

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7651292号
(P7651292)

(45)発行日 令和7年3月26日(2025.3.26)

(24)登録日 令和7年3月17日(2025.3.17)

(51)国際特許分類		F I		
G 0 6 F	13/16 (2006.01)	G 0 6 F	13/16	5 1 0 A
G 0 6 F	12/00 (2006.01)	G 0 6 F	12/00	5 9 7 D
G 0 6 F	11/10 (2006.01)	G 0 6 F	11/10	6 5 2
G 1 1 C	29/42 (2006.01)	G 1 1 C	29/42	
請求項の数 20 外国語出願 (全39頁)				
(21)出願番号	特願2020-194691(P2020-194691)	(73)特許権者	591003943	
(22)出願日	令和2年11月24日(2020.11.24)		インテル・コーポレーション	
(65)公開番号	特開2021-179962(P2021-179962 A)		アメリカ合衆国 9 5 0 5 4 カリフォルニア州・サンタクララ・ミッション カレッジ ブレーバード・2 2 0 0	
(43)公開日	令和3年11月18日(2021.11.18)	(74)代理人	110000877	
審査請求日	令和5年9月1日(2023.9.1)		弁理士法人 R Y U K A 国際特許事務所	
(31)優先権主張番号	16/875,642	(72)発明者	クルジット エス・ ベインズ	
(32)優先日	令和2年5月15日(2020.5.15)		アメリカ合衆国 9 5 0 5 4 カリフォルニア州・サンタクララ・ミッション カレッジ ブレーバード・2 2 0 0 インテル・コーポレーション内	
(33)優先権主張国・地域又は機関	米国(US)	(72)発明者	ラジャット アガールウォル	
			アメリカ合衆国 9 5 0 5 4 カリフォルニア州・サンタクララ・ミッション カ	
			最終頁に続く	

(54)【発明の名称】 オンダイ E C C を選択的に無効化するための読み出しリトライ

(57)【特許請求の範囲】

【請求項 1】

メモリデバイスに結合されているハードウェアインタフェースと、
前記メモリデバイスのためのリトライコマンドを生成するコマンドロジックと、
を備え、
前記コマンドロジックは、前の読み出しコマンドに応答して前記メモリデバイスから送信される、検出された訂正不可能エラー（DUE）の指標の受信に応答して、前記リトライコマンドを生成し、

前記リトライコマンドは、前記メモリデバイスの E C C 回路によるエラー検査訂正（E C C）を適用せずに読み出しデータを返すように前記メモリデバイスをトリガする、メモリコントローラ。

【請求項 2】

前記前の読み出しコマンドに応答して受信された、前記 E C C が適用された、エイリアスされた訂正済みの読み出しデータを格納する第 1 のレジスタと、
前記リトライコマンドに応答して受信された、前記 E C C が適用されていない、訂正されていない読み出しデータを格納する第 2 のレジスタと、
前記第 1 のレジスタ内の前記エイリアスされた訂正済みの読み出しデータと、前記第 2 のレジスタ内の前記訂正されていない読み出しデータとを比較し、比較結果に基づいて前記 DUE を訂正可能エラー（CE）に変換する、比較器とをさらに備える、請求項 1 に記載のメモリコントローラ。

【請求項 3】

前記コマンドロジックは、前記リトライコマンドを示すためのコマンドエンコーディングを有するコマンドを生成する、請求項 1 に記載のメモリコントローラ。

【請求項 4】

前記コマンドロジックは、前記リトライコマンドを示すために、同じアドレスに一連の連続する読み出しコマンドを生成する、請求項 1 に記載のメモリコントローラ。

【請求項 5】

前記コマンドロジックは、前記メモリデバイスのモードレジスタをリトライコマンドモードに設定するために、モードレジスタ書き込みコマンドを生成し、

前記コマンドロジックは、前記メモリデバイスが前記リトライコマンドモードである間に、前記リトライコマンドを示すために送信する読み出しコマンドを生成する、請求項 1 に記載のメモリコントローラ。

【請求項 6】

前記コマンドロジックは、前記前の読み出しコマンドに応答して、前記メモリデバイスから、前記 DUE と共に送信されるエイリアスされたエラーデータを受信する、請求項 1 ～ 5 のいずれか一項に記載のメモリコントローラ。

【請求項 7】

前記メモリデバイスは、ダブルデータレート（DDR）規格と互換性のあるシンクロナスダイナミックランダムアクセスメモリ（SDRAM）デバイスを備える、請求項 1 ～ 6 のいずれか一項に記載のメモリコントローラ。

【請求項 8】

メモリデバイスであって、
読み出しコマンドに応答して、読み出しデータを受信し、前記メモリデバイスの内部で前記読み出しデータに ECC を適用し、訂正済みデータを返すエラー検査訂正（ECC）回路と、

前記読み出しコマンドの後にリトライコマンドを受信するハードウェアインタフェースであって、前記ハードウェアインタフェースは、前記リトライコマンドに応答して、前記 ECC 回路による ECC を適用せずに前記読み出しデータを返す、ハードウェアインタフェースと、

を備え、

前記 ECC 回路は、前記リトライコマンドを送信するようにホストをトリガするために、検出された訂正不可能エラー（DUE）を前記ホストに示す、メモリデバイス。

【請求項 9】

前記リトライコマンドは、前記リトライコマンドを示すためのコマンドエンコーディングを有するコマンドを含む、請求項 8 に記載のメモリデバイス。

【請求項 10】

前記リトライコマンドは、他の介在コマンドを伴わない一連の読み出しコマンドを含む、請求項 8 に記載のメモリデバイス。

【請求項 11】

前記読み出しコマンドのアドレスを記憶するレジスタをさらに備え、前記メモリデバイスは、前記レジスタに記憶されている前記読み出しコマンドの前記アドレスが、他の介在コマンドを伴わずに送信された後続の読み出しコマンドのアドレスと一致する場合、前記一連の読み出しコマンドを前記リトライコマンドとしてデコードする、請求項 10 に記載のメモリデバイス。

【請求項 12】

リトライコマンドモードを示すためのフィールドを記憶するモードレジスタをさらに備え、

前記リトライコマンドは、前記リトライコマンドモードにおいて受信された読み出しコマンドを含む、請求項 8 に記載のメモリデバイス。

【請求項 13】

前記ハードウェアインタフェースは、E C Cが適用されていない前記読み出しデータを返すために、前記 E C C 回路をバイパスする、請求項 8 ～ 1 2 のいずれか一項に記載のメモリデバイス。

【請求項 1 4】

前記メモリデバイスは、ダブルデータレート（D D R）規格と互換性のある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）デバイスを含む、請求項 8 ～ 1 3 のいずれか一項に記載のメモリデバイス。

【請求項 1 5】

読み出しコマンドに応答して、読み出しデータに E C C を内部で適用し、および訂正済みデータをメモリコントローラに返すエラー検査訂正（E C C）回路を有するメモリデバイスであって、前記 E C C 回路は、前記訂正済みデータに、検出された訂正不可能エラー（D U E）が含まれることを検出すると、前記訂正済みデータと共に前記 D U E の指標も前記メモリコントローラに返す、メモリデバイスと、

前記読み出しコマンドを生成し、前記読み出しコマンドの後に前記メモリデバイスから受信した前記 D U E の前記指標に応答してリトライコマンドを生成するコマンドロジックを有するメモリコントローラと、

を備え、

前記リトライコマンドは、前記メモリデバイスの前記 E C C 回路による E C C を適用せずに読み出しデータを返すように前記メモリデバイスをトリガする、データを記憶するシステム。

【請求項 1 6】

前記リトライコマンドは、前記リトライコマンドを示すためのコマンドエンコーディングを有するコマンドを含む、請求項 1 5 に記載のシステム。

【請求項 1 7】

前記リトライコマンドは、他の介在コマンドを伴わない一連の読み出しコマンドを含み、前記メモリデバイスは、前記一連の読み出しコマンドを前記リトライコマンドとしてデコードする、請求項 1 5 に記載のシステム。

【請求項 1 8】

前記メモリデバイスは、リトライコマンドモードを示すためのフィールドを記憶するモードレジスタをさらに備え、

前記メモリコントローラは、前記メモリデバイスが前記リトライコマンドモードである間に、前記リトライコマンドを示すために読み出しコマンドを送信する、請求項 1 5 に記載のシステム。

【請求項 1 9】

前記メモリデバイスは、ダブルデータレート（D D R）規格と互換性のある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）デバイスを含む、請求項 1 5 ～ 1 8 のいずれか一項に記載のシステム。

【請求項 2 0】

前記メモリコントローラに結合されているホストプロセッサデバイス、
ホストプロセッサに通信可能に結合されているディスプレイ、
前記ホストプロセッサに通信可能に結合されているネットワークインタフェース、または、
前記システムに電力を供給するバッテリー、
のうちの 1 つまたは複数をさらに備える、請求項 1 5 ～ 1 9 のいずれか一項に記載のシステム。

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

本説明は、概してメモリデバイスに関し、より詳細な説明は、内部エラー検査訂正（E C C）の選択的適用に関する。

【背景技術】

【0002】

メモリサブシステムにおけるエラー処理は、RAS期待を満たすために用いられるエラー処理を指すように、RAS（信頼性、可用性、保守性）と称することができる。RAS期待は、多くの場合、フルSDDC（シングルデバイスデータ訂正）能力に関する期待を含み、ここでは、フルデバイス障害から生じるエラーを訂正することができる。しかしながら、将来世代のメモリサブシステムは、より少数のデバイスからより多くのデータが提供されるため、フルSDDCを提供するのに十分なECC（エラー検査訂正）ビットを有しないであろう。x4（デバイス毎の4ビット幅データインタフェース）DRAM（ダイナミックランダムアクセスメモリ）デバイスがDDR4（ダブルデータレート、バージョン4、JESD79-4、JEDECによって2012年9月に最初に公開された）の下で単一のデバイスから4B（4バイト）のデータを提供する一方で、DDR5（現在JEDECによって議論中のDDRバージョン5）x4DRAMデバイスは、単一のデバイスから8Bのデータを提供する。HBM（例えば、高帯域幅メモリDRAMであるJESD235A、JEDECによって2015年11月に最初に公開された）では、単一のデバイスは、全32Bの読み出しデータを提供する。

10

【0003】

さらに、オンダイECCは一般に、シングルビットエラー（SBE）の訂正を提供することができるが、デバイスがマルチビットエラー（MBE）を有する場合、メモリデバイスECC回路は、MBEをSBEとして誤訂正することによって、さらなるエラーを発生させる可能性がある。このような誤訂正は、エイリアシングと称され、マルチビットエラーに基づいて生成されたECCシンドロームは、コードマトリクスの列と一致し、結果として正しいビットを誤って反転させてしまう。したがって、例えば、内部ECCによって、2ビットエラーが3ビットエラーへと「訂正」される可能性がある。

20

【0004】

メモリコントローラは、複数のメモリデバイスからのデータを適用することにより、システムレベルでよりロバストなECCを適用することが可能であり得る。しかしながら、エイリアシングは、メモリコントローラが他の状況では訂正できるであろう訂正不可能エラーをもたらす可能性がある。

【図面の簡単な説明】

【0005】

30

以下の説明は、実装の例として与えられる例示を有する図についての議論を含む。図面は、例として理解されるべきであり、限定として理解されるべきではない。本明細書において用いられる場合、1つまたは複数の例への言及は、本発明の少なくとも1つの実装に含まれる特定の特徴、構造または特性の説明と理解されるべきである。本明細書に現れる「1つの例において」または「代替的な一例において」などの語句は、本発明の実装の例を提供し、必ずしも全てが同じ実装を指すわけではない。しかしながら、それらは、必ずしも相互に排他的であるわけでもない。

【0006】

【図1】メモリがオンメモリECCを無効化するリトライ読み出しを実行するコマンドをホストが選択的に提供するシステムの一例のブロック図である。

40

【0007】

【図2】メモリが繰り返し読み出しコマンドに基づいてオンメモリECCを無効化するリトライ読み出しを実行するコマンドをホストが選択的に提供するシステムの一例のブロック図である。

【0008】

【図3A】メモリデバイスECCの選択的バイパスを実装し得るメモリアーキテクチャの一例のブロック図である。

【0009】

【図3B】未訂正読み出しデータを受信するメモリコントローラの一例のブロック図である。

50

【0010】

【図4A】選択的メモリデバイスECCバイパスを可能にするメモリアーキテクチャの64ビットデータアーキテクチャの一例のブロック図である。

【0011】

【図4B】選択的メモリデバイスECCバイパスを可能にするメモリアーキテクチャの128ビットデータアーキテクチャの一例のブロック図である。

【0012】

【図5】選択的メモリデバイスECCバイパスを伴うメモリサブシステムの状態図の一例の図である。

【0013】

【図6】オンダイECCを選択的にバイパスするプロセスの一例のフロー図である。

【0014】

【図7】選択的オンダイECCバイパスを実装するオンダイエラー検査訂正（ECC）サブシステムの一部のブロック図である。

【0015】

【図8】選択的オンダイECCバイパスを実装できるメモリサブシステムの一部のブロック図である。

【0016】

【図9】選択的オンダイECCバイパスを実装できるコンピューティングシステムの一部のブロック図である。

【0017】

【図10】選択的オンダイECCバイパスを実装できるモバイルデバイスの一部のブロック図である。

【0018】

特定の詳細および実装の説明を以下で行う。当該説明は、例のいくつかまたは全てと他の潜在的な実装とを示し得る図の非限定的な説明を含む。

【発明を実施するための形態】

【0019】

本明細書で説明されるように、内部ECC（エラー検査訂正）を実行するメモリデバイスは、メモリコントローラからの異なる読み出しコマンドにตอบสนองして、内部ECCを適用してまたは内部ECCを適用せずに、読み出しデータを選択的に返すことができる。メモリデバイスは通常、通常の読み出しコマンドにตอบสนองして、ECCを適用して、訂正済みデータを返すことができる。リトライコマンドにตอบสนองして、メモリデバイスは、内部ECCを適用せずに読み出しデータを返すことができる。

