活性化される場合を示している。時刻 t4 では、時刻 t2 で活性化された各信号（例えば、213 または 211）が非活性化される。なお、実施形態は、本実施例に限定されない。例えば、いくつかの実施形態では、センスアンプ 206 を、時刻 t4 の後（例えば、信号 213 または信号 211 が非活性化された後）に活性化してもよい。

10

【0073】

図 2D - 1 及び 2D - 2 に示すように、時刻 t5 にて、行 R (204 - R) が活性化される。これにより、選択されたセルのコンデンサ 203 が、累算器に記憶された論理値に対応する電圧に駆動される。時刻 t6 では、行 R が非活性化され、時刻 t7 では、センスアンプ 206 が非活性化され（例えば、信号 228 及び 231 が非活性化される）、時刻 t8 にて平衡状態が生じる（例えば、信号 226 が活性化されて、相補センス線 205 - 0 / 205 - 1 の各電圧が平衡電圧に至る）。

20

【0074】

いくつかの実施形態では、図 2A に記載されたもののような検出回路（例えば、メモリセルと間隔を置いて形成された回路）により、多数の論理演算を並列に実行することが可能となる。例えば、16K の列を有するアレイでは、当該アレイ及び検出回路からバスを介してデータを転送することなく、かつ/または当該アレイ及び検出回路から I/O 線を介してデータを転送することなく、16K の論理演算を並列に実行することができる。

30

【0075】

また、当業者は、R 入力論理演算（例えば、NAND、AND、NOR、OR など）を実行できることにより、主要な数学的機能、及び/またはパターン比較機能の中でも、加算、減算、乗算及び除算などのより複雑な演算処理機能の実行が可能となることを理解するであろう。例えば、一連の NAND 演算を組み合わせ、全加算器機能を実行することができる。一実施例として、キャリーイン及びキャリーアウトと共に 2 つのデータ値を加算するために 12 個の NAND ゲートが全加算器に必要な場合、2 つの 32 ビット数を加えるためには、合計で 384 回の NAND 演算 (12 × 32) が実行される可能性がある。本開示の実施形態は、ブール代数でなくともよい論理演算（例えば、コピー、比較など）の実行に使用することもできる。

40

【0076】

加えて、いくつかの実施形態では、実行される論理演算への入力は、検出回路（例えば、150）が接続されているメモリアレイに記憶されたデータ値でなくともよい。例えば、論理演算への複数の入力を、メモリアレイ（例えば、230）の行を活性化せずにセンスアンプ（例えば、206）によって検出することができる。一実施例として、センスアンプ 206 が、これに接続された I/O 線を介して上記複数の入力を受け取ることができる。かかる入力は、例えば、ホストプロセッサ（例えば、ホスト 110）及び/または外部コントローラからのように、メモリアレイ 230 の外部ソースから、（例えば、適切な I/O 線を介して）センスアンプ 206 に提供されてもよい。別の実施例として、論理演算の実行に関連して、特定のセンスアンプ（例えば、206）及びそれに対応する計算素子（例えば、231）への入力を、異なるセンスアンプ/計算素子のペアから受け取って

50

もよい。例えば、セルの第 1 列に接続された第 1 累算器に記憶されたデータ値（例えば、論理的結果）を、セルの異なる列に関連した異なる（例えば、隣接した）センスアンプ / 計算素子のペアに転送してもよい。このペアは、第 1 列と同一のアレイに位置していてもよく、位置していなくてもよい。

【 0 0 7 7 】

本開示の実施形態は、図 2 A に示した特定の検出回路構成に限定されない。例えば、異なる計算素子回路を使用して、本明細書に記載されたいくつかの実施形態に係る論理演算を実行することができる。図 2 A には示されていないが、いくつかの実施形態では、アレイ 2 3 0、センスアンプ 2 0 6 及び / または計算素子 2 3 1 に制御回路を接続することができる。このような制御回路は、例えば、当該アレイ及び検出回路と同一のチップに、かつ / または外部プロセッサなどの外部処理リソースに実装されてもよく、本明細書に記載された論理演算を実行するために、上記アレイ及び検出回路に対応する様々な信号のイネーブル / ディセーブルを制御することができる。

10

【 0 0 7 8 】

図 2 A、2 B、2 C - 1、2 C - 2、2 D - 1 及び 2 D - 2 に関連して記載された論理演算フェーズの実施例では、データ値（例えば、メモリセルから検出されたデータ値及び / またはセンス線の電圧または電流に対応するデータ値）を累算することが必要となる。エネルギー保存の法則により、論理演算フェーズを実行する際に消費されるエネルギーは、センス線 D または D_{__} の容量を VDD / 2 から VDD まで充電する間に消費されるエネルギーにほぼ等しい。この充電は、（例えば、図 2 B、2 C - 1、2 C - 2、2 D - 1 及び 2 D - 2 に示した時刻 t₃ にて）センスアンプが活性化されるときに開始される。このように、論理演算を実行することにより、近似的には、センス線（例えば、桁線）を VDD / 2 から VDD まで充電するのに用いられるエネルギーが消費される。一方、従来の様々な処理手法では、往々にして、レールからレールまで（例えば、グラウンドから VDD まで）センス線を充電するのに用いられるエネルギー量が少なくとも消費されるが、これは、本明細書に記載された実施形態と比べて 2 倍以上となる場合がある。

20

【 0 0 7 9 】

図 3 A は、本開示のいくつかの実施形態に係る一連の除算演算の実行に関連した特定のフェーズにおける一部のアレイ 3 3 0 のセルの状態を示す論理図である。図 3 A は、図 2 A のメモリアレイ 2 3 0 及び図 1 のメモリアレイ 1 3 0 に類似したメモリアレイ 3 3 0 を含む。メモリアレイ 3 3 0 は、センス線 3 0 5 - 0、3 0 5 - 1、3 0 5 - 2、3 0 5 - 3、3 0 5 - 4、3 0 5 - 5（例えば、通常、センス線 3 0 5 と呼ばれる）を含む。これよりも多くのまたは少ないセンス線をメモリアレイ 3 3 0 に含めることができる。各センス線 3 0 5 は、異なる除算演算を表す。例えば、6 つの異なる除算演算を、6 本の異なるセンス線に接続されている複数のメモリセルに記憶された値に基づき、（例えば、並列に）実行することができる。

30

【 0 0 8 0 】

メモリアレイ 3 3 0 は、選択線 3 0 4 - 0、3 0 4 - 1、3 0 4 - 2、3 0 4 - 3、3 0 4 - 4、3 0 4 - 5、3 0 4 - 6、3 0 4 - 7、3 0 4 - 8、3 0 4 - 9、3 0 4 - 10、3 0 4 - 11、3 0 4 - 12、3 0 4 - 13、3 0 4 - 14、3 0 4 - 15、3 0 4 - 16（例えば、通常、選択線 3 0 4 と呼ばれる）も含む。図 3 A では、第 1 の除算演算は、被除数値 3 2 0 - 0 を除数値 3 2 2 - 0 で割ることによって構成することができる。この第 1 の除算演算の結果を、商の値 3 2 4 - 0 及び剰余値 3 2 6 - 0 に記憶させることができる。第 2 の除算演算は、被除数値 3 2 0 - 1 を除数値 3 2 2 - 1 で割ることによって構成することができる。この第 2 の除算演算の結果を、商の値 3 2 4 - 1 及び剰余値 3 2 6 - 1 に記憶させることができる。第 3 の除算演算は、被除数値 3 2 0 - 2 を除数値 3 2 2 - 2 で割ることによって構成することができる。この第 3 の除算演算の結果を、商の値 3 2 4 - 2 及び剰余値 3 2 6 - 2 に記憶させることができる。第 4 の除算演算は、被除数値 3 2 0 - 3 を除数値 3 2 2 - 3 で割ることによって構成することができる。この第 4 の除算演算の結果を、商の値 3 2 4 - 3 及び剰余値 3 2 6 - 3 に記憶させることができる。

40

50

。第5の除算演算は、被除数値320-4を除数値322-4で割ることによって構成することができる。この第5の除算演算の結果を、商の値324-4及び剰余値326-4に記憶させることができる。第6の除算演算は、被除数値320-5を除数値322-5で割ることによって構成することができる。この第6の除算演算の結果を、商の値324-5及び剰余値326-5に記憶させることができる。

【0081】

被除数値320-0、320-1、320-2、320-3、320-4、320-5（例えば、通常、被除数値320と呼ばれる）、除数値322-0、322-1、322-2、322-3、322-4、322-5（例えば、通常、除数値322と呼ばれる）、商の値324-0、324-1、324-2、324-3、324-4、324-5、及び剰余値326-0、326-1、326-2、326-3、326-4、326-5を、ビットベクトルとして、選択線304及びセンス線305に接続されているメモリセルに記憶させることができる。例えば、複数のメモリセルに記憶されている被除数値320-0は、5に等しい値を有することができる。被除数値320-0は、ビットベクトル[0101]によって表すことができ、このビットベクトルを、センス線320-0及び選択線304-0、304-1、304-2、304-3に接続されているメモリセルに記憶させることができる。

10

【0082】

図3Aは、メモリアレイ330内のメモリセルの初期状態を示している。例えば、商の値及び剰余値を記憶するメモリセルは、0の値を表すビットベクトル[0000]を用いて初期化することができる。被除数値320-0を記憶するメモリセルは、初期化されてビットベクトル[0101]を記憶する。被除数値320-0は5に等しい。被除数値320-1を記憶するメモリセルは、初期化されてビットベクトル[0111]を記憶する。被除数値320-1は7に等しい。被除数値320-2を記憶するメモリセルは、初期化されてビットベクトル[0011]を記憶する。被除数値320-2は3に等しい。被除数値320-3を記憶するメモリセルは、初期化されてビットベクトル[0110]を記憶する。被除数値320-3は6に等しい。被除数値320-4を記憶するメモリセルは、初期化されてビットベクトル[1000]を記憶する。被除数値320-4は8に等しい。被除数値320-5を記憶するメモリセルは、初期化されてビットベクトル[1100]を記憶する。被除数値320-5は12に等しい。