【0020】

次世代のメモリデバイスは、読み出し操作毎のビットの総数に対するデバイス毎のデータ信号線の比はるかにより小さいこと（チャネルの縮小）が期待されている。加えて、ECCビットは、ポイズニング、ディレクトリ、セキュリティ、または他の用途のために再利用化されることがある。したがって、将来世代のメモリデバイスにおいて、フルSDDC（シングルデバイスデータ訂正）能力を提供するのに十分なECCビットが存在する可能性は低い。メモリシステム内で同RAS（信頼性、可用性、保守性）の期待の達成は、ホストまたはメモリコントローラによるECC操作に依拠する。メモリコントローラによる操作は、ECCエイリアシングによって妨げられるが、これには、メモリデバイスによる内部ECCの選択的な使用によって対抗することができる。

【0021】

メモリデバイスが、内部ECC（例えば、SBE（シングルビットエラー）訂正）を有効化せずにデータを送信することになるリトライ読み出しコマンドは、ホストが、誤訂正を伴わずに特定のメモリデバイスからデータを受信することを可能にする。ホストは、内部ECCによってもたらされたエラーを識別するために、エイリアスされた訂正済みデータを未訂正データと比較することができる。このような操作によって、ホストがDUE（

10

20

30

40

50

検出可能だが訂正不可能なエラー) エラーを C E (訂正可能なエラー) エラーに変換することを可能にすることができる。したがって、ホストは、メモリサブシステムの R A S カバレッジを改善することができる。R A S カバレッジは、メモリアーキテクチャに応じて、例えば、最大 1 6 b (ビット) または最大 3 2 b について対象とされ得る。

【0022】

3 2 G B (ギガバイト) x 4 R D I M M (レジスタードデュアルインラインメモリモジュール) の試験では、フルデバイス障害が一般的な障害の形態でないことが示された。識別され得るエラーに関しては、S B E の数が、他のいかなるエラーよりもはるかに多かった。S B E の比率は、より小さなパターンングおよびフィーチャサイズの縮小、ならびに V R T (可変保持時間) の使用に起因して、メモリデバイスにおいて増加し続ける。その次に最も一般的なものは M B E (マルチビットエラー) であり、その次は行エラー (またはキャッシュラインエラー) であった。内部 E C C またはオンダイ E C C は、一般に S B E の対処に有効であるが、M B E を D U E エラーにエイリアスする可能性がある。

【0023】

1 つの例において、メモリコントローラは、一群のメモリデバイスからのデータにおける D U E の検出に応答して、リトライコマンド (代替的には、読み出しリトライコマンド、またはリトライ読み出しコマンド) を生成する。例えば、内部 E C C の適用の一部として、メモリデバイスは、内部 E C C によって、メモリデバイスが内部で訂正できないエラーが検出された場合、D U E 指標を生成することができる。1 つの例において、メモリコントローラは、メモリデバイスから受信したデータおよび E C C チェックビットから、D U E を検出することができる。

【0024】

メモリコントローラがリトライコマンドを生成するのに使用することができる様々なメカニズムが存在する。1 つの例において、メモリコントローラは、メモリデバイスまたは一群のメモリデバイスを読み出しリトライモードまたは同等の状態に置き、メモリコントローラは、メモリデバイスの 1 つまたは複数の構成ビットを設定する。1 つまたは複数の構成ビットは、例えば、モードレジスタまたは他の構成レジスタにおけるフィールドであり得る。

【0025】

1 つの例において、メモリコントローラおよびメモリデバイスは、コマンドプロトコルの一部として、リトライコマンドエンコーディングを備える。例えば、標準的な読み出しコマンドが、C A (コマンドおよびアドレス) ビットの特定のエンコーディングによって定義される一方で、読み出しリトライコマンドもまた定義され得る。このような読み出しリトライコマンドは、読み出し、書き込み、リフレッシュ、または他のコマンドなどの他のコマンドとともに、コマンドプロトコルに含まれ得ることが理解される。コマンドエンコーディングは、コマンドが構成設定によってトリガされるリトライモードと連携して用いられるのではない限り、プロトコルにおける他のコマンドとは異なっている必要がある。

【0026】

1 つの例において、メモリコントローラは、連続または一連の読み出しコマンドによってリトライコマンドをトリガする。連続コマンドまたは一連の同じコマンドとは、メモリコントローラが、いかなる介在コマンドも伴わずに同じコマンドを発行することを指すことができる。特定の介在コマンドが、プロトコルによって許容され得る。1 つの例において、「同じコマンド」の発行とは、同じアドレスを有する同じコマンドエンコーディングを指す。したがって、エンコーディングは、一連の同一の C A ビットエンコーディングを含むことができる。

【0027】

1 つの例において、連続読み出しコマンドメカニズムが機能するために、メモリデバイスは、コマンドを同じコマンドとしてデコードする回路を含むことができる。したがって、メモリデバイスは、一連のコマンドを読み出しコマンドのトリガとして内部でデコードすることができる。リトライコマンドの目的が、以前の訂正済みデータと比較するために

10

20

30

40

50

未訂正データを提供することであれば、一連のコマンドによって、メモリデバイスが、元の読み出しコマンドのデコーディングに回答してデータを返し、続いてリトライコマンドをデコードし、内部ECCを適用せずに同じデータを返すようにすることができる。

【0028】

メモリコントローラは、シングルメモリデバイスに結合され得ることが理解される。しかしながら、メモリコントローラによるシステムレベルの訂正は、シングルメモリデバイスのためのホストにおいては異ならないであろう。通常、メモリデバイスは、群で使用される。メモリコントローラは、メモリチャネルを通じて複数のメモリデバイスまたは一群のメモリデバイスに結合され、メモリデバイスは、コマンド信号線およびアドレス信号線を（例えば、マルチドロップバスとともに）共有し、各メモリデバイスは、メモリデバイスとメモリコントローラとの間に結合されている専用のデータ線（DQおよび関連付けられたストロブまたはクロック（DQS）線）を有する。他の制御信号線もまた、共有することができる（例えば、ランク内の複数のデバイスは、デバイスイネーブル（EN）信号線またはアラート信号線を共有することができる）し、または固有のものとすることができる。

10

【0029】

図1は、メモリがオンメモリECCを無効化するリトライ読み出しを実行するコマンドをホストが選択的に提供するシステムの一例のブロック図である。システム100は、コンピュータシステムの要素を表す。システム100は、メモリシステムがコンピュータシステムの一部と見なされるのであれば、メモリシステムまたはメモリサブシステムの要素を示すものと見なすことができる。

20

【0030】

システム100は、ホスト110を備え、ホスト110は、メモリを含むコンピュータシステムのためのハードウェアおよびソフトウェアプラットフォームを表す。ホスト110は、プロセッサ112を備え、プロセッサ112は、任意のタイプのプロセッサ、マイクロプロセッサ、中央処理ユニット（CPU）、グラフィックス処理ユニット（GPU）、またはメモリへのアクセスを要求することになる命令を実行する他のコントローラもしくはプロセッサであり得る。プロセッサ112は、シングルコアプロセッサであり得る。1つの例において、プロセッサ112は、マルチコアプロセッサである。

【0031】

ホスト110は、メモリ130へのアクセスを管理するホスト110における回路を表すメモリコントローラ120を備える。1つの例において、メモリコントローラ120は、プロセッサダイ（統合メモリコントローラまたはオンダイコントローラ）の一部である。1つの例において、メモリコントローラ120は、プロセッサ112およびメモリコントローラ120を含むシステムオンチップ（SOC）に統合された回路である。このようなSOCは、周辺機器コントローラ、通信コントローラ、グラフィックスコントローラ、または他の回路も含み得る。

30

【0032】

1つの例において、メモリコントローラ120は、コマンド（CMD）ロジック122を備え、コマンド（CMD）ロジック122は、プロセッサ112によるメモリアクセスへの要求に回答してメモリ130に対するコマンドを生成するためのロジックを表す。コマンドロジック122は、メモリコントローラ120上の回路およびファームウェアまたはコードロジックを含むことができる。1つの例において、コマンドロジック122は、メモリ130によるリトライコマンドをトリガするためのコマンドエンコーディングを持つコマンドを生成することができる。コマンドロジック122によるコマンドのエンコーディングに基づいて、1つの例において、メモリ130のコマンド（CMD）デコーダ132は、コマンドを、メモリデバイスに内部ECCを適用せずに読み出しデータを返させるリトライコマンドとして解釈する。

40

【0033】

メモリコントローラ120は、ECC124を備え、ECC124は、ホスト側のエラ

50

一検査訂正ロジックを表す。E C C 1 2 4は、システムレベルE C Cを実行する。システムレベルE C Cとは、複数のメモリデバイス1 3 0から受信されたデータのエラー検査訂正を指す。1つの例において、E C C 1 2 4は、メモリ1 3 0から受信されたデータに対してE C C操作を実行するための回路を含む。1つの例において、E C C 1 2 4は、書き込みコマンドのための書き込みデータとともに送信すべきE C Cチェックビットを生成する。
【0 0 3 4】

メモリコントローラ1 2 0は、I / O（入力／出力）1 2 8を備え、I / O（入力／出力）1 2 8は、メモリ1 3 0とインタフェースするためのハードウェアを表す。ハードウェアは、メモリコントローラ1 2 0とメモリ1 3 0との電氣的相互接続を提供するシステム1 0 0の接続基板（例えば、S O Cまたはマザーボード）上の信号線とインタフェースするためのピン、ボール、または他のコネクタを含むことができる。I / O 1 2 8のハードウェアは、ドライバと、受信回路（終端ハードウェア、クロック／タイミング回路、遅延ロックループ（D L L）もしくは位相ロックループ（P L L）などの信号位相制御、または他のハードウェアを含む）を備えることができる。

【0 0 3 5】

メモリ1 3 0は、D R A M（ダイナミックランダムアクセスメモリ）デバイスまたはS D R A M（シンクロナスD R A M）デバイスなどの個々のメモリデバイスを表す。システム1 0 0は、それぞれメモリ1 3 0としてラベルが付されている、複数のメモリデバイスを示す。メモリ1 3 0は、I / O 1 3 8を備え、I / O 1 3 8は、メモリコントローラとインタフェースおよび通信するためのメモリデバイスのハードウェアを表す。バス1 5 0は、メモリ1 3 0とメモリコントローラ1 2 0とを電氣的／通信可能に接続するための1つまたは複数の群の信号線またはトレースを表す。I / O 1 3 8のハードウェアは、メモリコントローラのI / O 1 2 8と同じまたは同様とすることができ、別個に説明しない。

【0 0 3 6】

メモリ1 3 0は、コマンド（C M D）デコーダ1 3 2を備え、コマンド（C M D）デコーダ1 3 2は、メモリコントローラ1 2 0から受信されたコマンドをデコードする回路要素およびロジックを表す。コマンドは、読み出しコマンド、書き込みコマンド、リフレッシュコマンド、または他のコマンドを含むことができる。1つの例において、コマンドデコーダ1 3 2は、リトライコマンドをデコードするためのロジックを含む。

【0 0 3 7】

メモリ1 3 0は、E C C 1 3 4を備え、E C C 1 3 4は、オンダイE C CまたはオンデバイスE C Cを表す。E C C 1 3 4は、メモリ1 3 0がメモリコントローラ1 2 0にデータを提供する前に、当該データのデータエラーを訂正するようにメモリ1 3 0内で動作する。メモリ1 3 0のE C C 1 3 4の動作は、メモリコントローラ1 2 0のE C C 1 2 4の動作とは異なることが理解される。メモリコントローラ1 2 0は、E C Cを計算してチェックビットを送信し、これは複数のメモリデバイスのうちの1つまたは複数に記憶することができる。一部のメモリデバイスは、データのみを記憶し、他のメモリデバイスは、E C Cチェックビットのみを記憶してよい。例えば、一般的なアーキテクチャは、1 0 x 4を備え、すなわち、8つがデータを記憶し、2つがE C C情報を記憶する1 0のメモリデバイスを備え、メモリデバイスがx 4データインタフェースを有する。E C Cのためのより多くのデバイスを追加することで、部品が増加し、また追加のデバイスに接続するための追加の信号線のチャネルピンアウトが増加するため、コストが増大する。

【0 0 3 8】

したがって、E C C 1 2 4は、全てのデータビットに対するE C Cを実行し、全てのメモリデバイスに全体として送信する。E C C 1 3 4は、メモリデバイスに記憶されたビットの一部を返す前に、メモリデバイス内でエラーを検査して訂正するように、メモリ1 3 0内で動作する。システムレベルE C CのR A Sのパフォーマンスは、より多いE C C情報を利用可能であることに起因して、個々のデバイス内のE C CのR A Sのパフォーマンスよりも高いことが期待されることが理解される。

【0 0 3 9】

メモリ130は、レジスタ136を備え、レジスタ136は、メモリ130の構成レジスタを表す。例えば、レジスタ136は、モードレジスタであり得る。レジスタ136は、メモリ130の構成情報を記憶するための1つまたは複数のフィールドを備える。1つの例において、レジスタ136は、メモリコントローラ120がメモリ130をリトライコマンドモードにするために設定することができる1ビットまたは複数ビットの情報を記憶するためのフィールドを備える。リトライコマンドモードは、メモリ130の特定のモードまたは特定の動作状態であり得る。リトライコマンドモードにおいて、コマンドデコーダ132が読み出しコマンドを受信した場合、コマンドデコーダ132は、その読み出しコマンドを、ECC134を適用せずに読み出しをトリガするコマンドとしてデコードする。したがって、メモリ130は、アレイ140からデータを読み出し、内部ECCを適用せずに、そのデータをメモリコントローラ120に返す。このようなモードは、コマンドロジック122によるコマンドエンコーディングに基づくリトライコマンドの代替となるものであり得る。

10

【0040】

メモリ130は、データを記憶するためのアレイ140を備える。アレイ140は、アドレス指定可能なメモリ位置のアレイを表す。通常、アレイ140は、少なくとも行（例えば、ワード線）および列（例えば、ビット線）によってアドレス指定可能である。アレイ140は、様々なバンクもしくはサブバンク、または他の区分に細分化され得る。アレイ140は、データおよびECCデータを記憶する。データは、メモリコントローラ120から書き込まれたデータを含み得ることが理解される。1つの例において、選択されたメモリデバイスにメモリコントローラによって書き込まれたデータの一部または全ては、システムECCデータであり得る。アレイ140内のECCは、内部ECCのためにECC134によって生成されたローカルECCデータまたはECCチェックビットを表す。

20

【0041】

バス150は、メモリコントローラ120とメモリ130との間の電氣的接続および通信接続を提供する。バス150を指し示す破線のボックスは、コマンド（CMD）シーケンス152の1つの例を示す。コマンドシーケンス152は、リトライコマンドが使用されるコマンドシーケンスの1つの非限定的な例として理解される。

【0042】

1つの例において、読み出しコマンドが、メモリコントローラ120からメモリ130に送信される（コントローラからメモリへ、すなわちC→M）。読み出しコマンドに応答して、メモリ130は、アレイ140からのデータにアクセスし、ECC134によってコードワードデータにECC操作を適用してローカルまたは内部ECCチェックビットを加える。内部ECCの適用後、メモリ130は、メモリコントローラ120に訂正済みデータを返す（メモリからコントローラへ、すなわちM→C）。

30

【0043】

1つの例において、「訂正済み」データとは、単に、ECC134によって検査され訂正されたデータを指しており、必ずしもエラーの無いデータを意味しない。ECC134は、SBEを訂正するのに有効であり得るが、アレイ140から読み出されたデータが既にMBEを有している場合は、更なるビットエラーをエイリアスする可能性がある。1つの例において、メモリ130は、データを適切に訂正できなかった場合、DUE指標を発行するなど、ECC134によって指標を生成することができる。1つの例において、メモリコントローラ120は、読み出しデータを受信して、DUEを検出する。

40

【0044】

1つの例において、メモリコントローラ120は、未訂正または訂正不可能エラーの検出、または、メモリ130からのデータにおける既に訂正されているはずの別のエラーの検出に~~応答して~~、メモリ130にリトライコマンドを送信する。リトライコマンドは、同じデータを再度読み出し、ECC134を適用せずにメモリコントローラ120に未訂正データを返すようにメモリ130をトリガする。

【0045】

50

1つの例において、メモリコントローラ120は、フォールト検出器126を備える。1つの例において、リトライコマンドの送信後、メモリコントローラ120は、どのメモリデバイスがエラーを有するデータを返したかを特定するために、フォールト検出器126によってフォールト検出を実行することができる。標準的なフォールト検出またはプロプライエタリフォールト検出が用いられ得る。フォールト検出の具体的なプロセスは、本開示の範囲外である。フォールト検出器126は、メモリコントローラ120が、未訂正エラーを識別し、システムレベルECCを実行してデータの訂正を提供することを可能にする。

【0046】

1つの例において、メモリコントローラ120は、全データにリトライコマンドを使用して、シングルビットエラーを特定することができる。このような例では、コマンドエンコーディングまたはリトライモードを用いる場合、リトライコマンドの適用は、必ずしもエラーの検出に応答するものではない。したがって、リトライコマンドは、最初に通常の読み出しコマンドを送信することなく発行することができる。メモリデバイスのランタイム動作では、1つの例において、リトライコマンドは、読み出しコマンドが送信されるまで送信されない。1つの例において、メモリコントローラ120は、より良好なRASカバレッジを可能にするように、SBEのマップを生成することができる。1つの例において、エラーのマップおよび他のフォールト検出に基づいて、メモリコントローラ120は、過剰に多いエラーの検出に基づいてページをオフラインにすることができる。

【0047】

図2は、メモリが繰り返し読み出しコマンドに基づいてオンメモリECCを無効化するリトライ読み出しを実行するコマンドをホストが選択的に提供するシステムの一例のブロック図である。システム200は、図1のシステム100に係るシステムの一例を提供する。システム200は、一連の繰り返し読み出しコマンドによって示されるリトライコマンドのアーキテクチャをより具体的に示す。簡略化のために、1つのみのメモリデバイスが示されているが、1つよりも多いメモリデバイスが使用される場合があり、典型的な実装では複数のメモリデバイスを含むことが理解され得る。

【0048】

ホスト210は、システム200のホストプラットフォームを表す。ホスト210は、プロセッサ212およびメモリコントローラ220を備える。システム100のプロセッサ112に関する上記の説明は、ホスト210のプロセッサ212にも当てはまり得る。システム100のメモリコントローラ120、コマンドロジック122、ECC124、フォールト検出器126、およびI/O128に関する上記の説明は、システム200のメモリコントローラ220、ECC224、フォールト検出器226、およびI/O228にもそれぞれ当てはまり得る。

【0049】

システム100のメモリ130、コマンドデコーダ132、ECC134、I/O138、およびアレイ140に関する上記の説明は、メモリ230、コマンド(CMD)デコーダ232、ECC234、I/O238、およびアレイ240にもそれぞれ当てはまり得る。同様に、上記のバス150の説明は、バス250に当てはまり得る。

【0050】

コマンド(CMD)ロジック222に関して、コマンドロジック222は、同じアドレスに連続読み出しコマンドを発行することによって、リトライコマンドを発行する。バス250上のコマンドシーケンスは具体的に示されていないが、バス150に関して説明したものと同一コマンドシーケンスがバス250にも適用され得る。シーケンス中のリトライコマンドは、より具体的には、システム200の読み出しコマンドの繰り返しによってトリガされるリトライコマンドである。

【0051】

1つの例において、メモリ230は、読み出しコマンドのアドレスデータを記憶するためにアドレスレジスタまたは他のストレージ要素を備える。1つの例において、コマンド

10

20

30

40

50

デコーダ 232 が読み出しコマンドを検出した場合、コマンドデコーダは、読み出しコマンドのアドレス (ADDR) をレジスタ 262 内に記憶するようにメモリ 230 をトリガする。メモリ 230 は、レジスタ 262 のアドレスを、レジスタ 264 内の前の読み出しアドレス (ADDR-1) と比較することができる。比較器 266 は、これらのアドレスが同じであるかを判定し、比較に応答してリトライ準備出力を発行することができる。

【0052】

1つの例において、連続する読み出しコマンドに応答してのみ、比較が行われる。したがって、例えば、比較器 266 の動作は、現在のコマンドが前のコマンドと同じであるという別の比較に基づいて有効化（またはゲート制御）することができる。1つの例において、メモリ 230 の回路は、あらゆるアクセスコマンドのアドレスをレジスタ 262 内に記憶し、次のアクセスコマンドを受けたら、アドレスを次のアクセスコマンドのアドレスと置き換えられるようにレジスタ 264 にプッシュする。このような実装では、メモリ 230 は、コマンドが読み出しコマンドであるとの判定、またはコマンドが読み出しコマンドであり且つ前のコマンドもまた読み出しコマンドであった（例えば、2つの読み出しコマンドが連続して受信されたという指標）との判定により、読み出しリトライ出力をゲーティング制御する等、読み出しリトライデコーディングを制御するための追加の回路を備えることができる。このようなロジックは、出力によってゲート制御でき、または、読み出しリトライエンコーディング出力を生成するように比較器 266 内の回路によって使用され得る。

【0053】

介在コマンドが受信されなかったことを確実にするために、同様または追加のロジックが追加され得る。例えば、メモリ 230 は、現在のコマンドおよび後続の読み出しコマンド（代替的には現在の読み出しコマンドおよび前の読み出しコマンドと見なされ得る）などの2つの読み出しコマンド間で介在コマンドが受信されていない場合にのみ、連続するコマンドのデコーディングが読み出しリトライをもたらしことを確実にするための回路を備えることができる。介在コマンドに言及する際、読み出しコマンドとそれに続く CAS コマンドとは、1つの読み出しコマンドと見なされ得ることが理解される。したがって、CAS コマンドは、CAS コマンドが読み出し操作の実行をトリガするシーケンスの一部である場合、比較の目的では介在コマンドと見なされない。

【0054】

したがって、1つの例において、メモリ 230 は、読み出しコマンドアドレスを以前の読み出しコマンドアドレスと比較する。読み出しコマンドアドレスが一致する場合、1つの例において、メモリ 230 は、アレイ 240 からのデータにアクセスし、ECC 234 による ECC を適用せずに、そのデータを生データとしてメモリコントローラ 220 に返すことができる。このような操作は、新たなコマンドエンコーディングまたはリトライモードを用いずに、リトライコマンドを提供することができる。

【0055】

図 3A は、メモリデバイスによる ECC の選択的バイパスを実装し得るメモリアーキテクチャの一例のブロック図である。メモリセグメント 302 は、メモリデバイスが接続されているデータチャンネルにデータを提供するためのメモリアレイの一部を表す。セグメント 302 は、8b の ECC データを有する 64b のデータの2つの部分を示す。図表 304 は、ドライバを備えるセグメント 302 のアーキテクチャを示す。

【0056】

図表 304 は、8ビットのデータを各側に駆動するサブワード線 (SWL) ドライバを含み得る。図示するように、破線のボックスは、ビットの中央にあるドライバ (DRVR) によって駆動される 16ビットのデータを示す。ドライバの左のビットは、ドライバの左サブアレイまたは左アームと見なされ得る。同様に、ドライバの右のビットは、ドライバの右サブアレイまたは右アームと見なされ得る。

【0057】

図表 306 は、1つのドライバと、その関連付けられたサブアレイとの拡大図を示す。

左アームとしての左サブアレイおよび右アームとしての右サブアレイは、図表 3 0 6 において具体的にラベルが付されている。右および左は、相対的な用語であり、単に図における向きを指していることが理解される。実際の実装では、左または右と見なされるビットは、入れ替わり得る。したがって、左アームおよび右アームという説明は、一群のビットまたはメモリセルもしくはメモリ位置の間のドライバが、ドライバ回路とメモリアレイのメモリセルまたはビットセルとの間で駆動される必要がある信号線の長さを低減するために、ビットをドライバ回路側のいずれかの物理側に駆動することを単に指す。

【0058】

ドライバの能動コンポーネントから各方向に駆動されるビットが存在する。図示するように、1つのアームに障害が起きると、1つのサブアレイがアクセス不可能になるので、1つのアーム（SWLアーム）の障害は8b障害をもたらすことが理解される。ドライバの障害（SWLドライバ障害）は、両方のアームがアクセス不可能になることを考慮すると、16b障害をもたらす。

【0059】

本明細書に記載するように、1つのアームまたはドライバの障害は、追加のビットエラーにエイリアスされ得るマルチビットエラーをもたらす可能性がある。各部分からのデータは、コードマトリクスを実装するハードコーディング回路、または、コードマトリクスを実装するための動作を実装するソフトウェア／ファームウェアを有するコントローラのいずれかであるECCハードウェアに選択的にフィードされ得る。ECCの適用は、リトライコマンドに応答してECCを無効化することによって選択的にバイパスすることができる。

【0060】

図3Bは、未訂正読み出しデータを受信するメモリコントローラの一例のブロック図である。メモリコントローラ310は、システム100のメモリコントローラ120またはシステム200のメモリコントローラ220に従うメモリコントローラの一例を示す。メモリコントローラ310は、メモリアレイのセグメントであり得るセグメント302から受信されたデータに対して検査を実行する。

【0061】

1つの例において、メモリコントローラ310は、データ受信312によって示されるように、読み出しデータを受信する回路を備える。1つの例において、メモリコントローラ310は、I/Oハードウェアからの受信データを記憶するためのレジスタ314と、内部ECCがメモリデバイスにおいて有効化されている場合における読み出しコマンドに応答して、内部ECCがメモリデバイスにおいて有効化されている場合に受信されたデータの前のバージョンを記憶するためのレジスタ316とを備える。1つの例において、メモリコントローラ310は、リトライコマンドに応答して受信されたデータを表すレジスタ314のデータを、エラーが検出された場合の通常の読み出しコマンドに応答して受信されたデータを表すレジスタ316のデータと比較する。すなわち、メモリコントローラ310は、リトライコマンドを発行済みで、訂正済みデータ（レジスタ316）と比較するために生データ（レジスタ314）を受信済みである。

【0062】

比較器318は、訂正されたデータのビットと未訂正データとの差を示すことができるフォールト検出出力を生成する、ビット毎の比較であり得る。1つの例において、メモリコントローラ310は、エラー訂正を実行するために、データにおけるエラーを検出するためのフォールト検出出力を適用する。このようなフォールト検出は、エラーのソースを識別することができる。

【0063】

メモリデバイスにおける内部ECCで訂正することができるSBEに加えて、他の一般的な障害モードには、ドライバ全体またはドライバの1つのブランチのいずれかであるサブワード線（SWL）ドライバの障害が含まれる。SWLドライバは、1つのアーム上の8ビットおよび別のアームまたはブランチ上の8ビットを駆動するアーキテクチャを備え

10

20

30

40

50

得る。1つのアームまたはブランチの障害は、8ビット障害をもたらす可能性があり、ドライバの障害は、16ビット障害をもたらす可能性がある。いずれのシナリオでも、アーム上の8ビットデータは、エイリアスエラーをもたらす可能性がある。メモリコントローラ310が16bフォールトをカバーしようと試みている場合、エイリアスされたデータが、誤って訂正された1つの追加のビットとともに現れ、他の状況では訂正可能なエラーから訂正不可能なエラーをもたらし得る。

【0064】

1つの例において、メモリコントローラは、生データ、すなわち内部ECCを適用していないデータを提供すべきことをメモリに指示するために、読み出しCAS（列アドレスストロブ）のためのエンコーディングを持つリトライコマンドを送信する。1つの例において、メモリコントローラ310は、未訂正データまたは生読み出しデータを受信するために、構成を介してメモリをリトライコマンドモードに置く。1つの例において、メモリコントローラ310は、リトライコマンドをトリガするために、同じアドレスに連続読み出しコマンドを発行し、この場合は、別個のモードも異なるコマンドエンコーディングも要求されない。このような一例では、（図2を参照して記載したような）メモリは、一連のコマンドをリトライコマンドとしてデコードすることができる。