20

30

【0083】

除数値322-0を記憶するメモリセルは、初期化されてビットベクトル[0100]を記憶する。記憶された除数値322-0は、4に等しい。除数値322-1を記憶するメモリセルは、初期化されてビットベクトル[0010]を記憶する。除数値322-1は2に等しい。除数値322-2を記憶するメモリセルは、初期化されてビットベクトル[0001]を記憶する。除数値322-2は1に等しい。除数値322-3を記憶するメモリセルは、初期化されてビットベクトル[0011]を記憶する。除数値322-3は3に等しい。除数値322-4を記憶するメモリセルは、初期化されてビットベクトル[0101]を記憶する。除数値322-4は5に等しい。除数値322-5を記憶するメモリセルは、初期化されてビットベクトル[0110]を記憶する。除数値322-5は6に等しい。

40

【0084】

図3Aは、マスク328-0、328-1、328-2、328-3、328-4、328-5（例えば、通常、マスク328と呼ばれる）も含む。マスク328については、図3Cにおいてさらに説明する。

【0085】

図3B～3Kは、本開示のいくつかの実施形態に係る除算演算の実行に関連した種々のフェーズにわたる、（例えば、計算後の）アレイ部330のメモリセルの状態を示す論理図である。図3Aは、アレイ部330のメモリセルの最初の（例えば、初期化された）状態を示している。

50

【 0 0 8 6 】

図 3 B は、除算演算の実行に関連した特定のフェーズにおける（例えば、図 3 A に示したものに続くフェーズにおける）アレイ部 3 3 0 のセルの状態を示す論理図である。

【 0 0 8 7 】

除算演算を実行する際の第 1 計算は、被除数値 3 2 0 を表すビットベクトルの第 4 インデックスに関連付けられたメモリセルに記憶されたビットを、剰余値 3 2 6 を表すビットベクトルの第 1 インデックスに関連付けられたメモリセルにコピーすることである。例えば、被除数値 3 2 0 - 0 を記憶するメモリセルのうち、図 3 A のセンス線 3 0 5 - 0 及び選択線 3 0 4 - 3 に接続されているメモリセルに記憶されたビット（例えば、0）を、剰余値 3 2 6 - 0 を記憶するメモリセルのうちのセンス線 3 0 5 - 0 及び選択線 3 0 4 - 1 2 に接続されているメモリセルにコピーする。被除数値 3 2 0 - 1 を記憶するメモリセルのうち、図 3 A のセンス線 3 0 5 - 1 及び選択線 3 0 4 - 3 に接続されているメモリセルに記憶されたビット（例えば、0）を、剰余値 3 2 6 - 1 を記憶するメモリセルのうちのセンス線 3 0 5 - 1 及び選択線 3 0 4 - 1 2 に接続されているメモリセルにコピーする。被除数値 3 2 0 - 1 を記憶するメモリセルのうち、図 3 A のセンス線 3 0 5 - 2 及び選択線 3 0 4 - 3 に接続されているメモリセルに記憶されたビット（例えば、0）を、剰余値 3 2 6 - 2 を記憶するメモリセルのうちのセンス線 3 0 5 - 2 及び選択線 3 0 4 - 1 2 に接続されているメモリセルにコピーする。被除数値 3 2 0 - 3 を記憶するメモリセルのうち、図 3 A のセンス線 3 0 5 - 3 及び選択線 3 0 4 - 3 に接続されているメモリセルに記憶されたビット（例えば、0）を、剰余値 3 2 6 - 3 を記憶するメモリセルのうちのセンス線 3 0 5 - 3 及び選択線 3 0 4 - 1 2 に接続されているメモリセルにコピーする。被除数値 3 2 0 - 4 を記憶するメモリセルのうち、図 3 A のセンス線 3 0 5 - 4 及び選択線 3 0 4 - 3 に接続されているメモリセルに記憶されたビット（例えば、1）を、剰余値 3 2 6 - 4 を記憶するメモリセルのうちのセンス線 3 0 5 - 4 及び選択線 3 0 4 - 1 2 に接続されているメモリセルにコピーする。被除数値 3 2 0 - 5 を記憶するメモリセルのうち、図 3 A のセンス線 3 0 5 - 5 及び選択線 3 0 4 - 3 に接続されているメモリセルに記憶されたビット（例えば、1）を、剰余値 3 2 6 - 5 を記憶するメモリセルのうちのセンス線 3 0 5 - 5 及び選択線 3 0 4 - 1 2 に接続されているメモリセルにコピーする。従って、このコピー演算の後、剰余値 3 2 6 - 0、...、3 2 6 - 5 を表すビットベクトルは、[0 0 0 0]、[0 0 0 0]、[0 0 0 0]、[0 0 0 0]、[0 0 0 1] 及び [0 0 0 1] にそれぞれ更新されて、更新後の剰余値 0、0、0、0、1 及び 1 をそれぞれ表す。

【 0 0 8 8 】

図 3 C は、除算演算の実行に関連した特定のフェーズにおける（例えば、図 3 B に示したものに続くフェーズにおける）アレイ部 3 3 0 のセルの状態を示す論理図である。

【 0 0 8 9 】

図 3 C では、複数のマスク 3 2 8 が作成される。本明細書で使用する場合、マスクは、2 値を比較した（例えば、被除数値を剰余値と比較した）後に作成されるマスク、及び / または複数の値を比較した後に作成されるマスクとすることができる。例えば、マスク 3 2 8 - 0 は、除数値 3 2 2 - 0 を剰余値 3 2 6 - 0 と比較した結果を示すことができる。マスク 3 2 8 - 1 は、除数値 3 2 2 - 1 を剰余値 3 2 6 - 1 と比較した結果を示すことができる。マスク 3 2 8 - 2 は、除数値 3 2 2 - 2 を剰余値 3 2 6 - 2 と比較した結果を示すことができる。マスク 3 2 8 - 3 は、除数値 3 2 2 - 3 を剰余値 3 2 6 - 3 と比較した結果を示すことができる。マスク 3 2 8 - 4 は、除数値 3 2 2 - 4 を剰余値 3 2 6 - 4 と比較した結果を示すことができる。マスク 3 2 8 - 5 は、除数値 3 2 2 - 5 を剰余値 3 2 6 - 5 と比較した結果を示すことができる。比較により、除数値が剰余値以下であるかどうかを判定することができる。マスク 3 2 8 - 0、3 2 8 - 1、3 2 8 - 2、3 2 8 - 3、3 2 8 - 4、3 2 8 - 5（例えば、通常、マスク 3 2 8 と呼ばれる）を記憶するメモリセルは、除数値が剰余値よりも大きいときに 0 に設定することができ、除数値が剰余値以下であるときには 1 に設定される。除数値 3 2 2 がいずれも剰余値 3 2 6 以下ではないため、図 3 C のマスク 3 2 8 は、全て 0 に設定される。例えば、除数値 3 2 2 - 5（例えば

、6)が剰余値326-5(例えば、1)よりも大きいため、マスク328-5のメモリセルは0に設定される。

【0090】

図3Dは、除算演算の実行に関連した特定のフェーズにおける(例えば、図3Cに示したものに続くフェーズにおける)アレイ部330のセルの状態を示す論理図である。

【0091】

図3Dでは、0に等しい値を有するマスク328に関連する商の値324の論理表現(例えば、ビットベクトル)が、商の値を更新するためにシフトされる。すなわち、除数値322が剰余値326よりも大きいため、商の値324の論理表現がシフトされる。いくつかの実施例では、シフトは、左シフト及び/または下へのシフトとして表すこともできる。シフトは、メモリセルに記憶されたビットを、当該ビットが当初記憶されたメモリセルよりも大きいインデックスに関連した異なるメモリセルにコピーすることを含むことができる。例えば、センス線305-0及び選択線304-8に接続されているメモリセルに記憶されたビット(例えば、0)を、センス線305-0及び選択線304-9に接続されているメモリセルにコピーするときシフトが行われる。上記の実施例では、センス線305-0及び選択線304-8に接続されているメモリセルは、1に等しいインデックスに関連付けられている。このインデックスは、センス線305-0及び選択線304-9に接続されているメモリセルに関連付けられた、2に等しいインデックスよりも小さい。

10

【0092】

図3Eは、除算演算の実行に関連した特定のフェーズにおける(例えば、図3Dに示したものに続くフェーズにおける)アレイ部330のセルの状態を示す論理図である。

20

【0093】

図3Eでは、全てのマスク328が0に等しい値を有するため、(剰余値324を更新するために)全ての剰余値324の論理表現が同様にシフトされる。例えば、センス線305-4及び選択線304-12~304-15に接続されたメモリセルに記憶されたビットは、センス線305-4及び選択線304-13~304-16に接続されたメモリセルにシフトされる。センス線305-4及び選択線304-12に接続されたメモリセルに記憶されたビットは、センス線305-4及び選択線304-13に接続されたメモリセルにシフトされる(例えば、コピーされる)。センス線305-4及び選択線304-12に接続されたメモリセルは、当該メモリセルに記憶されたビットを0に等しく設定することによって空にされる。(例えば、ビットベクトル[0001]からビットベクトル[0010]に)シフトすることにより、剰余値326-4及び326-5が更新されて、2(two)の値を有ようになる。

30

【0094】

この状態が、除算演算の第1反復の終わりである。除算演算の第2反復(図3A~3Kには図示せず)は、センス線305-0~センス線305-5及び選択線304-2に接続されているメモリセルに記憶されたビットを、センス線305-0~センス線305-5及び選択線304-12に接続されているメモリセルにコピーすることを含む。この第2反復は、マスクを再作成することを含むこともできる。第2反復は、マスクが1または0に等しい値を有するかどうかを判定することを含むことができる。第2反復は、マスクが1に等しい値を有すると判定された場合、剰余値から除数値を引くこと、及びその結果の論理表現を剰余値として記憶させることを含む。マスクが1(one)に等しい値を有する場合、商の値の論理表現(ビットベクトル)がシフトされ、1(one)だけインクリメントされて、商の値が更新される。マスクが1(one)に等しい値を有する場合、かつ、センス線305-0~センス線305-5及び選択線304-2に接続されているメモリセルが第1インデックスを有しない場合、上記の減算演算の結果に対する論理表現をシフトして剰余値を更新する。