【0065】

リトライコマンドに応答して、メモリは、メモリ内の生データ、すなわちシングルビット訂正を適用していないデータを返し、これにより、メモリコントローラ310がエイリアスエラーを有しないデータを受信することが可能になる。したがって、メモリからのリトライデータは、RASコントローラ訂正能力によって予め規定されるような、例えば、8b、16b、または32b境界内に含まれるエラーを有する。リトライデータは、SBEを訂正したであろうメモリデバイスからのシングルビットエラーも有する。

【0066】

リトライデータに応答して、1つの例において、メモリコントローラ120は、元のデータをリトライデータと比較する。いずれかのリトライデータがシングルビット誤比較を有している場合、1つの例において、メモリコントローラ310は、シングルビットを反転させる。したがって、データの比較の適用により、エイリアスエラーを示し、メモリコントローラ310がエイリアスエラーを訂正することを可能にすることができる。

【0067】

図4Aは、選択的メモリデバイスECCバイパスを可能にするメモリアーキテクチャの64ビットデータアーキテクチャの一例のブロック図である。図表402は、システム100またはシステム200などのシステムに組み込まれ得る要素を表す。図表402は、DDR4実装などの64ビットデータアーキテクチャのためのコアメモリアーキテクチャを表す。図示するように、メモリコアは、128b内部プリフェッチを実行し、内部ECCのための8bECCを備える。図表402はx4アーキテクチャを具体的に示しているが、x8実装がプリフェッチデータの全128ビットを使用し得る。

【0068】

図示のx4実装において、 $4 \times 16 = 64$ であるので、16のバースト長（BL16）を有する4つの信号線に基づいて、64ビットが、メモリアクセストランザクション毎にホストまたは関連付けられたメモリコントローラと交換され得る。図表402の明るい網掛け付きのブロックが上半分を表す場合、網掛け無しのブロックは、下半分を表し、その逆も同様である。暗い網掛け付きのビットは、ECCビットを表す。図示するように、64bのデータは、下半分からまたは上半分からフェッチすることができ、または上半分と下半分とに分割することができる。これらの半分の分割は、サブアレイからの各8bのうち、4bのみの使用を示唆している。

【0069】

図表402は、4つのDQ（データ）信号線（DQ[3:0]）とインタフェースするためのI/O回路462を持つメモリデバイス412の適用を示す。メモリデバイス412は、メモリアレイ420を備え、メモリアレイ420は、プリフェッチ430において

10

20

30

40

50

異なるサブアレイへと分離されるのが示されている。全てのプリフェッチされたデータが使用されるのではない場合、全てのプリフェッチされたデータをセンス増幅器内に置くことができ、その後、選択 4 4 2 によって示されるように、選択されたデータのみがメモリアクセスのために用いられることが理解される。したがって、プリフェッチ 4 3 0 は、センス増幅器内のデータを表すことができ、選択 4 4 2 は、読み出し操作のために I / O に送信された、センス増幅器からのデータを表す。

【 0 0 7 0 】

例えば、データをセンスアンプ内に置くことができ、その後、アドレス指定が、書き込み操作のために書き込まれる、または読み出し操作のために読み出されるデータの一部を選択するように動作する。アドレス指定は、アーキテクチャの理に合う任意の方式でデータを選択することができる。1つの例において、プリフェッチされたデータの一部は、アクセス操作のために選択される。図示のように、8ビットのうち4ビットが、各サブアレイから選択される。

【 0 0 7 1 】

1つの例において、読み出し操作のために選択された、すなわち選択 4 4 2 されたビットは、E C C 回路 4 5 2 に提供することができる。E C C 回路 4 5 2 は、メモリデバイス 4 1 2 のオンデバイスまたは内部 E C C ロジックを表す。1つの例において、E C C 回路 4 5 2 は、E C C エンジンと見なす、または称することができる。E C C 回路 4 5 2 は、エラーを訂正するために、読み出しデータに E C C を適用した後、そのデータを I / O 回路 4 6 2 に提供してホスト（具体的に図示せず）に出力されるようにする。

【 0 0 7 2 】

1つの例において、メモリデバイス 4 1 2 は、E C C バイパス 4 7 2 を備え、E C C バイパス 4 7 2 は、内部 E C C を適用せずにホストに生読み出しデータを提供するメカニズムを表す。1つの例において、E C C 回路 4 5 2 は、E C C 回路の操作を選択的にバイパスするためのマルチプレクサ (m u x) または他のハードウェアを備える。1つの例において、E C C 回路 4 5 2 は、毎回その操作を実行し、E C C を適用せずにデータを提供する代替データ経路が存在し、これは I / O 回路 4 6 2 において多重化または選択することができる。

【 0 0 7 3 】

リトライコマンドに応答して、メモリデバイス 4 1 2 は、E C C 回路を無効し、E C C 回路を迂回するように生データをルーティングするか、あるいは、生データを選択的に出力する別の経路を設けるかのいずれかによって、E C C 回路 4 5 2 をバイパスすることで、E C C を無効化するものと見なすことができる。したがって、メモリデバイス 4 1 2 は、通常の読み出しコマンドに응答して E C C を適用し、リトライコマンドに응答して E C C の適用をバイパスすることができる。

【 0 0 7 4 】

図 4 B は、選択的メモリデバイス E C C バイパスを可能にするメモリアーキテクチャの 1 2 8 ビットデータアーキテクチャの一例のブロック図である。図表 4 0 4 は、システム 1 0 0 またはシステム 2 0 0 などのシステムに組み込まれ得る要素を表す。図表 4 0 4 は、D D R 5 実装などのコアメモリアーキテクチャを表す。図示するように、メモリコアは、1 2 8 b 内部プリフェッチを実行し、内部 E C C のための 8 b E C C を備える。x 4 実装において、4 × 3 2 = 1 2 8 であるので、3 2 のバースト長 (B L 3 2) を有する 4 つの信号線に基づいて、1 2 8 ビットが、メモリアクセストランザクション毎にホストまたは関連付けられたメモリコントローラと交換され得る。したがって、図示するように、アレイの上半分とアレイの下半分との両方が、それぞれ 6 4 b のデータを提供する。上半分は、メモリアーキテクチャおよびアドレス指定構造に基づくことができ、これは図表 4 0 4 に具体的に示されない。図表 4 0 4 の明るい網掛け付きのブロックは、上半分を表し、網掛け無しのブロックは、下半分を表し、その逆も同様である。暗い網掛け付きのビットは、E C C ビットを表す。

【 0 0 7 5 】

図表 4 0 4 は、4 つの D Q (データ) 信号線 (D Q [3 : 0]) とインタフェースするための I / O 回路 4 6 4 を伴うメモリデバイス 4 1 4 の適用を示す。メモリデバイス 4 1 4 は、メモリアレイ 4 2 0 を備え、メモリアレイ 4 2 0 は、プリフェッチ 4 3 0 において異なるサブアレイへと分離されるのが示されている。選択 4 4 4 は、読み出し操作のために I / O に送信される、センス増幅器からのデータを表し、全 1 2 8 のプリフェッチされたビットを含み得る。

【 0 0 7 6 】

1 つの例において、読み出し操作のために選択された、すなわち選択 4 4 4 されたビットは、E C C 回路 4 5 4 に提供することができる。E C C 回路 4 5 4 は、メモリデバイス 4 1 4 のオンデバイスまたは内部 E C C ロジックを表す。1 つの例において、E C C 回路 4 5 4 は、E C C エンジンと見なす、または称することができる。E C C 回路 4 5 4 は、エラーを訂正するために、読み出しデータに E C C を適用した後、そのデータを I / O 回路 4 6 2 に提供してホスト（具体的に図示せず）に出力されるようにする。

【 0 0 7 7 】

1 つの例において、メモリデバイス 4 1 4 は、E C C バイパス 4 7 4 を備え、E C C バイパス 4 7 4 は、内部 E C C を適用せずにホストに生読み出しデータを提供するメカニズムを表す。1 つの例において、E C C 回路 4 5 4 は、E C C 回路の操作を選択的にバイパスするためのマルチプレクサ (m u x) または他のハードウェアを備える。1 つの例において、E C C 回路 4 5 4 は、毎回その操作を実行し、E C C を適用せずにデータを提供する代替データ経路が存在し、これは I / O 回路 4 6 4 において多重化または選択することができる。

【 0 0 7 8 】

リトライコマンドに応答して、メモリデバイス 4 1 4 は、E C C 回路を無効し、E C C 回路を迂回するように生データをルーティングするか、あるいは生データを選択的に出力する別の経路を設けるかのいずれかによって、E C C 回路 4 5 4 をバイパスすることで、E C C を無効化するものと見なすことができる。したがって、メモリデバイス 4 1 4 は、通常の読み出しコマンドに응答して E C C を適用し、リトライコマンドに응答して E C C の適用をバイパスすることができる。

【 0 0 7 9 】

図 5 は、選択的メモリデバイス E C C バイパスを伴うメモリサブシステムの状態図の一例の図である。図表 5 0 0 は、本明細書の任意の例に従うリトライコマンドのためのリトライシーケンスの状態を示す。最も左の列は、様々な状態を示し、他の列は、D R A M 0、D R A M 1、D R A M 2、およびホスト R A S の状態をそれぞれ示す。

【 0 0 8 0 】

状態 5 1 0 は、元の D R A M の状態を表す。状態 5 2 0 は、D R A M が E C C を有効化してデータを読み出す状態を表す。状態 5 3 0 は、D R A M が E C C を無効化してデータを読み出す状態を表す。状態 5 4 0 は、フォールト検出を実行するためのホストによる操作を表す。1 つの例において、ホストは、M B E を有する D R A M を検出するために、P F D (恒久フォールト検出) シーケンスを発行する。ホストは、フォールトが検出されるまで検出を実行するように、動作を反復して実行することができる。状態 5 5 0 は、ホストが、様々な D R A M に対する新たなデータをホストにおいて生成する状態を表す。

【 0 0 8 1 】

状態 5 1 0 に関して、D R A M 0 が 8 ビットエラーを有し、D R A M 1 が 1 ビットエラーを有し、D R A M 2 がエラーを有しない一例を検討する。これらの D R A M からのデータが、ホストにおいて S D D C コード保護を持つ状態のホストに提示される場合、この例において S D D C コードは 8 b 訂正に限られているので、結果として D U E がもたらされる。

【 0 0 8 2 】

状態 5 2 0 では、これらの D R A M デバイスは、読み出し値に E C C を適用する。したがって、D R A M 0 は、別のビットを反転させることによって 8 ビットエラーを誤訂正し

10

20

30

40

50

得るので、エイリアシング問題を有し得る。したがって、8ビットエラーは、+1ビットエイリアスエラーを伴う8ビットエラーになる。オンダイECCは、DRAM1上のエラーを修正し、DRAM2上ではデータは変更されない。したがって、DRAM1およびDRAM2の両方が、0のエイリアスエラーを伴う0のエラーを有する。しかしながら、DRAM0の8ビットエラーに起因して、ホストはDUEを検出する。

【0083】

ホストによるDUEの検出に回答して、1つの例において、ホストは、DRAMにおいてECCを無効化して読み出しをリトライするために、状態530に移行する。状態530のリトライコマンドに回答して、DRAM0は、エイリアスエラーを伴わない8ビットエラーを返す。DRAM1は、1bエラーを返し、DRAM2は、エラーを伴わないデータを返す。未訂正データには、ホストにおけるDUEをもたらすエラーが存在することが理解される。

【0084】

1つの例において、状態540では、ホストは、MBEを検出するためのPFD動作を実行する。1つの例において、ホストは、固定書き込みパターン（例えば、AAAAh）コマンドと、それに続いてページを閉じるプリチャージコマンドとを発行する。ページを閉じることで、データがセンスアンプからDRAMセルに移動されることが確実になる。1つの例において、ホストは、その後、同じアドレスに読み出しコマンドを発行し、次に、そのデータを書き込みパターンと比較する。このような操作により、DRAM0が複数のエラーを返すので、マルチビットエラーをDRAM0に特定することができる。

【0085】

状態550にて、ホストは、DRAM0の8bエラーを識別し、エイリアスエラーを取り除くことができる。したがって、ホストは、訂正されたデータを用いて、DRAM1およびDRAM2には、そのデータにエラーが無く、DRAM0には8bエラーのみがある状態にする。ホストにおけるRAS結果は、DUEではなく訂正可能エラー（CE）である。

【0086】

図6は、オンダイECCを選択的にバイパスするプロセスの一例のフロー図である。プロセス600は、読み出し操作にオンダイECCを選択的に適用するプロセスを表す。プロセス600は、システム100またはシステム200に従うシステムによって実行することができる。1つの例において、602にて、ホストは、メモリデバイスに読み出しコマンドを送信する。1つの例において、ECCを使用する読み出しコマンドを最初に送信することなく、リトライコマンドが使用され得る。604にて、読み出しコマンドに回答して、メモリデバイスは、読み出しコマンドをデコードする。

【0087】

1つの例において、メモリデバイスは、読み出しコマンドが通常の読み出しコマンドであるか、またはリトライコマンドであるかを判定する。606のNO分岐にて、読み出しコマンドが通常の読み出しコマンドである場合、608にて、メモリデバイスは通常、内部でECCを実行する。610にて、ECCの実行は、エラーの検出および訂正を含むことができる。次に、612にて、メモリデバイスは、ホストに訂正済みデータを返す。上述したように、このような操作は、エイリアスエラーをもたらす可能性がある。

【0088】

メモリデバイスが、内部ECCが適用されたデータを返した後、ホストは、受信されたデータがリトライコマンドのためのものであるか、または通常のコマンドのためのものであるかを判定する。616のNO分岐にて、データがリトライコマンドのためのものでない場合、ホストは、そのデータがDUEをもたらすかを判定することができる。622のNO分岐にて、データがDUEエラーを含まない場合、626にて、ホストはデータを処理する。

【0089】

1つの例において、622のYES分岐にて、ホストがDUEを特定した場合、624

10

20

30

40

50

にて、ホストは、メモリに読み出しリトライコマンドを送信する。1つの例において、リトライコマンドは、メモリデバイスにおいて受信され、604にて、メモリデバイスは、読み出しコマンドをデコードする。606のYES分岐にて、リトライコマンドに回答して、メモリデバイスは、読み出しコマンドをリトライコマンドとしてデコードする。614にて、リトライコマンドに回答して、メモリデバイスは、ECCを無効化してデータを読み出す。612にて、ECCを無効化して読み出す場合、メモリデバイスは、ホストに生データを返す。