40

【0095】

図3Fは、除算演算の実行に関連した特定のフェーズにおける(例えば、図3Eに示し

50

たもの続くフェーズにおける)アレイ部330のセルの状態を示す論理図である。

【0096】

図3Fでは、被除数値320を表すビットベクトルの第1インデックスに関連付けられたメモリセルに記憶されたビットを、剰余値326を表すビットベクトルの第1インデックスに関連付けられたメモリセルにコピーする。例えば、センス線320-0~センス線320-5及び選択線304-0に接続されたメモリセルに記憶されたビットを、センス線320-0~センス線320-5及び選択線304-12に接続されたメモリセルにコピーする。

【0097】

図3Gは、除算演算の実行に関連した特定のフェーズにおける(例えば、図3Fに示したものに続くフェーズにおける)アレイ部330のセルの状態を示す論理図である。

10

【0098】

図3Gでは、マスク328が再作成されている。例えば、除数値322-0(例えば、4)が剰余値326-0(例えば、5)未満であるため、マスク328-0は1に等しい。除数値322-3(例えば、3)が剰余値326-6(例えば、0)よりも大きいので、マスク328-3は0に等しい。

【0099】

図3Hは、除算演算の実行に関連した特定のフェーズにおける(例えば、図3Gに示したものに続くフェーズにおける)アレイ部330のセルの状態を示す論理図である。

【0100】

図3Hでは、関連マスクが1に等しかった剰余値326から除数値322が引かれている。例えば、除数値322-0(例えば、4)は、図3Gの剰余値326-0(例えば、5)から引かれ、その結果は、図3Hに示すように、剰余値326-0(例えば、1)を記憶するメモリセルに記憶された。同様に、除数値322-4(例えば、5)は、図3Gの剰余値326-4(例えば、8)から引かれ、この減算結果は、図3Hの剰余値326-4(例えば、3)を記憶するメモリセルに記憶された。しかしながら、除数値322-3及び除数値322-5は、剰余値326-3及び326-5から引かれなかったが、これは、関連マスク328-3及びマスク328-5が0に等しかったためである。

20

【0101】

図3Iは、除算演算の実行に関連した特定のフェーズにおける(例えば、図3Hに示したものに続くフェーズにおける)アレイ部330のセルの状態を示す論理図である。

30

【0102】

図3Iでは、商の値324-0、324-1、324-2及び324-4の論理表現がシフトされ、1だけインクリメントされている。これは、対応するマスク328-0、328-1、328-2及び324-4が1に等しかったためである。例えば、図3Hの商の値324-1(例えば、1)を記憶するメモリセルに記憶されたビットが下にシフトされて、この商の値が一時的に2の値に更新される。図3Iでは、(例えば、一時的に2の)商の値324-1の論理表現が1だけインクリメントされて、図3Iの商の値は3に等しい値に更新される。

【0103】

図3Iでは、剰余値326-0、326-1、326-2、326-4の論理表現は、関連マスクの値328-0、328-1、328-2、328-4が1に等しくてもシフトされない。これは、図3Fにおいてコピーされたこれらのビットが、1に等しいインデックスを有する、センス線320-0~センス線320-5及び選択線304-0に接続されているメモリセルからコピーされたためである。すなわち、コピーするビットがこれ以上ないため、剰余値326-0、326-1、326-2、326-4は、除算演算の最終的な剰余値である。同様に、商の値324-0、324-1、324-2、324-4は、除算演算の最終的な商の値である。

40

【0104】

しかしながら、図3J及び図3Kに示すように、0に等しい値を有するマスクに関連す

50

る商の値及び剰余値の論理表現はシフトされる。

【0105】

図3Jは、除算演算の実行に関連した特定のフェーズにおける（例えば、図3Iに示したものに続くフェーズにおける）アレイ部330のセルの状態を示す論理図である。

【0106】

図3Jでは、商の値 $324 - 3$ 、 $324 - 5$ の論理表現が左にシフトされる。例えば、図3Iの商の値 $324 - 3$ 、 $324 - 5$ は、これらが左にシフトされる前には1に等しい値を有し、これらが左にシフトされた後には2に等しい値を有した。

【0107】

図3Kは、除算演算の実行に関連した特定のフェーズにおける（例えば、図3Jに示したものに続くフェーズにおける）アレイ部330のセルの状態を示す論理図である。

【0108】

図3Kでは、剰余値 $326 - 3$ 、 $326 - 5$ の論理表現が左にシフトされる。例えば、図3Jの剰余値 $326 - 3$ 、 $326 - 5$ は、これらが左にシフトされる前には0に等しい値を有し、これらが左にシフトされた後には0に等しい値を有した。このようにシフト操作を行っても、剰余値 $326 - 3$ 、 $326 - 5$ は影響を受けない。これは、これらの剰余値が、このシフト操作の前に0に等しく、結果として、これらの剰余値に0ビットがシフトされても、これらの剰余値は影響を受けないためである。

【0109】

[結論]

本開示は、センス線を用いた除算のための装置及び方法を含む。実施例の装置は、センス線及び第1の本数の選択線に接続された第1の数のメモリセルを含むメモリアレイの第1アドレス空間を備える。第1アドレス空間は、被除数値を記憶する。メモリアレイの第2アドレス空間は、センス線及び第2の本数の選択線に接続された第2の数のメモリセルを備える。第2アドレス空間は、除数値を記憶する。メモリアレイの第3アドレス空間は、センス線及び第3の本数の選択線に接続された第3の数のメモリセルを備える。第3アドレス空間は、剰余値を記憶する。検出回路は、被除数値及び除数値を受け取り、被除数値を除数値で割り、除算の剰余結果を第3の数のメモリセルに記憶させるように構成することができる。

【0110】

本明細書では特定の実施形態について例示し、図示してきたが、当業者は、図示された特定の実施形態を、同一結果を得るために意図された配置構成で置き換えることができることを理解するであろう。本開示は、本開示の1つ以上の実施形態の適応または変形を包含することを意図したものである。上記の説明は、例示的になされたものであり、限定的になされたものではないことを理解すべきである。上記の実施形態と、本明細書に記載されていない他の実施形態との組み合わせは、上記の説明を考察すれば当業者にとって明らかとなるであろう。本開示の1つ以上の実施形態の範囲には、上記の構造及び方法を使用する他の適用例が含まれる。従って、本開示の1つ以上の実施形態の範囲は、添付された特許請求の範囲、並びにかかる特許請求の範囲の権利が与えられる均等物の全範囲を参照して定められるべきである。

【0111】

前述の詳細な説明では、本開示を簡素化するために、単一の実施形態にいくつかの特徴をまとめている。本開示の方法は、開示された本開示の実施形態が、各請求項に明示的に記載されたものよりも多くの特徴を用いる必要があるという意図を反映したものであるとして解釈すべきではない。むしろ、以下の特許請求の範囲が反映するように、発明の主題は、単一の開示された実施形態の全ての特徴より少ないところにある。従って、以下の特許請求の範囲は、各請求項が別個の実施形態として独自に存在する状態で、本明細書によって発明を実施するための形態に援用される。