【0090】

616のYES分岐にて、ホストにおいて受信されたデータがリトライコマンドのためのものである場合、1つの例において、618にて、ホストは、フォールト検出を実行する。フォールト検出は、ホストがメモリデータ内でエイリアスエラーがどこに存在し得るかを特定することを可能にする。1つの例において、620にて、ホストは、生データとメモリ訂正済みデータとの比較から収集された情報に基づいて、エラー訂正を実行する。

【0091】

図7は、選択的オンダイECCバイパスを実装するためのオンデバイスECCエンジンの一例のブロック図である。システム700は、システム100またはシステム200に従うシステムのためのオンダイECC回路の一例を提供する。ホスト710は、メモリコントローラ、または、メモリ720へのアクセスを管理する同等もしくは代替の回路もしくはコンポーネントを備える。ホスト710は、メモリ720から読み出されたデータに対して外部ECCを実行する。メモリ720は、ホスト710にデータを送信する前に当該データを検査して訂正するオンダイECCを実装する。

【0092】

システム700は、メモリ720における書き込み経路732を示す。書き込み経路732は、ホスト710からメモリ720に書き込まれるデータの経路を表す。ホスト710は、メモリアレイに書き込むために、データ742をメモリ720に提供する。1つの例において、メモリ720は、チェックビット生成器722でチェックビット744を生成し、メモリ内のデータとともに記憶する。チェックビット744は、ECCビットと称され、メモリ720が、メモリアレイへの書き込みおよびメモリアレイからの読み出しの際に発生し得るエラーを訂正することを可能にすることができる。データ742およびチェックビット744は、メモリリソースに書き込まれるコードワードin746として含むことができる。

【0093】

読み出し経路734は、メモリ720からホスト710に対し読み出されるデータの経路を表す。1つの例において、書き込み経路732および読み出し経路734の少なくとも特定のハードウェアコンポーネントは、同じハードウェアである。1つの例において、メモリ720は、ホスト710からの読み出しコマンドに回答して、コードワードout752をフェッチする。コードワードは、データ754およびチェックビット756を含むことができる。データ754およびチェックビット756はそれぞれ、書き込み経路732において書き込まれるデータ742およびチェックビット744に対応し得る。したがって、読み出しは、データおよびECCビットにアクセスすることができる。

【0094】

読み出し経路734におけるエラー訂正は、エラーを検出して、(シングルビットエラーの場合)エラーを選択的に訂正するために、対応するH行列へのXOR(排他的論理和)ツリーの適用を含むことができることが理解される。XORツリーの構造は、H行列を実装するように設計することができる。当該技術分野で理解されるように、H行列はコードワードのデジット(digits)の線形結合がいかにゼロに等しいかを示すハミングコードパリティチェック行列を指す。したがって、H行列の行は、コードワードの一部となる要素またはデジットのために満たさなければならないパリティチェック式の係数を特定する。1つの例において、メモリ720は、シンドロームデコード724を備える。シンドロームデコード724は、メモリが読み出しデータ内のエラーを検出するためにチェ

10

20

30

40

50

ックビット 7 5 6 をデータ 7 5 4 に適用することを可能にする。シンドロームデコード 7 2 4 は、読み出しデータのための適切なエラー情報の生成に使用されるシンドローム 7 5 8 を生成することができる。データ 7 5 4 は、検出されたエラーの訂正のために、エラー訂正 7 2 8 にも転送され得る。

【0095】

1つの例において、シンドロームデコード 7 2 4 は、シンドローム 7 5 8 をシンドローム生成器 7 2 6 に送り、エラーベクトルを生成させる。1つの例において、チェックビット生成器 7 2 2 およびシンドローム生成器 7 2 6 は、メモリデバイスのための対応する H 行列によって完全に指定される。1つの例において、読み出しデータ内にエラーが無い（例えば、ゼロシンドローム 7 5 8）場合、シンドローム生成器 7 2 6 は、エラー無し信号 7 6 2 を生成する。1つの例において、読み出しデータ内に複数のエラーが存在する（例えば、対応する H 行列のいずれの列にも一致しない非ゼロシンドローム 7 5 8）場合、シンドローム生成器 7 2 6 は、検出された未訂正エラーを示す DUE（検出された未訂正エラー）信号 7 6 4 を生成する。DUE 信号 7 6 4 は、メモリ 7 2 0 が内部 ECC によって訂正することができなかったマルチビットエラーを示すことができる。

【0096】

1つの例において、シングルビットエラー（例えば、対応する H 行列の 1つの列に一致する非ゼロシンドローム 7 5 8）が存在する場合、シンドローム生成器 7 2 6 は、エラー位置 7 6 0 を伴う CE（訂正済みエラー）信号を生成することができ、これは、エラー訂正ロジック 7 2 8 への訂正済みエラーの指標である。エラー訂正 7 2 8 は、データ 7 5 4 内の指定位置に訂正済みエラーを適用して、ホスト 7 1 0 に出力すべき訂正済みデータ 7 6 6 を生成することができる。1つの例において、エラー訂正 7 2 8 は、チェックビット 7 6 8 も生成し、チェックビット 7 6 8 は、読み出しデータのためのチェックビットを含む。

【0097】

チェックビット 7 6 8 は、ホスト 7 1 0 に送信される読み出しデータ内のエラーの状態を示すエラーベクトルと見なされ得る。ゼロシンドローム（エラー無し 7 6 2）条件および訂正済みデータ 7 6 6 を生じさせる訂正済み SBE は、エラー無しをホスト 7 1 0 に示す同じチェックビット 7 6 8 を有することが理解される。したがって、チェックビット 7 6 8 は、メモリ 7 2 0 内の SBE に関する情報を提供せず、マルチビットエラーのみに関する情報を提供する。1つの例において、システム 7 0 0 は、訂正済みデータをメモリアレイに書き戻す。

【0098】

1つの例において、読み出し経路 7 3 4 は、リトライコマンド 7 7 0 を含み、リトライコマンド 7 7 0 は、リトライコマンドに応答するためのシステム 7 0 0 のロジックを表す。1つの例において、リトライコマンドに応答して、メモリ 7 2 0 は、データ 7 5 4 を未訂正データ 7 7 2 として転送する。リトライコマンド 7 7 0 からの未訂正データ 7 7 2 を出力する経路は、ECC バイパス経路と見なされ得、この場合、ECC 回路は読み出し経路 7 3 4 に適用されない。リトライコマンド 7 7 0 は、受信された読み出しコマンドがリトライコマンドであることを示す入力を有することができる。したがって、他の経路は、ホスト 7 1 0 にデータを出力するために用いられないが、リトライ経路は、生データ、すなわち内部 ECC を適用せずにメモリアレイから取得されたままのデータ 7 5 4 を提供する。

【0099】

図 8 は、選択的オンダイ ECC バイパスを実装できるメモリサブシステムの一例のブロック図である。システム 8 0 0 は、コンピューティングデバイスのプロセッサおよびメモリサブシステムの要素を含む。システム 8 0 0 は、システム 1 0 0 またはシステム 2 0 0 と互換性のあるシステムを組み込み得るシステムの 1つの例である。

【0100】

1つの例において、メモリ 8 4 0 は、メモリコントローラ 8 2 0 に読み出しデータを返

10

20

30

40

50

す前に、当該読み出しデータにECCを実行するためのオンダイECC882を備える。1つの例において、メモリ840は、オンダイECC882を選択的にバイパスするためのバイパス884を備える。内部ECCのバイパスは、本明細書の任意の説明に従うものであることができ、メモリ840がリトライコマンドに応答して読み出しを実行する。メモリ840は、リトライコマンドに基づいて、データを返す前に、データにECCを実行する代わりに、メモリアレイ860から生データを返す。

【0101】

プロセッサ810は、オペレーティングシステム(OS)およびアプリケーションを実行し得るコンピューティングプラットフォームの処理ユニットを表し、当該処理ユニットは、メモリのホストまたはユーザと総称され得る。OSおよびアプリケーションは、メモリアクセスをもたらす操作を実行する。プロセッサ810は、1つまたは複数の別個のプロセッサを含み得る。別個のプロセッサのそれぞれは、シングル処理ユニット、マルチコア処理ユニットまたはそれらの組み合わせを含み得る。処理ユニットは、CPU(中央処理ユニット)などのプライマリプロセッサ、GPU(グラフィックス処理ユニット)などの周辺プロセッサまたはその組み合わせであり得る。メモリアクセスは、ネットワークコントローラまたはハードディスクコントローラなどのデバイスによっても開始され得る。そのようなデバイスは、バス(例えば、PCIエクスプレス)を介して、いくつかのシステム内のプロセッサと統合され得るか、もしくはプロセッサに取り付けられ得るか、またはそれらの組み合わせであり得る。システム800は、SoC(システムオンチップ)として実装され得るか、またはスタンドアロンコンポーネントとともに実装され得る。

【0102】

メモリデバイスへの言及は、様々なメモリタイプに適用できる。メモリデバイスは、多くの場合、揮発性メモリ技術を指す。揮発性メモリとは、デバイスへの電力が遮断された場合に状態(およびそれに格納されるデータ)が不定になるメモリである。不揮発性メモリとは、デバイスへの電力が遮断された場合でも状態が確定しているメモリを指す。ダイナミック揮発性メモリでは、状態を維持するためにデバイスに格納されているデータをリフレッシュする必要がある。ダイナミック揮発性メモリの1つの例には、DRAM(ダイナミックランダムアクセスメモリ)、またはシンクロナスDRAM(SDRAM)などの何らかの派生形が含まれる。本明細書で説明されるメモリサブシステムは、DDR4(ダブルデータレート(DDR)バージョン4、JESD79-4、2012年9月にJEDEC(Joint Electron Device Engineering Council)により最初に公開された)、LPDDR4(低電力DDRバージョン4、JESD209-4、2014年8月にJEDECにより最初に公開された)、WIO2(Wide I/O 2(Wide IO2)、JESD229-2、2014年8月にJEDECにより最初に公開された)、HBM(高帯域幅メモリDRAM、JESD235A、2015年11月にJEDECにより最初に公開される)、DDR5(DDRバージョン5、現在JEDECによって議論中)、LPDDR5(LPDDRバージョン5、JESD209-5、2019年2月にJEDECにより最初に公開された)、HBM2((HBMバージョン2)、現在JEDECによって議論中)、またはその他またはメモリ技術の組み合わせ、およびそのような仕様の派生または拡張に基づく技術などの複数のメモリ技術と互換性があり得る。

【0103】

1つの例において、揮発性メモリに加えて、または代替的に、メモリデバイスへの言及は、デバイスへの電力が遮断された場合でも状態が確定する不揮発性メモリデバイスを指すことができる。1つの例において、不揮発性メモリデバイスは、NAND(非AND)またはNOR(非OR)技術などのブロックアドレス指定可能なメモリデバイスである。したがって、メモリデバイスは、三次元クロスポイントメモリデバイス、他のバイトアドレス指定可能な不揮発性メモリデバイスなど、将来世代の不揮発性デバイスも含むことができる。メモリデバイスは、メモリセルの抵抗状態またはメモリセルの位相に基づいてデータを格納する、不揮発性のバイトアドレス指定可能媒体を含み得る。1つの例において

、メモリデバイスは、カルコゲナイド相変化材料（例えば、カルコゲナイドガラス）を用い得る。1つの例において、メモリデバイスは、多閾値レベルのNANDフラッシュメモリ、NORフラッシュメモリ、シングルまたはマルチレベルの相変化メモリ（PCM）またはスイッチ付き相変化メモリ（PCMS）、抵抗メモリ、ナノワイヤメモリ、強誘電体トランジスタランダムアクセスメモリ（FeTRAM）、メモリストア技術を組み込んだ磁気抵抗ランダムアクセスメモリ（MRAM）メモリ、またはスピントランスファトルク（STT）-MRAM、または上記のいずれかの組み合わせ、またはその他のメモリとすることができるか、あるいはそれらを含むことができる。

【0104】

メモリコントローラ820は、システム800用の1つまたは複数のメモリコントローラ回路またはデバイスを表す。メモリコントローラ820は、プロセッサ810による操作の実行に応答してメモリアクセスコマンドを生成する制御ロジックを表す。メモリコントローラ820は、1つまたは複数のメモリデバイス840にアクセスする。メモリデバイス840は、上記で言及されたいずれかのものに係るDRAMデバイスであり得る。1つの例において、メモリデバイス840は、異なるチャネルとして編成および管理される。各チャネルは、複数のメモリデバイスに並列に結合するバスおよび信号線に結合する。各チャネルは、独立して動作可能である。故に、各チャネルは、独立してアクセスおよび制御され、タイミングと、データ転送と、コマンドおよびアドレスの交換と、他の操作とは、各チャネルについて別個である。結合は、電氣的結合、通信的結合、物理的結合またはこれらの組み合わせを指し得る。物理的結合は、直接接触を含み得る。電氣的結合は、コンポーネント間に電気が流れることを可能にするか、もしくはコンポーネント間のシグナリングを可能にするか、またはそれらの両方を可能にするインタフェースまたは相互接続を含む。通信的結合は、有線または無線を含む、コンポーネントがデータを交換することを可能にする接続を含む。

【0105】

1つの例において、各チャネルの設定は、別個のモードレジスタまたは他のレジスタ設定により制御される。1つの例において、各メモリコントローラ820が別個のメモリチャネルを管理するが、システム800は、単一のコントローラにより管理される複数のチャネルまたは単一のチャネル上の複数のコントローラを有するように構成され得る。1つの例において、メモリコントローラ820は、同じダイ上に実装されるか、またはプロセッサと同じパッケージ空間に実装されるロジックなど、ホストプロセッサ810の一部である。