【 2 B 】

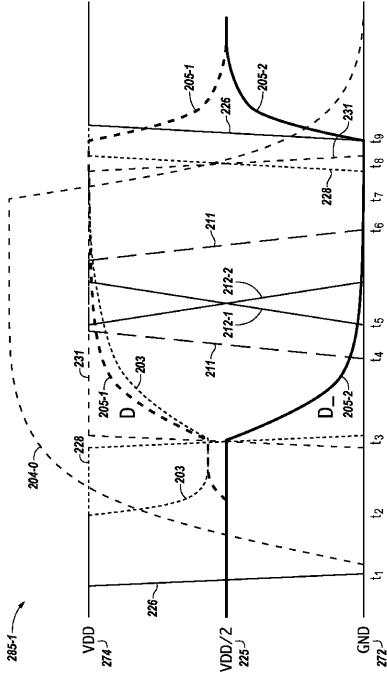


Fig. 2B

【 2 C - 1 】

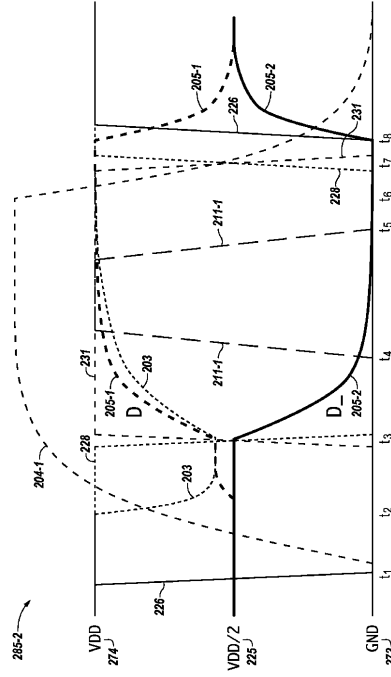


Fig. 2C-1

【 2 C - 2 】

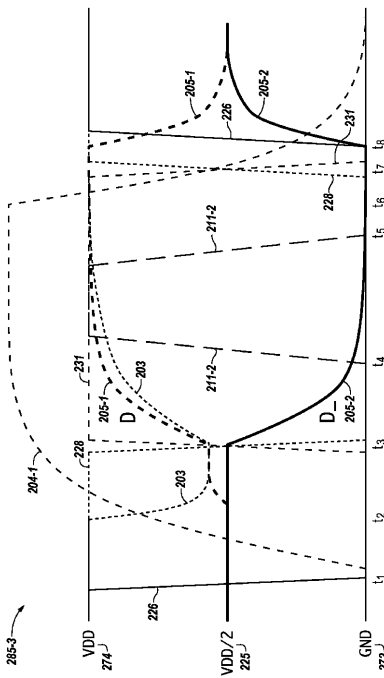


Fig. 2C-2

【 2 D - 1 】

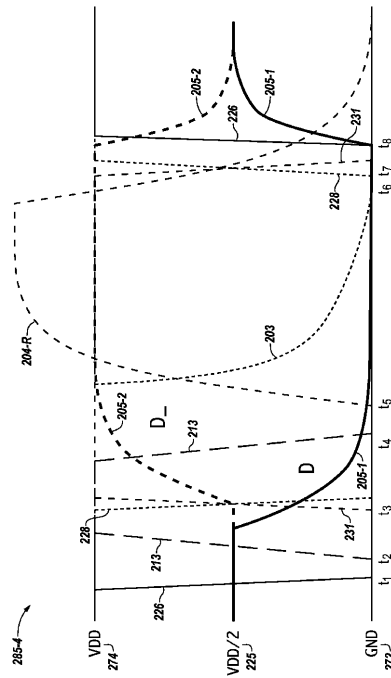


Fig. 2D-1

【 図 2 D - 2 】

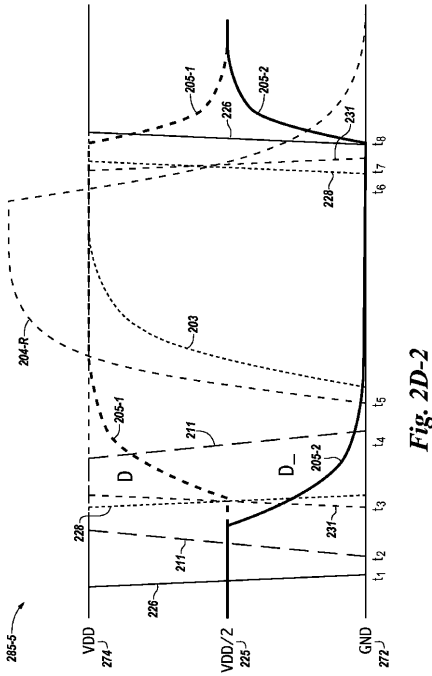
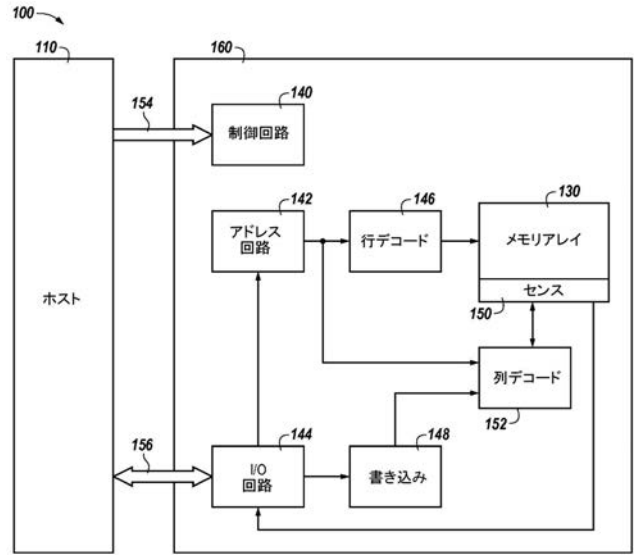
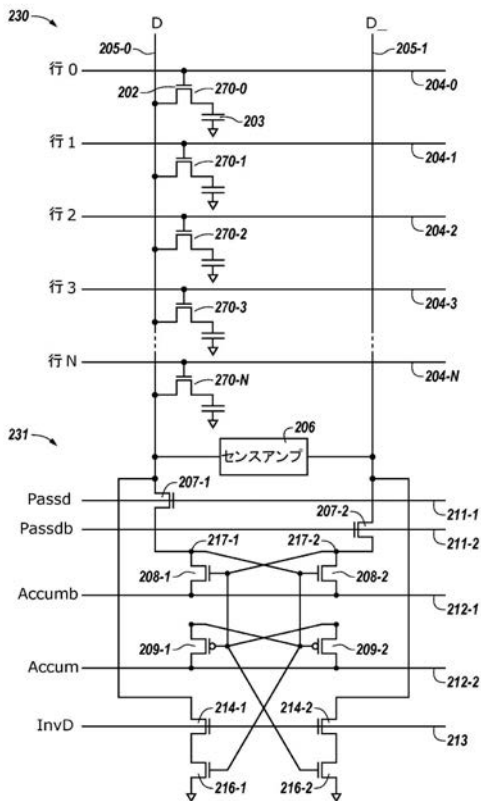


Fig. 2D-2

【 図 1 】



【 図 2 A 】



【 図 3 A 】

330

メモリアレイ

	320-0	320-1	320-2	320-3	320-4	320-5
被除数初期値	5	7	3	6	8	12
4ビット被除数(ビット行)	304-0	304-1	304-2	304-3	304-4	304-5
	1	1	1	0	0	0
	1	1	1	1	0	0
	1	1	0	1	0	1
	0	0	0	0	1	1
除数初期値	322-0	322-1	322-2	322-3	322-4	322-5
4ビット除数(ビット行)	304-4	304-5	304-6	304-7	304-8	304-9
	0	0	1	1	1	0
	0	1	0	1	0	1
	1	0	0	0	1	1
	0	0	0	0	0	0
商の初期値	324-0	324-1	324-2	324-3	324-4	324-5
4ビット商(ビット行)	304-8	304-9	304-10	304-11	304-12	304-13
	0	0	0	0	0	0
	0	0	0	0	0	0
	0	0	0	0	0	0
	0	0	0	0	0	0
剰余初期値	326-0	326-1	326-2	326-3	326-4	326-5
4ビット剰余(ビット行)	304-12	304-13	304-14	304-15	304-16	304-17
	0	0	0	0	0	0
	0	0	0	0	0	0
	0	0	0	0	0	0
	0	0	0	0	0	0
マスク行	305-0	305-1	305-2	305-3	305-4	305-5
	0	0	0	0	0	0

【 図 3 B 】

330 → メモリアレイ

被除数初期値	320-0	320-1	320-2	320-3	320-4	320-5
4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値	322-0	322-1	322-2	322-3	322-4	322-5
4ビット除数(ビット行)	4	2	1	3	5	6
304-4	0	0	1	1	1	0
304-5	0	1	0	1	0	1
304-6	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の値	324-0	324-1	324-2	324-3	324-4	324-5
4ビット商(ビット行)	0	0	0	0	0	0
304-8	0	0	0	0	0	0
304-9	0	0	0	0	0	0
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余値	326-0	326-1	326-2	326-3	326-4	326-5
4ビット剰余(ビット行)	0	0	0	0	1	1
304-12	0	0	0	0	1	1
304-13	0	0	0	0	0	0
304-14	0	0	0	0	0	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	0	0	0	0	0	0
305-0	0	0	0	0	0	0
305-1	0	0	0	0	0	0
305-2	0	0	0	0	0	0
305-3	0	0	0	0	0	0
305-4	0	0	0	0	0	0
305-5	0	0	0	0	0	0

【 図 3 C 】

330 → メモリアレイ

被除数初期値	320-0	320-1	320-2	320-3	320-4	320-5
4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値	322-0	322-1	322-2	322-3	322-4	322-5
4ビット除数(ビット行)	4	2	1	3	5	6
304-4	0	0	1	1	1	0
304-5	0	1	0	1	0	1
304-6	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の値	324-0	324-1	324-2	324-3	324-4	324-5
4ビット商(ビット行)	0	0	0	0	0	0
304-8	0	0	0	0	0	0
304-9	0	0	0	0	0	0
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余値	326-0	326-1	326-2	326-3	326-4	326-5
4ビット剰余(ビット行)	0	0	0	0	1	1
304-12	0	0	0	0	1	1
304-13	0	0	0	0	0	0
304-14	0	0	0	0	0	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	0	0	0	0	0	0
305-0	0	0	0	0	0	0
305-1	0	0	0	0	0	0
305-2	0	0	0	0	0	0
305-3	0	0	0	0	0	0
305-4	0	0	0	0	0	0
305-5	0	0	0	0	0	0

【 図 3 D 】

330 → メモリアレイ

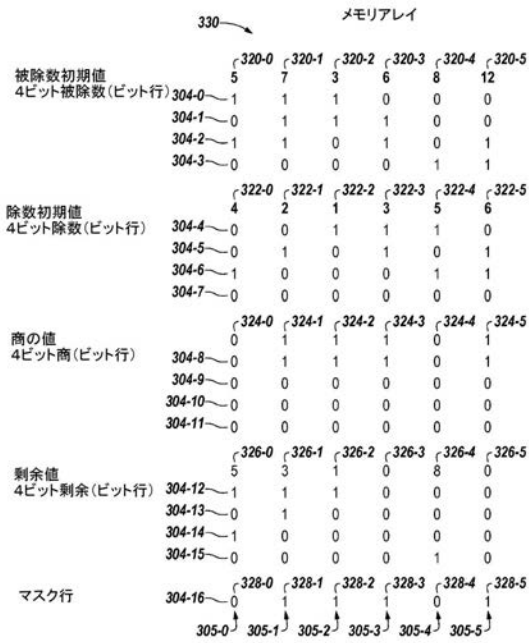
被除数初期値	320-0	320-1	320-2	320-3	320-4	320-5
4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値	322-0	322-1	322-2	322-3	322-4	322-5
4ビット除数(ビット行)	4	2	1	3	5	6
304-4	0	0	1	1	1	0
304-5	0	1	0	1	0	1
304-6	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の値	324-0	324-1	324-2	324-3	324-4	324-5
4ビット商(ビット行)	0	0	0	0	0	0
304-8	0	0	0	0	0	0
304-9	0	0	0	0	0	0
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余値	326-0	326-1	326-2	326-3	326-4	326-5
4ビット剰余(ビット行)	0	0	0	0	1	1
304-12	0	0	0	0	1	1
304-13	0	0	0	0	0	0
304-14	0	0	0	0	0	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	0	0	0	0	0	0
305-0	0	0	0	0	0	0
305-1	0	0	0	0	0	0
305-2	0	0	0	0	0	0
305-3	0	0	0	0	0	0
305-4	0	0	0	0	0	0
305-5	0	0	0	0	0	0