【0106】

メモリコントローラ820は、上記で言及したメモリチャネルなどのメモリバスに結合するI/Oインタフェースロジック822を含む。I/Oインタフェースロジック822（およびメモリデバイス840のI/Oインタフェースロジック842）は、ピン、パッド、コネクタ、信号線、トレースもしくはワイヤ、もしくは当該デバイスに接続する他のハードウェア、またはこれらの組み合わせを含み得る。I/Oインタフェースロジック822は、ハードウェアインタフェースを含み得る。示されるように、I/Oインタフェースロジック822は少なくとも、信号線用のドライバ/トランシーバを含む。一般に、集積回路インタフェース内のワイヤは、パッド、ピンまたはコネクタと結合して、デバイス間の信号線もしくはトレースまたは他のワイヤとインタフェースする。I/Oインタフェースロジック822は、デバイス間の信号線上で信号を交換するためのドライバ、受信機、トランシーバもしくは終端もしくは他の回路または回路の組み合わせを含み得る。信号の交換は、送信または受信のうちの少なくとも1つを含む。メモリコントローラ820からメモリデバイス840のI/O842へI/O822を結合するように示されるが、メモリデバイス840の群が並列にアクセスされるシステム800の実装では、複数のメモリデバイスがメモリコントローラ820の同じインタフェースへのI/Oインタフェースを含み得ることが理解される。1つまたは複数のメモリモジュール870を含むシステム800の実装では、I/O842は、メモリデバイス自体上のインタフェースハードウェア

10

20

30

40

50

アに加え、メモリモジュールのインタフェースハードウェアを含み得る。他のメモリコントローラ 820 は、他のメモリデバイス 840 への別個のインタフェースを含む。

【0107】

メモリコントローラ 820 とメモリデバイス 840 との間のバスは、メモリコントローラ 820 をメモリデバイス 840 へ結合する複数の信号線として実装され得る。バスは通常、少なくとも、クロック (CLK) 832、コマンド／アドレス (CMD) 834 および書き込みデータ (DQ) および読み出しデータ (DQ) 836 ならびにゼロまたはそれより多くの他の信号線 838 を含み得る。1つの例において、メモリコントローラ 820 とメモリとの間のバスまたは接続は、メモリバスと称され得る。1つの例において、メモリバスは、マルチドロップバスである。CMD 用の信号線は、「C/A バス」(もしくは ADD/CMD バス、またはコマンド (C または CMD) およびアドレス (A または ADD) 情報の転送を示す何らかの他の名称) と称することができ、書き込みおよび読み出し DQ 用の信号線は、「データバス」と称することができる。1つの例において、独立のチャネルは、異なるクロック信号、C/A バス、データバスおよび他の信号線を有する。故に、システム 800 は、独立のインタフェース経路が別個のバスと見なされ得るという意味において、複数の「バス」を有していると思なされ得る。明示的に示される線に加え、バスは、ストロブシグナリング線、アラート線、補助線もしくは他の信号線またはそれらの組み合わせのうちの少なくとも 1 つを含み得ることが理解される。シリアルバス技術は、メモリコントローラ 820 とメモリデバイス 840 との間の接続のために用いられ得ることも理解される。シリアルバス技術の一例が、8B10B エンコーディングおよび各方向における信号の単一の差動ペアを介した埋め込みクロックでの高速データの伝送である。1つの例において、CMD 834 は、複数のメモリデバイスと並列に共有される信号線を表す。1つの例において、複数のメモリデバイスが CMD 834 のエンコーディングコマンド信号線を共有し、それぞれが、個々のメモリデバイスを選択するための別個のチップ選択 (CS_n) 信号線を有する。

【0108】

システム 800 の例において、メモリコントローラ 820 とメモリデバイス 840 との間のバスは、補助コマンドバス CMD 834 と書き込みデータおよび読み出しデータを伝達するための補助バス DQ 836 とを含むことが理解される。1つの例において、データバスは、読み出しデータおよび書き込み／コマンドデータ用の双方向線を含み得る。別の例において、補助バス DQ 836 は、ホストからメモリへの書き込みおよびデータのための一方向書き込み信号線を含み得るとともに、メモリからホストへの読み出しデータのための一方向線を含み得る。選択されたメモリ技術およびシステム設計に従って、他の信号 838 は、ストロブ線 DQS などのバスまたはサブバスに付随し得る。システム 800 の設計、またはある設計が複数の実装をサポートする場合の実装に基づいて、データバスは、メモリデバイス 840 毎に多少の帯域幅を有し得る。例えば、データバスは、x4 インタフェース、x8 インタフェース、x16 インタフェースまたは他のインタフェースのいずれかを有するメモリデバイスをサポートできる。規定「xW」のWは、メモリデバイス 840 のインタフェースサイズまたはインタフェースの幅を指す整数であり、メモリコントローラ 820 とデータを交換するための信号線の数を表す。メモリデバイスのインタフェースサイズは、いくつのメモリデバイスがシステム 800 内のチャネル毎に同時に用いられ得るか、または同じ信号線へ並列に結合され得るかに関する制御要因である。1つの例において、高帯域幅メモリデバイス、ワイドインタフェースデバイスもしくは積層メモリ構成またはそれらの組み合わせは、x128 インタフェース、x256 インタフェース、x512 インタフェース、x1024 インタフェースまたは他のデータバスインタフェース幅などのより幅広いインタフェースを有効化できる。

【0109】

1つの例において、メモリデバイス 840 およびメモリコントローラ 820 は、バーストで、または一連の連続するデータ転送で、データバスを介してデータを交換する。バーストは、バス周波数に関連する転送サイクルの数に対応する。1つの例において、転送サ

10

20

30

40

50

イクルは、同じクロックまたはストロブ信号エッジ（例えば、立ち上がりエッジ）で生じる転送の全クロックサイクルであり得る。1つの例において、システムクロックのサイクルを参照する全クロックサイクルは、複数の単位間隔（UI）へ分割される。各UIは、転送サイクルである。例えば、ダブルデータレート転送は、クロック信号の両方のエッジ（例えば、立ち上がりおよび立ち下がり）でトリガする。バーストは、構成された数のUIにわたって続き得る。これは、レジスタに格納される構成またはオンザフライでトリガされる構成であり得る。例えば、一連の8個の連続する転送期間は、バースト長8（BL8）と見なされ得、各メモリデバイス840は、各UIでデータを転送できる。故に、BL8で動作するx8メモリデバイスは、84ビットのデータを転送できる（8本のデータ信号線xバーストを介して線毎に転送される8データビット）。この単純な例は、例示に過ぎず、限定的なものではないことが理解される。

10

【0110】

メモリデバイス840は、システム800用のメモリリソースを表す。1つの例において、各メモリデバイス840は、別個のメモリダイである。1つの例において、各メモリデバイス840は、デバイスまたはダイ毎に複数の（例えば2個の）チャンネルを有するインタフェースであり得る。各メモリデバイス840は、デバイスの実装により決定される帯域幅（例えば、x16もしくはx8または何らかの他のインタフェース帯域幅）を有するI/Oインタフェースロジック842を含む。I/Oインタフェースロジック842により、メモリデバイスは、メモリコントローラ820とインタフェースすることが可能になる。I/Oインタフェースロジック842は、ハードウェアインタフェースを含み得、メモリコントローラのI/O822に応じ得るが、メモリデバイスエンドにおけるものである。1つの例において、複数のメモリデバイス840は、同じコマンドおよびデータバスへ並列に接続される。別の例において、複数のメモリデバイス840は、同じコマンドバスへ並列に接続され、異なるデータバスに接続される。例えば、システム800は、並列に結合されている複数のメモリデバイス840を用いて構成され得る。各メモリデバイスは、コマンドに応答し、各内部のメモリリソース860にアクセスする。書き込み操作の場合、個々のメモリデバイス840は、データワード全体のうちのある部分を書き込むことができ、読み出し操作の場合、個々のメモリデバイス840は、データワード全体のある部分をフェッチできる。ワードの残りのビットは、他のメモリデバイスにより並列に提供または受信される。

20

30

【0111】

1つの例において、メモリデバイス840は、コンピューティングデバイスのマザーボードまたはホストシステムプラットフォーム（例えば、プロセッサ810が配置されるPCB（プリント回路基板））上に直接配置される。1つの例において、メモリデバイス840は、メモリモジュール870へ編成され得る。1つの例において、メモリモジュール870は、デュアルインラインメモリモジュール（DIMM）を表す。1つの例において、メモリモジュール870は、アクセスまたは制御回路のうちの少なくとも一部分を共有するための複数のメモリデバイスの他の編成を表す。当該回路は、ホストシステムプラットフォームとは別個の回路、別個のデバイスまたは別個の基板であり得る。メモリモジュール870は、複数のメモリデバイス840を含むことができ、メモリモジュールは、それらに配置されて含まれるメモリデバイスへの複数の別個のチャンネルのためのサポートを含むことができる。別の例において、メモリデバイス840は、例えばマルチチップモジュール（MCM）、パッケージオンパッケージ、スルーシリコンビア（TSV）もしくは他の技術またはそれらの組み合わせなどの技術により、メモリコントローラ820と同じパッケージへ組み込まれてよい。同様に、1つの例において、複数のメモリデバイス840は、メモリモジュール870へ組み込まれてよく、メモリモジュール870自体が、メモリコントローラ820と同じパッケージへ組み込まれ得る。これらの実装および他の実装の場合、メモリコントローラ820はホストプロセッサ810の一部であり得ることが理解される。

40

【0112】

50

メモリデバイス 840 はそれぞれ、1 つまたは複数のメモリアレイ 860 を含む。メモリアレイ 860 は、アドレス指定可能なメモリ位置またはデータの記憶位置を表す。通常、メモリアレイ 860 は、データの行として管理され、ワード線（行）およびビット線（行内の個々のビット）の制御を介してアクセスされる。メモリアレイ 860 は、メモリの別個のチャネル、ランク、およびバンクとして編成され得る。チャネルは、メモリデバイス 840 内の記憶位置への独立の制御経路を指し得る。ランクは、並列に並んだ複数のメモリデバイスにわたる共通の位置（例えば、異なるデバイス内の同じ行アドレス）を指し得る。バンクは、メモリデバイス 840 内のメモリ位置のサブアレイを指し得る。1 つの例において、メモリのバンクは、サブバンク用の共有回路（例えば、ドライバ、信号線、制御ロジック）のうちの少なくとも一部分を有するサブバンクへ分割される。これにより、別個のアドレス指定およびアクセスが可能になる。チャネル、ランク、バンク、サブバンク、バンク群、またはメモリ位置の他の編成、および当該編成の組み合わせは、それらの物理リソースへの応用において重複し得ることが理解される。例えば、同じ物理メモリ位置は、ランクに属することもあり得る特定のバンクとしての特定のチャネルを介してアクセスされ得る。故に、メモリリソースの編成は、排他的ではなく、包括的に理解される。

【0113】

1 つの例において、メモリデバイス 840 は、1 つまたは複数のレジスタ 844 を含む。レジスタ 844 は、メモリデバイスの操作の構成または設定を提供する 1 つまたは複数のストレージデバイスまたは記憶位置を表す。1 つの例において、レジスタ 844 は、制御操作または管理操作の一部としてメモリコントローラ 820 によりアクセスされるデータを格納するためのメモリデバイス 840 の記憶位置を提供できる。1 つの例において、レジスタ 844 は、1 つまたは複数のモードレジスタを含む。1 つの例において、レジスタ 844 は、1 つまたは複数の多目的レジスタを含む。レジスタ 844 内の位置の構成により、メモリデバイス 840 は、異なる「モード」で動作するように構成され得る。コマンド情報により、モードに基づいて、メモリデバイス 840 内の異なる操作がトリガされ得る。加えて、または代替的に、異なるモードにより、モードに応じて、アドレス情報または他の信号線から異なる操作もトリガされ得る。レジスタ 844 の設定は、I/O 設定の構成（例えば、タイミング、終端または ODT（オンダイ終端）846、ドライバ構成または他の I/O 設定）を示し得る。

【0114】

1 つの例において、メモリデバイス 840 は、I/O 842 に関連付けられたインタフェースハードウェアの一部としての ODT 846 を含む。ODT 846 は、上述のように構成され得るとともに、指定された信号線へのインタフェースに適用されるインピーダンスの設定を提供し得る。1 つの例において、ODT 846 は、DQ 信号線に適用される。1 つの例において、ODT 846 は、コマンド信号線に適用される。1 つの例において、ODT 846 は、アドレス信号線に適用される。1 つの例において、ODT 846 は、前述のものの任意の組み合わせに適用され得る。ODT 設定は、メモリデバイスがアクセス操作の選択されたターゲットであるか、または非ターゲットデバイスであるかに基づいて変更され得る。ODT 846 設定は、終端線上のシグナリングのタイミングおよび反映に影響を及ぼし得る。ODT 846 に対する慎重な制御により、適用されるインピーダンスおよびロードの一致度が向上した、より高速な操作が可能になり得る。ODT 846 は、I/O インタフェース 842、822（例えば、DQ 線用の ODT または CA 線用の ODT）の特定の信号線に適用され得るが、必ずしも全ての信号線に適用されるわけではない。

【0115】

メモリデバイス 840 は、メモリデバイス内の内部操作を制御するためのメモリデバイス内の制御ロジックを表すコントローラ 850 を含む。例えば、コントローラ 850 は、メモリコントローラ 820 により送信されるコマンドをデコードするとともに、コマンドを実行しまたは満足させるための内部操作を生成する。コントローラ 850 は、内部コントローラと称され得、ホストのメモリコントローラ 820 とは別個のものである。コントローラ 850 は、どのモードが選択されているかをレジスタ 844 に基づいて決定でき、

メモリリソース 860 へのアクセスのための操作または他の操作の内部での実行を、選択されたモードに基づいて構成できる。コントローラ 850 は、メモリデバイス 840 内のビットのルーティングを制御するための制御信号を生成して、選択されたモード用の適切なインタフェースを提供し、コマンドを適切なメモリ位置またはアドレスへ送る。コントローラ 850 は、コマンド信号線およびアドレス信号線上で受信されるコマンドエンコーディングをデコードできるコマンドロジック 852 を含む。故に、コマンドロジック 852 は、コマンドデコーダであり得るか、またはコマンドデコーダを含み得る。コマンドロジック 852 を用いることで、メモリデバイスは、コマンドを識別でき、要求されたコマンドを実行するための内部操作を生成できる。