【 図 3 E 】

330 → メモリアレイ

被除数初期値	320-0	320-1	320-2	320-3	320-4	320-5
4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値	322-0	322-1	322-2	322-3	322-4	322-5
4ビット除数(ビット行)	4	2	1	3	5	6
304-4	0	0	1	1	1	0
304-5	0	1	0	1	0	1
304-6	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の値	324-0	324-1	324-2	324-3	324-4	324-5
4ビット商(ビット行)	0	0	0	0	0	0
304-8	0	0	0	0	0	0
304-9	0	0	0	0	0	0
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余値	326-0	326-1	326-2	326-3	326-4	326-5
4ビット剰余(ビット行)	0	0	0	0	2	2
304-12	0	0	0	0	0	0
304-13	0	0	0	0	1	1
304-14	0	0	0	0	0	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	0	0	0	0	0	0
305-0	0	0	0	0	0	0
305-1	0	0	0	0	0	0
305-2	0	0	0	0	0	0
305-3	0	0	0	0	0	0
305-4	0	0	0	0	0	0
305-5	0	0	0	0	0	0

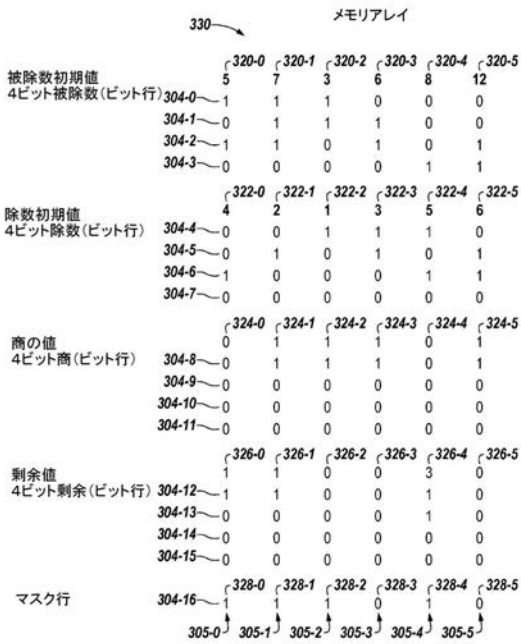
【 図 3 F 】



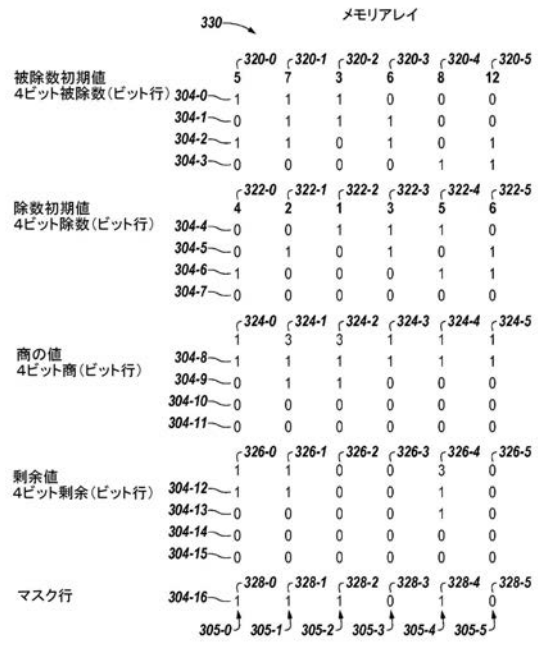
【 図 3 G 】



【 図 3 H 】



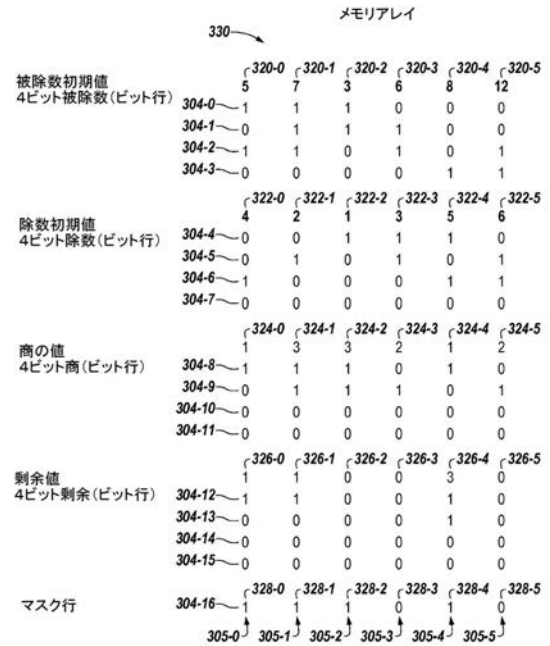
【 図 3 I 】



【 図 3 J 】



【 図 3 K 】



【 手続 補正書 】

【 提出日 】 平成28年6月14日 (2016.6.14)

【 手続 補正 1 】

【 補正対象書類名 】 特許請求の範囲

【 補正対象項目名 】 全文

【 補正方法 】 変更

【 補正の内容 】

【 特許請求の範囲 】

【 請求項 1 】

除算演算の実行方法であって、

メモリアレイのセンス線に接続されたN個のメモリセルの第1部分に記憶された被除数値の論理表現、及び前記メモリアレイの前記センス線に接続された前記N個のメモリセルの第2部分に記憶された除数値の論理表現を用いて除算演算を実行すること、

前記除算演算の結果の商の値の論理表現を、前記メモリアレイの前記センス線に接続された前記N個のメモリセルの第3部分に記憶させること、を含み、

前記除算演算を実行することは、

前記N個のメモリセルの前記第1部分のN番目のメモリセルに記憶されたビットを前記N個のメモリセルの第4部分の第1メモリセルにコピーすることであって、前記N個のメモリセルの前記第4部分は、前記除算演算の前記結果の剰余値の論理表現を記憶すること、並びに

前記除数値を前記剰余値と比較すること、を含む方法。

【 請求項 2 】

前記被除数値の前記論理表現を、前記センス線に接続された前記N個のメモリセルの前記第1部分にビットベクトルとして記憶させることをさらに含み、前記N個のメモリセルの前記第1部分の各メモリセルは、前記ビットベクトルの各ビットを記憶する、請求項1

に記載の方法。

【請求項 3】

前記除数値の前記論理表現を、前記センス線に接続された前記 N 個のメモリセルの前記第 2 部分にビットベクトルとして記憶させることをさらに含み、前記 N 個のメモリセルの前記第 2 部分の各メモリセルは、前記ビットベクトルの各ビットを記憶する、請求項 1 または請求項 2 に記載の方法。

【請求項 4】

前記商の値の前記論理表現を記憶させることは、前記商の値の前記論理表現を、前記センス線に接続された前記 N 個のメモリセルの前記第 3 部分にビットベクトルとして記憶させることを含み、前記 N 個のメモリセルの前記第 3 部分の各メモリセルは、前記ビットベクトルの各ビットを記憶する、請求項 1 から請求項 3 のいずれか一項に記載の方法。

【請求項 5】

前記除数値が前記剰余値以下である場合、

前記除数値を前記剰余値から引き、得られた値の論理表現を前記剰余値の前記論理表現として前記 N 個のメモリセルの前記第 4 部分に記憶させること、

前記商の値の前記論理表現をシフトし、インクリメントして、前記商の値を更新すること、及び

前記剰余値の前記論理表現をシフトして前記剰余値を更新することをさらに含む、請求項 1 から請求項 4 のいずれか一項に記載の方法。

【請求項 6】

前記除数値が前記剰余値よりも大きい場合、

前記商の値の前記論理表現をシフトして前記商の値を更新すること、及び

前記剰余値の前記論理表現をシフトして前記剰余値を更新することをさらに含む、請求項 5 に記載の方法。

【請求項 7】

前記 N 個のメモリセルの前記第 1 部分の N - 1 番目のメモリセルに記憶されたビットを前記 N 個のメモリセルの前記第 4 部分の前記第 1 メモリセルにコピーすること、及び

前記除数値を前記剰余値と比較することをさらに含む、請求項 6 に記載の方法。

【請求項 8】

前記比較に応じたマスクを作成することをさらに含み、前記マスクは、前記除数値が前記剰余値以下であるかどうか、または前記除数値が前記剰余値よりも大きいかどうかを示す、請求項 6 に記載の方法。

【請求項 9】

センス線及び第 1 の本数の選択線に接続された第 1 の数のメモリセルを含むメモリアレイの第 1 アドレス空間であって、被除数値を記憶するように構成されている第 1 アドレス空間と、

前記センス線及び第 2 の本数の選択線に接続された第 2 の数のメモリセルを含む前記メモリアレイの第 2 アドレス空間であって、除数値を記憶するように構成されている第 2 アドレス空間と、

前記センス線及び第 3 の本数の選択線に接続された第 3 の数のメモリセルを含む前記メモリアレイの第 3 アドレス空間であって、剰余値を記憶するように構成されている第 3 アドレス空間と、