【0116】

メモリコントローラ 820 を再び参照すると、メモリコントローラ 820 は、メモリデバイス 840 へ送信すべきコマンドを生成するためのロジックまたは回路を表すコマンド (CMD) ロジック 824 を含む。コマンドの生成は、スケジューリング前のコマンド、またはキューに入れられたコマンドの送信の準備を整えることを指し得る。概して、メモリサブシステム内のシグナリングは、メモリデバイスがコマンドを実行すべき 1 つまたは複数のメモリ位置を指示または選択するためのコマンド内またはコマンドに付随するアドレス情報を含む。メモリデバイス 840 のトランザクションのスケジューリングに応答して、メモリコントローラ 820 は、メモリデバイス 840 にコマンドを実行させるために E/O 822 を介してコマンドを発行できる。1 つの例において、メモリデバイス 840 のコントローラ 850 は、メモリコントローラ 820 から E/O 842 を介して受信したコマンドおよびアドレス情報を受信およびデコードする。受信したコマンドおよびアドレス情報に基づいて、コントローラ 850 は、メモリデバイス 840 内のロジックおよび回路の操作のタイミングを制御することでコマンドを実行できる。コントローラ 850 は、タイミング要件およびシグナリング要件など、メモリデバイス 840 内の規格または仕様への準拠を担う。メモリコントローラ 820 は、アクセスのスケジューリングおよび制御により、規格または仕様への準拠を実装できる。

【0117】

メモリコントローラ 820 は、メモリデバイス 840 へ送信すべきトランザクションを生成および命令するためのロジックまたは回路を表すスケジューラ 830 を含む。1 つの観点から、メモリコントローラ 820 の主な機能は、メモリデバイス 840 へのメモリアクセスおよび他のトランザクションをスケジューリングすることとすることができる。そのようなスケジューリングは、プロセッサ 810 によるデータについての要求を実装するために、かつ、(例えば、リフレッシュに関連するコマンドの利用などで) データの完全性を維持するためにトランザクション自体を生成することを含み得る。トランザクションは、1 つまたは複数のコマンドを含み得るので、クロックサイクルまたは単位間隔などの 1 つまたは複数のタイミングサイクルでのコマンドもしくはデータまたはそれらの両方の転送をもたらし得る。トランザクションは、読み出しもしくは書き込みもしくは関連するコマンドまたはそれらの組み合わせなどのアクセスのためのものであり得る。他のトランザクションは、構成、設定、データ完全性についてのメモリ管理コマンド、もしくは他のコマンドまたはそれらの組み合わせを含み得る。

【0118】

メモリコントローラ 820 は通常、トランザクションの選択および順序付けを可能にしてシステム 800 の性能を向上させるためのスケジューラ 830 などのロジックを含む。故に、メモリコントローラ 820 は、未処理のトランザクションのうちのどれがどの順序でメモリデバイス 840 へ送信されるべきかを選択できる。これは通常、単純な先入れ先出しアルゴリズムよりもはるかに複雑なロジックを用いて実現される。メモリコントローラ 820 は、メモリデバイス 840 へのトランザクションの伝送を管理するとともに、トランザクションに関連するタイミングを管理する。1 つの例において、トランザクションには確定的なタイミングがある。当該タイミングは、メモリコントローラ 820 により管理され得るとともに、トランザクションをスケジューラ 830 でどのようにスケジューリ

10

20

30

40

50

ングするか決定において用いられ得る。

【0119】

1つの例において、メモリコントローラ820は、リフレッシュ（REF）ロジック826を含む。リフレッシュロジック826は、揮発性であるメモリリソースに用いられるが、確定的な状態を保持するためにリフレッシュされる必要があり得る。1つの例において、リフレッシュロジック826は、リフレッシュの位置と、実行すべきリフレッシュのタイプとを示す。リフレッシュロジック826は、メモリデバイス840内でのセルフリフレッシュをトリガできるか、もしくはリフレッシュコマンドを送信することにより外部リフレッシュ（自動リフレッシュコマンドと称され得る）を実行できるか、またはそれらの組み合わせを行うことができる。1つの例において、メモリデバイス840内のコントローラ850は、メモリデバイス840内でリフレッシュを適用するためのリフレッシュロジック854を含む。1つの例において、リフレッシュロジック854は、内部操作を生成して、メモリコントローラ820から受信された外部リフレッシュに従ってリフレッシュを実行する。リフレッシュロジック854は、リフレッシュがメモリデバイス840に向けられているか、またコマンドに応答してどのメモリリソース860をリフレッシュすべきかを決定することができる。

【0120】

図9は、選択的オンダイECCバイパスを実装できるコンピューティングシステムの一例のブロック図である。システム900は、本明細書における任意の例に係るコンピューティングデバイスを表し、ラップトップコンピュータ、デスクトップコンピュータ、タブレットコンピュータ、サーバ、ゲーミングシステムもしくはエンタテインメント制御システム、組み込みコンピューティングデバイスまたは他の電子デバイスであり得る。システム900は、システム100またはシステム200と互換性のあるシステムを組み込み得るシステムの1つの例である。

【0121】

1つの例において、メモリ930は、メモリコントローラ922にデータを返す前に、読み出しデータにECCを実行するためのオンダイECC992を備える。1つの例において、メモリ930は、ECC992を選択的にバイパスするためのバイパス994を備える。内部ECCのバイパスは、本明細書の任意の説明に従うものであることができ、メモリ930は、リトライコマンドに応答して読み出しを実行する。メモリ930は、リトライコマンドに基づいて、データを返す前に、データにECCを実行する代わりに、メモリアレイからの生データを返す。

【0122】

システム900は、プロセッサ910を含む。プロセッサ910は、システム900のために命令の処理または実行を提供するための任意のタイプのマイクロプロセッサ、中央処理ユニット（CPU）、グラフィックス処理ユニット（GPU）、処理コアもしくは他の処理ハードウェアまたはそれらの組み合わせを含み得る。プロセッサ910は、システム900のオペレーション全体を制御し、1つまたは複数のプログラマブル汎用または特定用途向けマイクロプロセッサ、デジタル信号プロセッサ（DSP）、プログラマブルコントローラ、特定用途向け集積回路（ASIC）、プログラマブルロジックデバイス（PLD）またはそのようなデバイスの組み合わせであり得るか、またはそれらを含み得る。

【0123】

1つの例において、システム900は、プロセッサ910に結合されているインタフェース912を含む。インタフェース912は、メモリサブシステム920またはグラフィックスインタフェースコンポーネント940など、より高帯域幅の接続を必要とする、システムコンポーネント用のより高速なインタフェースまたは高スループットインタフェースを表し得る。インタフェース912は、スタンドアロンコンポーネントであり得るか、またはプロセッサダイに統合され得るインタフェース回路を表す。インタフェース912は、回路としてプロセッサダイ上に統合され得るか、またはコンポーネントとしてシステムオンチップに統合され得る。存在する場合、グラフィックスインタフェース940は、

システム 900 のユーザに視覚表示を提供するためのグラフィックスコンポーネントにインタフェースする。グラフィックスインタフェース 940 は、スタンドアロンコンポーネントであり得るか、またはプロセッサダイもしくはシステムオンチップに統合され得る。1つの例において、グラフィックスインタフェース 940 は、ユーザに出力を提供する高精細度 (HD) ディスプレイまたは超高精細度 (UHD) ディスプレイを駆動できる。1つの例において、ディスプレイは、タッチスクリーンディスプレイを含み得る。1つの例において、グラフィックスインタフェース 940 は、メモリ 930 に格納されたデータに基づいて、もしくはプロセッサ 910 により実行された操作に基づいて、またはそれらの両方に基づいて表示を生成する。

【0124】

メモリサブシステム 920 は、システム 900 のメインメモリを表し、プロセッサ 910 により実行されるコードまたはルーチンの実行に用いられるデータ値のための記憶領域を提供する。メモリサブシステム 920 は、リードオンリメモリ (ROM)、フラッシュメモリ、DRAM などの 1つまたは複数の種類のランダムアクセスメモリ (RAM)、3DXP (3次元クロスポイント) もしくは他のメモリデバイスまたはそのようなデバイスの組み合わせなど、1つまたは複数のメモリデバイス 930 を含み得る。メモリ 930 は、とりわけ、システム 900 内での命令の実行用のソフトウェアプラットフォームを提供するためのオペレーティングシステム (OS) 932 を格納およびホストする。加えて、アプリケーション 934 は、メモリ 930 から OS 932 のソフトウェアプラットフォーム上で実行され得る。アプリケーション 934 は、プログラムを表す。当該プログラムは、1つまたは複数の機能の実行をするためのそれらの独自の動作ロジックを有する。処理 936 は、OS 932 もしくは 1つまたは複数のアプリケーション 934 またはそれらの組み合わせに補助機能を提供するエージェントまたはルーチンを表す。OS 932、アプリケーション 934 および処理 936 は、システム 900 用の機能を提供するためのソフトウェアロジックを提供する。1つの例において、メモリサブシステム 920 は、コマンドを生成してメモリ 930 へ発行するメモリコントローラであるメモリコントローラ 922 を含む。メモリコントローラ 922 は、プロセッサ 910 の物理的部分またはインタフェース 912 の物理的部分であり得ることが理解される。例えば、メモリコントローラ 922 は、例えばプロセッサダイまたはシステムオンチップに統合されるなど、プロセッサ 910 とともに回路に統合される統合メモリコントローラであり得る。

【0125】

具体的には示されていないが、システム 900 は、メモリバス、グラフィックスバス、インタフェースバスまたは他のものなど、デバイス間の 1つまたは複数のバスまたはバスシステムを含み得ることが理解される。バスまたは他の信号線は、コンポーネントを互いに通信可能にまたは電氣的に結合できるか、またはコンポーネントを通信可能にかつ電氣的に結合できる。バスは、物理通信線、ポイントツーポイント接続、ブリッジ、アダプタ、コントローラもしくは他の回路またはそれらの組み合わせを含み得る。バスは、例えば、システムバス、ペリフェラルコンポーネント相互接続 (PCI) バス、ハイパートランスポートアーキテクチャバスもしくは業界標準アーキテクチャ (ISA) バス、スモールコンピュータシステムインタフェース (SCSI) バス、ユニバーサルシリアルバス (USB) もしくは他のバスのうちの 1つまたは複数またはそれらの組み合わせを含み得る。

【0126】

1つの例において、システム 900 は、インタフェース 912 に結合され得るインタフェース 914 を含む。インタフェース 914 は、インタフェース 912 よりも低速なインタフェースであり得る。1つの例において、インタフェース 914 は、スタンドアロンコンポーネントおよび集積回路を含み得るインタフェース回路を表す。1つの例において、複数のユーザインタフェースコンポーネントもしくはペリフェラルコンポーネントまたはそれらの両方がインタフェース 914 に結合する。ネットワークインタフェース 950 は、1つまたは複数のネットワークを介してリモートデバイス (例えば、サーバまたは他のコンピューティングデバイス) と通信する機能をシステム 900 に提供する。ネットワー

10

20

30

40

50

クインタフェース 950 は、イーサネット（登録商標）アダプタ、無線相互接続コンポーネント、セルラネットワーク相互接続コンポーネント、USB（ユニバーサルシリアルバス）または他の有線もしくは無線規格ベースインタフェースもしくはプロプライエタリインタフェースを含み得る。ネットワークインタフェース 950 は、リモートデバイスとデータを交換できる。当該交換は、メモリに格納されたデータの送信またはメモリに格納されたデータの受信を含み得る。

【0127】

1つの例において、システム 900 は、1つまたは複数の入力／出力（I／O）インタフェース 960 を含む。I／Oインタフェース 960 は、1つまたは複数のインタフェースコンポーネントを含み得る。当該インタフェースコンポーネントを通じて、ユーザは、システム 900 とインタラクトする（例えば、音声、英数字、触覚／タッチまたは他のインタフェース）。周辺インタフェース 970 は、具体的には上述されていない任意のハードウェアインタフェースを含み得る。周辺は概して、システム 900 へ従属的に接続するデバイスを指す。従属的な接続は、システム 900 がソフトウェアプラットフォームもしくはハードウェアプラットフォームを提供する接続であるか、または、当該プラットフォーム上で操作が実行され、かつ当該プラットフォームを用いてユーザがインタラクトする接続である。

【0128】

1つの例において、システム 900 は、データを不揮発性方式で格納するためのストレージサブシステム 980 を含む。1つの例において、特定のシステム実装では、ストレージ 980 の少なくとも特定のコンポーネントがメモリサブシステム 920 のコンポーネントと重複し得る。ストレージサブシステム 980 は、ストレージデバイス 984 を含む。ストレージデバイス 984 は、1つまたは複数の磁気ディスク、ソリッドステートディスク、3DXP ディスクもしくは光ベースディスクまたはそれらの組み合わせなど、大量のデータを不揮発性方式で格納するための任意の従来の媒体であり得るか、または当該媒体を含み得る。ストレージ 984 は、コードまたは命令およびデータ 986 を永続的な状態で保持する（すなわち、システム 900 への電力が遮断されても値が保持される）。ストレージ 984 は一般に「メモリ」と見なされ得るが、メモリ 930 は通常、命令をプロセッサ 910 に提供するための実行メモリまたは動作メモリである。ストレージ 984 は不揮発性であるが、メモリ 930 は揮発性メモリを含み得る（すなわち、システム 900 への電力が遮断された場合、データの値または状態は不定になる）。1つの例において、ストレージサブシステム 980 は、ストレージ 984 とインタフェースするためのコントローラ 982 を含む。1つの例において、コントローラ 982 は、インタフェース 914 またはプロセッサ 910 の物理的部分であるか、またはプロセッサ 910 およびインタフェース 914 の両方に回路またはロジックを含み得る。