前記メモリアレイに接続された制御部であって、

前記被除数値を前記除数値で割る検出回路、並びに

前記剰余値を前記第 3 アドレス空間に記憶させること

を制御するように構成された制御部と、を備え、

前記検出回路は、前記メモリアレイの前記メモリセルと間隔を置いて形成されたトランジスタを備える、装置。

【請求項 10】

前記センス線に接続された前記第 1 の数のメモリセルは、前記センス線に接続された N

個のメモリセルで構成されており、前記被除数値を前記除数値で割るように制御可能な前記検出回路は、

前記第 1 の数のメモリセルの N 番目のメモリセルに記憶されたビットを前記第 3 の数のメモリセルの第 1 メモリセルにコピーし、

前記除数値を前記剰余値と比較するように制御可能な前記検出回路からなる、請求項 9 に記載の装置。

【請求項 1 1】

前記剰余値を前記第 3 アドレス空間に記憶させることを制御するように構成されている前記制御部は、

前記除数値が前記剰余値以下である場合、

前記除数値を前記剰余値から引き、得られた値の論理表現を前記第 3 の数のメモリセルに記憶させ、

前記得られた値の前記論理表現をシフトして前記剰余値を更新する前記検出回路を制御するように構成されている前記制御部からなり、

前記被除数値を前記除数値で割るように制御可能な前記検出回路はさらに、前記メモリアレイの第 4 アドレス空間に記憶された商の値の論理表現をシフトし、インクリメントするように制御可能な前記検出回路からなる、請求項 10 に記載の装置。

【請求項 1 2】

前記検出回路は、センス線アドレスアクセスを介してデータを転送することなく、前記被除数値を前記除数値で割るように制御可能である、請求項 9 から請求項 1 1 のいずれか一項に記載の装置。

【請求項 1 3】

前記検出回路は、前記第 1 の数のメモリセル、前記第 2 の数のメモリセル及び前記第 3 の数のメモリセルと間隔を置いて設けられたトランジスタを含む計算素子を備える、請求項 9 から請求項 1 2 のいずれか一項に記載の装置。

【請求項 1 4】

複数の除算演算の実行方法であって、

メモリアレイの複数のセンス線に接続されたメモリセルの複数の第 1 部分に記憶された複数の被除数値を、前記メモリアレイの前記複数のセンス線に接続された前記メモリセルの複数の第 2 部分に記憶された複数の除数値で割ることであって、

前記メモリセルの前記複数の第 1 部分のそれぞれが、前記複数のセンス線の異なるセンス線に接続されており、前記複数の被除数値のそれぞれが、前記複数の第 1 部分の異なる第 1 部分に記憶され、

前記メモリセルの前記複数の第 2 部分のそれぞれが、前記複数のセンス線の異なるセンス線に接続されており、前記複数の除数値のそれぞれが、前記複数の第 2 部分の異なる第 2 部分に記憶されること、

前記複数の被除数値を前記複数の除数値で割った結果の複数の商の値を、前記メモリアレイの前記複数のセンス線に接続された前記メモリセルの複数の第 3 部分に記憶させること、並びに

前記複数の被除数値を前記複数の除数値で割った前記結果の複数の剰余値を、前記メモリアレイの前記複数のセンス線に接続された前記メモリセルの複数の第 4 部分に記憶させることを含み、

前記複数の除算演算の実行に用いられる複数の計算は、前記複数の除算演算のいずれか 1 つの実行に用いられる複数の計算と同一であり、

前記複数の除算演算のそれぞれは、前記複数の被除数値の各被除数値を前記複数の除数値の各除数値で割ることを含み、前記複数の除算演算の各除算演算の前記被除数値及び前記除数値は、前記複数のセンス線のうちの同一のセンス線に接続されたメモリセルに記憶され、

複数のセンス線に接続されたメモリセルの前記複数の第 1 部分のそれぞれは、前記複数のセンス線のうちの 1 本に接続された N 個のメモリセルで構成されており、

前記複数の被除数値を前記複数の除数値で割ることは、前記複数のメモリセルの前記第1部分のN番目のメモリセルに記憶された各ビットを前記複数のメモリセルの前記第4部分の第1メモリセルのそれぞれにコピーすることを含み、

前記複数のメモリセルの前記第1部分の前記N番目のメモリセルに記憶された各ビットをコピーすることに伴う複数の計算は、前記複数の被除数値、前記複数の除数値、及び前記複数の剰余値の大きさによらず変わらない、方法。

【請求項15】

センス線及び第1の本数の選択線に接続された第1の数のメモリセルを含むメモリアレイの第1アドレス空間であって、被除数値を記憶するように構成されている第1アドレス空間と、

前記センス線及び第2の本数の選択線に接続された第2の数のメモリセルを含む前記メモリアレイの第2アドレス空間であって、除数値を記憶するように構成されている第2アドレス空間と、

前記センス線及び第3の本数の選択線に接続された第3の数のメモリセルを含む前記メモリアレイの第3アドレス空間であって、剰余値を記憶するように構成されている第3アドレス空間と、

前記メモリアレイに接続された制御部であって、

前記被除数値を前記除数値で割る検出回路、並びに

前記剰余値を前記第3アドレス空間に記憶させること

を制御するように構成された制御部と、を備え、

前記センス線に接続された前記第1の数のメモリセルは、前記センス線に接続されたN個のメモリセルで構成されており、

前記被除数値を前記除数値で割るように制御可能な前記検出回路は、

前記第1の数のメモリセルのN番目のメモリセルに記憶されたビットを前記第3の数のメモリセルの第1メモリセルにコピーし、

前記除数値を前記剰余値と比較するように制御可能な前記検出回路からなる、装置。

【請求項16】

センス線及び第1の本数の選択線に接続された第1の数のメモリセルを含むメモリアレイの第1アドレス空間であって、被除数値を記憶するように構成されている第1アドレス空間と、

前記センス線及び第2の本数の選択線に接続された第2の数のメモリセルを含む前記メモリアレイの第2アドレス空間であって、除数値を記憶するように構成されている第2アドレス空間と、

前記センス線及び第3の本数の選択線に接続された第3の数のメモリセルを含む前記メモリアレイの第3アドレス空間であって、剰余値を記憶するように構成されている第3アドレス空間と、

前記メモリアレイに接続された制御部であって、

前記被除数値を前記除数値で割るように構成された検出回路、並びに

前記剰余値を前記第3アドレス空間に記憶させること

を制御するように構成された制御部と、を備え、

前記検出回路は、センス線アドレスアクセスを介してデータを転送することなく、前記被除数値を前記除数値で割るように構成されている、装置。

【請求項17】

センス線及び第1の本数の選択線に接続された第1の数のメモリセルを含むメモリアレイの第1アドレス空間であって、被除数値を記憶するように構成されている第1アドレス空間と、

前記センス線及び第2の本数の選択線に接続された第2の数のメモリセルを含む前記メモリアレイの第2アドレス空間であって、除数値を記憶するように構成されている第2アドレス空間と、

前記センス線及び第3の本数の選択線に接続された第3の数のメモリセルを含む前記メ

モリアレイの第3アドレス空間であって、剰余値を記憶するように構成されている第3アドレス空間と、

前記メモリアレイに接続された制御部であって、

前記被除数値を前記除数値で割るように構成された検出回路、並びに

前記剰余値を前記第3アドレス空間に記憶させること

を制御するように構成された制御部と、を備え、

前記検出回路は、前記第1の数のメモリセル、前記第2の数のメモリセル及び前記第3の数のメモリセルと間隔を置いて設けられたトランジスタを含む計算素子を備える、装置。

【手続補正2】

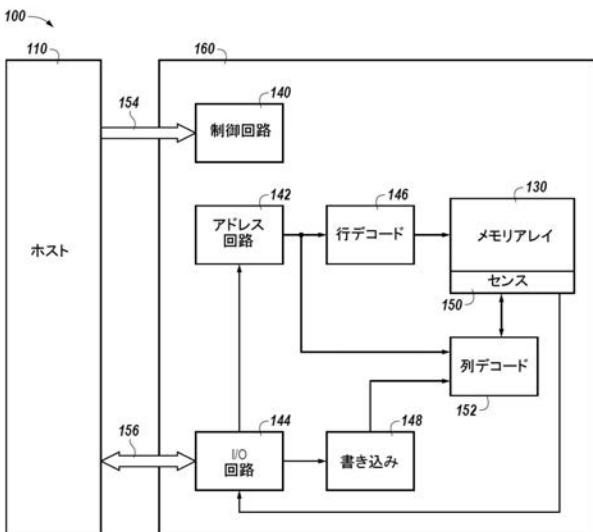
【補正対象書類名】図面

【補正対象項目名】全図

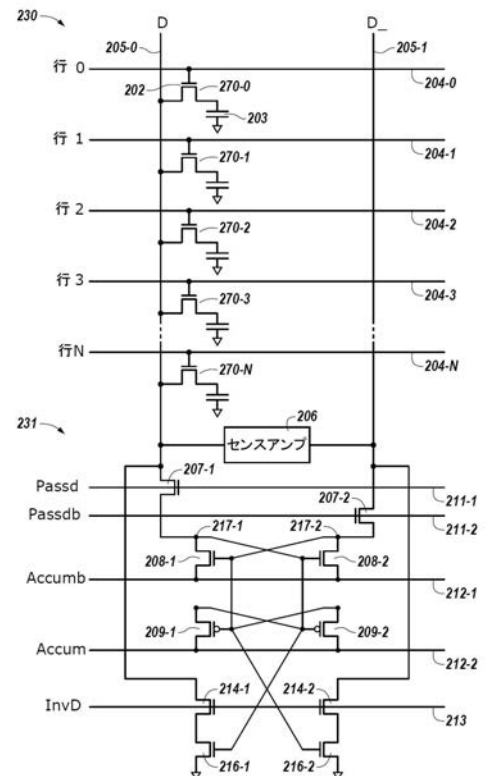
【補正方法】変更

【補正の内容】

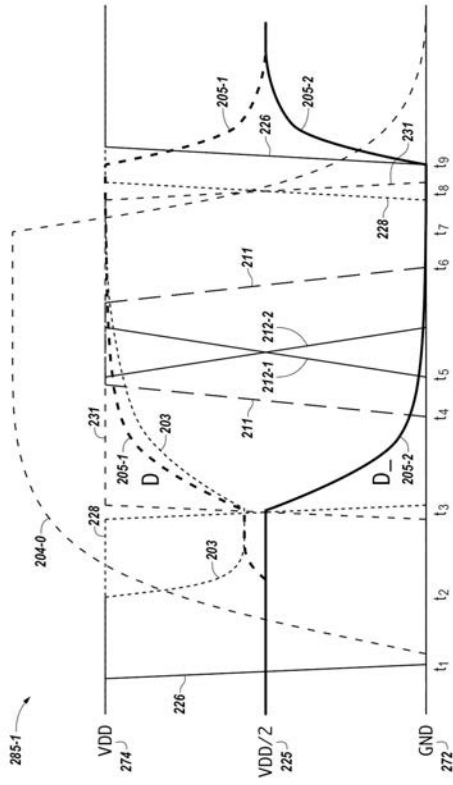
【図1】



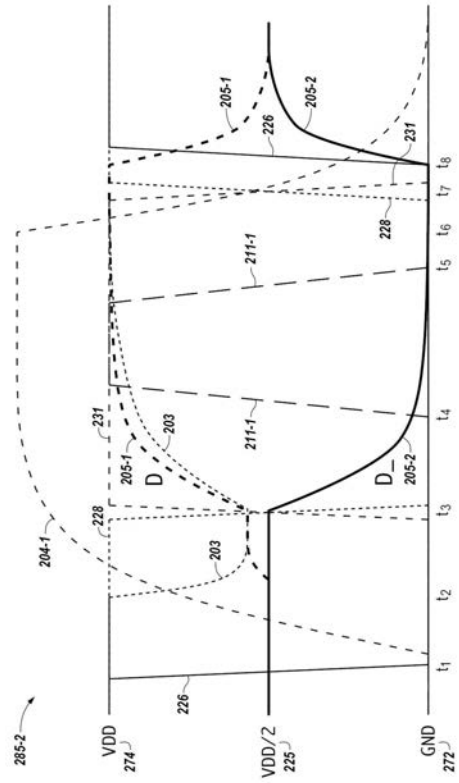
【図2A】



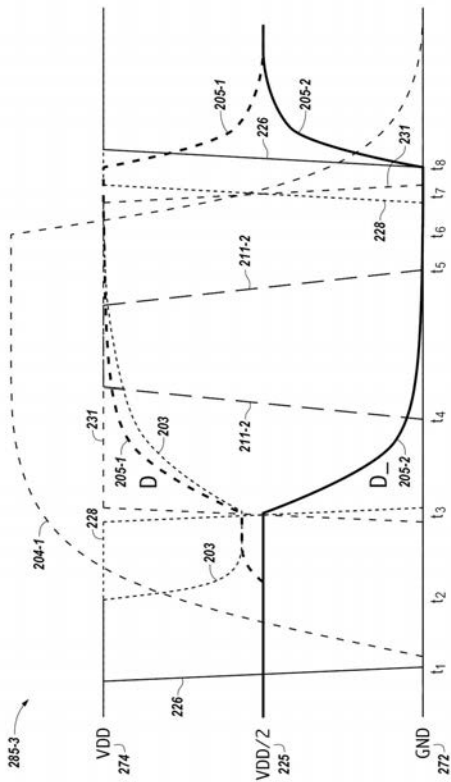
【 図 2 B 】



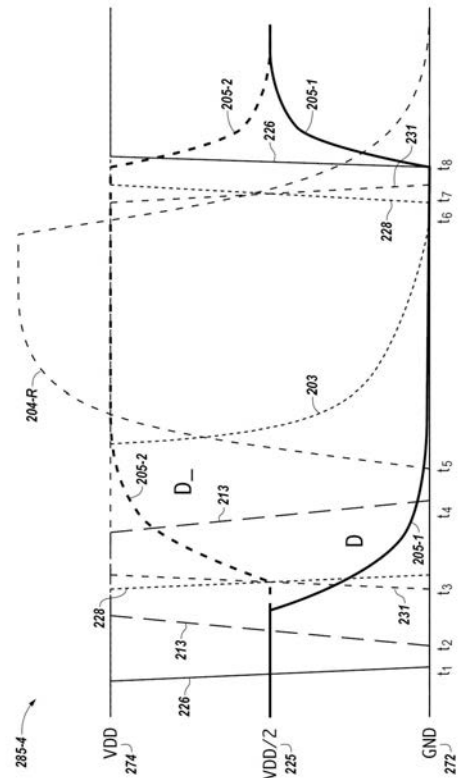
【 図 2 C - 1 】



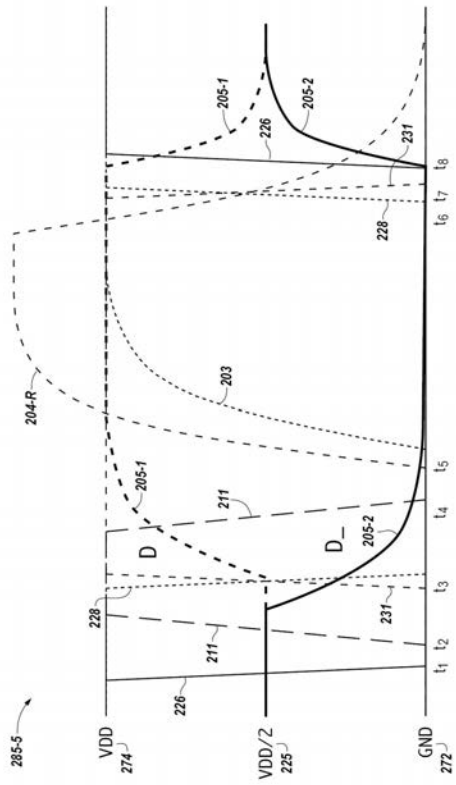
【 図 2 C - 2 】



【 図 2 D - 1 】



【 図 2 D - 2 】



【 図 3 A 】

330

メモリアレイ

被除数初期値	320-0	320-1	320-2	320-3	320-4	320-5
4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値	322-0	322-1	322-2	322-3	322-4	322-5
4ビット除数(ビット行)	4	2	1	3	5	6
304-4	0	0	1	1	1	0
304-5	0	1	0	1	0	1
304-6	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の初期値	324-0	324-1	324-2	324-3	324-4	324-5
4ビット商(ビット行)	0	0	0	0	0	0
304-8	0	0	0	0	0	0
304-9	0	0	0	0	0	0
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余初期値	326-0	326-1	326-2	326-3	326-4	326-5
4ビット剰余(ビット行)	0	0	0	0	0	0
304-12	0	0	0	0	0	0
304-13	0	0	0	0	0	0
304-14	0	0	0	0	0	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	0	0	0	0	0	0
305-0	305-1	305-2	305-3	305-4	305-5	

【 図 3 B 】

330

メモリアレイ

被除数初期値	320-0	320-1	320-2	320-3	320-4	320-5
4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値	322-0	322-1	322-2	322-3	322-4	322-5
4ビット除数(ビット行)	4	2	1	3	5	6
304-4	0	0	1	1	1	0
304-5	0	1	0	1	0	1
304-6	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の値	324-0	324-1	324-2	324-3	324-4	324-5
4ビット商(ビット行)	0	0	0	0	0	0
304-8	0	0	0	0	0	0
304-9	0	0	0	0	0	0
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余値	326-0	326-1	326-2	326-3	326-4	326-5
4ビット剰余(ビット行)	0	0	0	1	1	
304-12	0	0	0	1	1	
304-13	0	0	0	0	0	0
304-14	0	0	0	0	0	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	0	0	0	0	0	0
305-0	305-1	305-2	305-3	305-4	305-5	

【 図 3 C 】

330

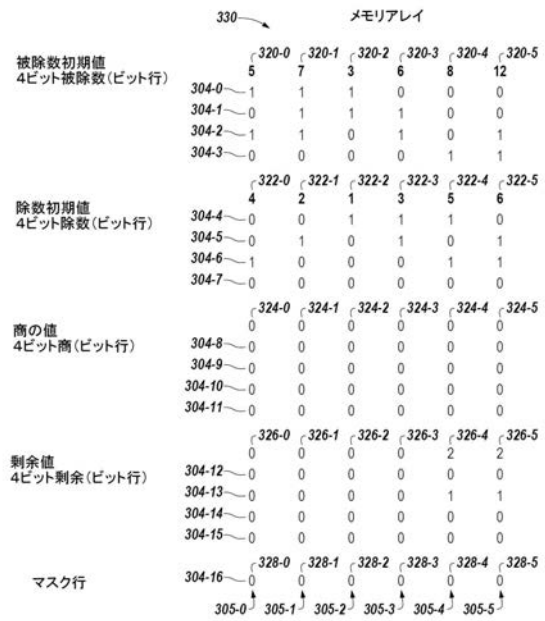
メモリアレイ

被除数初期値	320-0	320-1	320-2	320-3	320-4	320-5
4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値	322-0	322-1	322-2	322-3	322-4	322-5
4ビット除数(ビット行)	4	2	1	3	5	6
304-4	0	0	1	1	1	0
304-5	0	1	0	1	0	1
304-6	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の値	324-0	324-1	324-2	324-3	324-4	324-5
4ビット商(ビット行)	0	0	0	0	0	0
304-8	0	0	0	0	0	0
304-9	0	0	0	0	0	0
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余値	326-0	326-1	326-2	326-3	326-4	326-5
4ビット剰余(ビット行)	0	0	0	1	1	
304-12	0	0	0	1	1	
304-13	0	0	0	0	0	0
304-14	0	0	0	0	0	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	0	0	0	0	0	0
305-0	305-1	305-2	305-3	305-4	305-5	

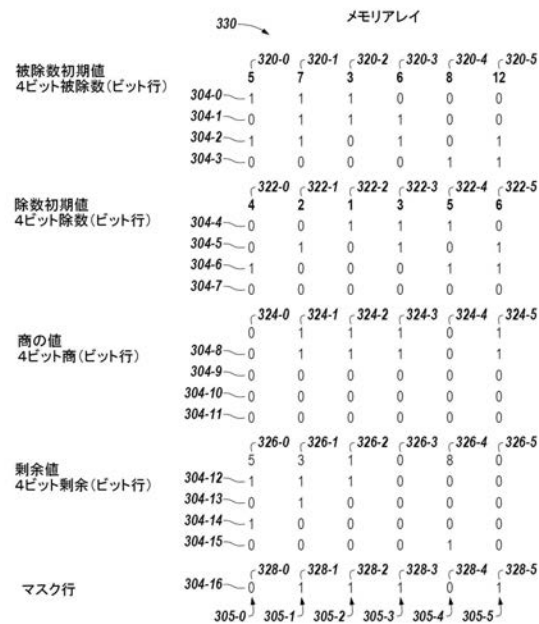
【 図 3 D 】



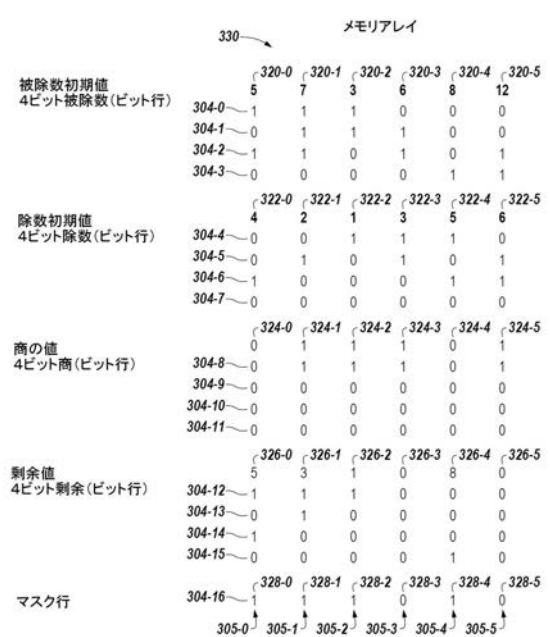
【 図 3 E 】



【 図 3 F 】



【 図 3 G 】



【 図 3 H 】

330 → メモリアレイ

	320-0	320-1	320-2	320-3	320-4	320-5
被除数初期値 4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値 4ビット除数(ビット行)	322-0	322-1	322-2	322-3	322-4	322-5
304-4	4	2	1	3	5	6
304-5	0	0	1	1	1	0
304-6	0	1	0	1	0	1
304-7	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の値 4ビット商(ビット行)	324-0	324-1	324-2	324-3	324-4	324-5
304-8	0	1	1	1	0	1
304-9	0	1	1	1	0	1
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余値 4ビット剰余(ビット行)	326-0	326-1	326-2	326-3	326-4	326-5
304-12	1	1	0	0	3	0
304-13	1	1	0	0	1	0
304-14	0	0	0	0	1	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	1	1	1	0	1	0
305-0	1	1	1	0	1	0
305-1	1	1	1	0	1	0
305-2	1	1	1	0	1	0
305-3	1	1	1	0	1	0
305-4	1	1	1	0	1	0
305-5	1	1	1	0	1	0

【 図 3 I 】

330 → メモリアレイ

	320-0	320-1	320-2	320-3	320-4	320-5
被除数初期値 4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値 4ビット除数(ビット行)	322-0	322-1	322-2	322-3	322-4	322-5
304-4	4	2	1	3	5	6
304-5	0	0	1	1	1	0
304-6	0	1	0	1	0	1
304-7	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の値 4ビット商(ビット行)	324-0	324-1	324-2	324-3	324-4	324-5
304-8	1	3	3	1	1	1
304-9	0	1	1	1	1	1
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余値 4ビット剰余(ビット行)	326-0	326-1	326-2	326-3	326-4	326-5
304-12	1	1	0	0	3	0
304-13	0	0	0	0	1	0
304-14	0	0	0	0	0	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	1	1	1	0	1	0
305-0	1	1	1	0	1	0
305-1	1	1	1	0	1	0
305-2	1	1	1	0	1	0
305-3	1	1	1	0	1	0
305-4	1	1	1	0	1	0
305-5	1	1	1	0	1	0

【 図 3 J 】

330 → メモリアレイ

	320-0	320-1	320-2	320-3	320-4	320-5
被除数初期値 4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値 4ビット除数(ビット行)	322-0	322-1	322-2	322-3	322-4	322-5
304-4	4	2	1	3	5	6
304-5	0	0	1	1	1	0
304-6	0	1	0	1	0	1
304-7	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の値 4ビット商(ビット行)	324-0	324-1	324-2	324-3	324-4	324-5
304-8	1	3	3	2	1	2
304-9	0	1	1	1	0	1
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余値 4ビット剰余(ビット行)	326-0	326-1	326-2	326-3	326-4	326-5
304-12	1	1	0	0	3	0
304-13	0	0	0	0	1	0
304-14	0	0	0	0	0	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	1	1	1	0	1	0
305-0	1	1	1	0	1	0
305-1	1	1	1	0	1	0
305-2	1	1	1	0	1	0
305-3	1	1	1	0	1	0
305-4	1	1	1	0	1	0
305-5	1	1	1	0	1	0

【 図 3 K 】

330 → メモリアレイ

	320-0	320-1	320-2	320-3	320-4	320-5
被除数初期値 4ビット被除数(ビット行)	5	7	3	6	8	12
304-0	1	1	1	0	0	0
304-1	0	1	1	1	0	0
304-2	1	1	0	1	0	1
304-3	0	0	0	0	1	1
除数初期値 4ビット除数(ビット行)	322-0	322-1	322-2	322-3	322-4	322-5
304-4	4	2	1	3	5	6
304-5	0	0	1	1	1	0
304-6	0	1	0	1	0	1
304-7	1	0	0	0	1	1
304-7	0	0	0	0	0	0
商の値 4ビット商(ビット行)	324-0	324-1	324-2	324-3	324-4	324-5
304-8	1	3	3	2	1	2
304-9	0	1	1	1	0	1
304-10	0	0	0	0	0	0
304-11	0	0	0	0	0	0
剰余値 4ビット剰余(ビット行)	326-0	326-1	326-2	326-3	326-4	326-5
304-12	1	1	0	0	3	0
304-13	0	0	0	0	1	0
304-14	0	0	0	0	0	0
304-15	0	0	0	0	0	0
マスク行	328-0	328-1	328-2	328-3	328-4	328-5
304-16	1	1	1	0	1	0
305-0	1	1	1	0	1	0
305-1	1	1	1	0	1	0
305-2	1	1	1	0	1	0
305-3	1	1	1	0	1	0
305-4	1	1	1	0	1	0
305-5	1	1	1	0	1	0

【 国際調査報告 】

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2014/063831

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. Claims Nos.: 10-11
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:



1. As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. As all searchable claims could be searched without effort justifying an additional fees, this Authority did not invite payment of any additional fees.
3. As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

- Remark on Protest**
- The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2014/063831

A. CLASSIFICATION OF SUBJECT MATTER G11C 8/06(2006.01)i, G11C 7/10(2006.01)i		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols) G11C 8/06; G11C 15/00; G06F 15/00; G11C 5/06; G11C 7/00; H03K 19/003; G11C 5/02; G11C 7/10		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched Korean utility models and applications for utility models Japanese utility models and applications for utility models		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) eKOMPASS(KIPO internal) & Keywords: memory, cell, array, sense, circuitry, dividend, divisor, remainder, quotient, logical, operation		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2010-0308858 A1 (HIDEYUKI NODA et al.) 09 December 2010 See paragraphs [0208], [0211], [0450], [0453]-[0454], [0461]-[462], [0466], [0562]; and figures 4, 46-47, 69.	1-5, 12, 22
A		6-9, 13-21, 23-31
A	US 2007-0180006 A1 (TAKAYUKI GYOTEN et al.) 02 August 2007 See paragraph [0055]; claim 1; and figures 2.	1-9, 12-31
A	US 06166942 A (HUY T. VO et al.) 26 December 2000 See column 9, lines 1-40; and figure 5.	1-9, 12-31
A	US 05638317 A (HIEP V. TRAN) 10 June 1997 See column 1, lines 16-18, 24-26; and figure 1.	1-9, 12-31
A	US 2008-0137388 A1 (RENGARAJAN S. KRISHNAN et al.) 12 June 2008 See paragraphs [0021]-[0022]; and figure 1A.	1-9, 12-31
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.		
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family		
Date of the actual completion of the international search 26 January 2015 (26.01.2015)		Date of mailing of the international search report 26 January 2015 (26.01.2015)
Name and mailing address of the ISA/KR  International Application Division Korean Intellectual Property Office 189 Cheongsa-ro, Seo-gu, Daejeon Metropolitan City, 302-701, Republic of Korea Facsimile No. +82 42 472 3473		Authorized officer BYUN, Sung Cheal  Telephone No. +82-42-481-8262

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2014/063831

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2010-0308858 A1	09/12/2010	JP 2006-127460 A US 2005-0285862 A1 US 2009-0027978 A1 US 7562198 B2 US 7791962 B2 US 8089819 B2	18/05/2006 29/12/2005 29/01/2009 14/07/2009 07/09/2010 03/01/2012
US 2007-0180006 A1	02/08/2007	JP 04989900 B2 JP 2007-206849 A US 7505352 B2	01/08/2012 16/08/2007 17/03/2009
US 06166942 A	26/12/2000	JP 2004-500673 A KR 10-0676864 B1 TW 507213 A US 6141286 A WO 2000-11676 A1	08/01/2004 05/02/2007 21/10/2002 31/10/2000 02/03/2000
US 05638317 A	10/06/1997	JP 03361825 B2 JP 06089575 A	07/01/2003 29/03/1994
US 2008-0137388 A1	12/06/2008	US 7471536 B2	30/12/2008

フロントページの続き

(81)指定国 AP(BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), EA(AM, AZ, BY, KG, KZ, RU, TJ, TM), EP(AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OA(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG), AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US

(特許庁注：以下のものは登録商標)

1 . R R A M

Fターム(参考) 5M024 AA90 CC70 KK29 KK35 PP01 PP03 PP07