【0129】

電源 902 は、システム 900 のコンポーネントに電力を提供する。より具体的には、電源 902 は通常、システム 900 内の1つまたは複数の電力供給装置 904 とインタフェースして、システム 900 のコンポーネントに電力を提供する。1つの例において、電力供給装置 904 は、壁のコンセントへ差し込むための AC-DC（交流-直流）アダプタを含む。そのような AC 電源は、再生可能エネルギー（例えば、太陽光発電）電源 902 であり得る。1つの例において、電源 902 は、外部 AC-DC 変換器などの DC 電源を含む。1つの例において、電源 902 または電力供給装置 904 は、充電磁場への近接を介して充電するための無線充電ハードウェアを含む。1つの例において、電源 902 は、内部バッテリーまたは燃料電池電源を含み得る。

【0130】

図 10 は、選択的オンダイ ECC バイパスを実装できるモバイルデバイスの一例のブロック図である。システム 1000 は、コンピューティングタブレット、携帯電話もしくはスマートフォン、ウェアラブルコンピューティングデバイスもしくは他のモバイルデバイスまたは組み込みコンピューティングデバイスなどのモバイルコンピューティングデバイ

10

20

30

40

50

スを表す。これらのコンポーネントのうち特定のものが一般的に示されているが、そのようなデバイスの全てのコンポーネントがシステム 1000 内に示されているわけではないことが理解される。システム 1000 は、システム 100 またはシステム 200 と互換性のあるシステムを組み込み得るシステムの 1 つの例である。

【0131】

1 つの例において、メモリ 1062 は、メモリコントローラ 1064 に読み出しデータを返す前に、読み出しデータに ECC を実行するためのオンダイ ECC 1092 を備える。1 つの例において、メモリ 1062 は、ECC 1092 を選択的にバイパスするためのバイパス 1094 を備える。内部 ECC のバイパスは、本明細書の任意の説明に従うものであることができ、メモリ 1062 がリトライコマンドに応答して読み出しを実行する。メモリ 1062 は、リトライコマンドに基づいて、データを返す前に、データに ECC を実行する代わりに、メモリアレイからの生データを返す。

【0132】

システム 1000 は、システム 1000 の主な処理操作を実行するプロセッサ 1010 を含む。プロセッサ 1010 は、マイクロプロセッサ、アプリケーションプロセッサ、マイクロコントローラ、プログラマブルロジックデバイスまたは他の処理手段などの 1 つまたは複数の物理デバイスを含み得る。プロセッサ 1010 により実行される処理操作は、アプリケーションおよびデバイス機能が実行されるオペレーティングプラットフォームまたはオペレーティングシステムの実行を含む。処理操作は、人間のユーザもしくは他のデバイスとの I/O（入力／出力）に関連する操作、電力管理に関連する操作、別のデバイスへのシステム 1000 の接続に関連する操作またはそれらの組み合わせを含む。処理操作は、音声 I/O、表示 I/O もしくは他のインタフェースまたはそれらの組み合わせに関連する操作も含み得る。プロセッサ 1010 は、メモリに格納されたデータを実行できる。プロセッサ 1010 は、メモリに格納されたデータを書き込むことができるか、または編集できる。

【0133】

1 つの例において、システム 1000 は、1 つまたは複数のセンサ 1012 を含む。センサ 1012 は、外部センサへの埋め込みセンサもしくはインタフェースまたはそれらの組み合わせを表す。センサ 1012 により、システム 1000 は、システム 1000 が実装される環境またはデバイスの 1 つまたは複数の条件を監視または検出することが可能になる。センサ 1012 は、環境センサ（温度センサ、動き検出器、光検出器、カメラ、化学センサ（例えば、一酸化炭素センサ、二酸化炭素センサもしくは他の化学センサ）など）、圧力センサ、加速度計、ジャイロスコープ、医療センサもしくは生理機能センサ（例えば、バイオセンサ、心拍数モニタもしくは生理学的な属性を検出するための他のセンサ）もしくは他のセンサまたはそれらの組み合わせを含み得る。センサ 1012 は、指紋認識システム、顔検出もしくは認識システムまたはユーザの特徴を検出または認識する他のシステムなどの生体認証システム用のセンサも含み得る。センサ 1012 は、幅広く理解されるべきであり、システム 1000 とともに実装され得る多くの異なるタイプのセンサに対する限定として理解されるべきではない。1 つの例において、1 つまたは複数のセンサ 1012 は、プロセッサ 1010 に統合されたフロントエンド回路を介して、プロセッサ 1010 に結合する。1 つの例において、1 つまたは複数のセンサ 1012 は、システム 1000 の別のコンポーネントを介して、プロセッサ 1010 に結合する。

【0134】

1 つの例において、システム 1000 は、ハードウェア（例えば、音声ハードウェアおよび音声回路）と、コンピューティングデバイスへの音声機能の提供に関連するソフトウェア（例えば、ドライバ、コーデック）コンポーネントとを表すオーディオサブシステム 1020 を含む。音声機能は、スピーカ出力またはヘッドフォン出力およびマイク入力を含み得る。そのような機能のためのデバイスは、システム 1000 へ統合され得るか、またはシステム 1000 に接続され得る。1 つの例において、ユーザは、プロセッサ 1010 により受信および処理される音声コマンドを提供することにより、システム 1000 と

インタラクトする。

【0135】

ディスプレイサブシステム1030は、ユーザに提示するための視覚表示を提供するハードウェア（例えば、ディスプレイデバイス）コンポーネントおよびソフトウェアコンポーネント（例えば、ドライバ）を表す。1つの例において、ディスプレイは、ユーザがコンピューティングデバイスとインタラクトするための触覚コンポーネントまたはタッチスクリーン要素を含む。ディスプレイサブシステム1030は、ユーザに表示を提供するために用いられる特定のスクリーンまたはハードウェアデバイスを含むディスプレイインタフェース1032を含む。1つの例において、ディスプレイインタフェース1032は、表示に関連する少なくともいくつかの処理を実行するための、プロセッサ1010（グラフィックスプロセッサなど）とは別個のロジックを含む。1つの例において、ディスプレイサブシステム1030は、ユーザへの出力および入力の両方を提供するタッチスクリーンデバイスを含む。1つの例において、ディスプレイサブシステム1030は、ユーザに出力を提供する高精細度（HD）ディスプレイまたは超高精細度（UHD）ディスプレイを含む。1つの例において、ディスプレイサブシステムは、タッチスクリーンディスプレイを含むか、またはタッチスクリーンディスプレイを駆動する。1つの例において、ディスプレイサブシステム1030は、メモリに格納されたデータに基づいて、もしくはプロセッサ1010により実行される操作に基づいて、またはそれらの両方に基づいて表示情報を生成する。

10

【0136】

I/Oコントローラ1040は、ユーザとのインタラクションに関連するハードウェアデバイスおよびソフトウェアコンポーネントを表す。I/Oコントローラ1040は、オーディオサブシステム1020もしくはディスプレイサブシステム1030またはそれらの両方の一部であるハードウェアを管理するように動作し得る。加えて、I/Oコントローラ1040は、システム1000に接続する追加のデバイスのための接続ポイントを示す。当該接続ポイントを通じて、ユーザは、システムとインタラクトし得る。例えば、システム1000に取り付けられ得るデバイスは、マイクデバイス、スピーカシステムもしくはステレオシステム、ビデオシステムもしくは他のディスプレイデバイス、キーボードデバイスもしくはキーパッドデバイス、ボタン/スイッチ、もしくは、カードリーダーなどの特定の応用品とともに用いるための他のI/Oデバイス、または他のデバイスを含み得る。

20

30

【0137】

上述のように、I/Oコントローラ1040は、オーディオサブシステム1020もしくはディスプレイサブシステム1030またはそれらの両方とインタラクトできる。例えば、マイクまたは他の音声デバイスを通じた入力により、システム1000の1つまたは複数のアプリケーションまたは機能についての入力またはコマンドが提供され得る。加えて、音声出力は、ディスプレイ出力の代わりに、またはディスプレイ出力に加えて提供され得る。別の例において、ディスプレイサブシステムがタッチスクリーンを含む場合、ディスプレイデバイスは、少なくとも部分的にI/Oコントローラ1040により管理され得る入力デバイスとしても機能する。I/Oコントローラ1040により管理されるI/O機能を提供するための追加のボタンまたはスイッチもシステム1000上に存在し得る。

40

【0138】

1つの例において、I/Oコントローラ1040は、加速度計、カメラ、光センサもしくは他の環境センサ、ジャイロスコープ、全地球測位システム（GPS）もしくはシステム1000に含まれ得る他のハードウェアまたはセンサ1012などのデバイスを管理する。入力は、直接的なユーザインタラクションの一部であり得るだけでなく、その操作（ノイズのフィルタリング、輝度検出のための表示の調節、カメラ用のフラッシュの適用または他の特徴など）に影響を及ぼすようにシステムに環境入力を提供することであり得る。

【0139】

1つの例において、システム1000は、バッテリーの電力使用量と、バッテリーの充電と

50

省電力動作に関連する機能とを管理する電力管理 1050 を含む。電力管理 1050 は、システム 1000 のコンポーネントに電力を提供する電源 1052 からの電力を管理する。1つの例において、電源 1052 は、壁のコンセントへ差し込むための A C - D C (交流 - 直流) アダプタを含む。そのような A C 電源は、再生可能エネルギー (例えば、太陽光発電、動きベース電力) であり得る。1つの例において、電源 1052 は、外部 A C - D C 変換器などの D C 電源により提供され得る D C 電力のみを含む。1つの例において、電源 1052 は、充電磁場への近接を介して充電するための無線充電ハードウェアを含む。1つの例において、電源 1052 は、内部バッテリーまたは燃料電池電源を含み得る。

【0140】

メモリサブシステム 1060 は、情報をシステム 1000 に格納するためのメモリデバイス 1062 を含む。メモリサブシステム 1060 は、不揮発性 (メモリデバイスへの電力が遮断された場合に状態が変わらない) メモリデバイスもしくは揮発性 (メモリデバイスへの電力が遮断された場合に状態が不定になる) メモリデバイスまたはそれらの組み合わせを含み得る。メモリ 1060 は、アプリケーションデータ、ユーザーデータ、音楽、写真、文書または他のデータだけでなく、システム 1000 のアプリケーションおよび機能の実行に関連するシステムデータ (長期間のものであるかまたは一時的なものであるかを問わない) を格納できる。1つの例において、メモリサブシステム 1060 は、メモリコントローラ 1064 (システム 1000 の制御部の一部とも見なされ得るとともに、プロセッサ 1010 の一部と潜在的に見なされ得る) を含む。メモリコントローラ 1064 は、コマンドを生成および発行してメモリデバイス 1062 へのアクセスを制御するスケジューラを含む。

【0141】

接続 1070 は、ハードウェアデバイス (例えば、無線コネクタもしくは有線コネクタおよび通信ハードウェアまたは有線ハードウェアと無線ハードウェアとの組み合わせ) およびシステム 1000 が外部デバイスと通信することを可能にするためのソフトウェアコンポーネント (例えば、ドライバ、プロトコルスタック) を含む。外部デバイスは、他のコンピューティングデバイス、無線アクセスポイントまたは基地局などの別個のデバイス、および、ヘッドセット、プリンタまたは他のデバイスなどの周辺機器であり得る。1つの例において、システム 1000 は、メモリへの格納のために、またはディスプレイデバイスでの表示のために、外部デバイスとデータを交換する。交換されるデータは、データの読み出し、書き込みまたは編集のためにメモリに格納されるデータまたはメモリに既に格納されているデータを含み得る。

【0142】

接続 1070 は、複数の異なるタイプの接続を含み得る。一般化するために、システム 1000 は、セルラ接続 1072 および無線接続 1074 とともに示される。セルラ接続 1072 は概して、例えば G S M (登録商標) (グローバルシステムフォーモバイルコミュニケーションズ) を介して提供されるなど無線キャリアにより提供されるセルラネットワーク接続もしくはその変形規格もしくは派生規格、もしくは C D M A (符号分割多重アクセス) もしくはその変形規格もしくは派生規格、T D M (時分割多重化) もしくはその変形規格もしくは派生規格、L T E (「4 G」とも称されるロングタームエボリューション)、5 G、または他のセルラサービス規格を指す。無線接続 1074 は、セルラではない無線接続を指し、パーソナルエリアネットワーク (B l u e t o o t h (登録商標) など)、ローカルエリアネットワーク (W i F i (登録商標) など) もしくはワイドエリアネットワーク (W i M A X など) もしくは他の無線通信またはそれらの組み合わせを含み得る。無線通信は、非固体媒体を通じた変調済み電磁放射線の使用を通じてデータを転送することを指す。有線通信は、固体通信媒体によって生じる。

【0143】

周辺接続 1080 は、ハードウェアインタフェースおよびコネクタならびに周辺接続を行うためのソフトウェアコンポーネント (例えば、ドライバ、プロトコルスタック) を含む。システム 1000 は、他のコンピューティングデバイスへの (「外への」 1082)

周辺デバイスであり、かつ、システム1000へ（「外から」1084）接続される周辺デバイスを備え得ることが理解される。システム1000は一般的に、システム1000上のコンテンツの管理（例えば、ダウンロード、アップロード、変更、同期）などの目的で他のコンピューティングデバイスに接続する「ドッキング」コネクタを有する。加えて、ドッキングコネクタにより、例えばオーディオビジュアルシステムまたは他のシステムに出力されるコンテンツをシステム1000が制御することを可能にする特定の周辺機器にシステム1000が接続することが可能になり得る。

【0144】

プロプライエタリドッキングコネクタまたは他のプロプライエタリ接続ハードウェアに加え、システム1000は、一般的なコネクタまたは規格ベースのコネクタを介して周辺接続1080を行い得る。一般的なタイプは、ユニバーサルシリアルバス（USB）コネクタ（多数の異なるハードウェアインタフェースのいずれかを含み得る）、Mini DisplayPort（MDP）、高精細度マルチメディアインタフェース（HDMI（登録商標））を含むDisplayPortまたは他のタイプを含み得る。

【0145】

概して本明細書の説明に関して、1つの例において、メモリコントローラは、メモリデバイスに結合されているハードウェアインタフェースと、メモリデバイスのためのリトライコマンドを生成するコマンドロジックと、を備え、リトライコマンドは、メモリデバイスのECC回路によるエラー検査訂正（ECC）を適用せずに読み出しデータを返すようにメモリデバイスをトリガするようになっている。

【0146】

1つの例において、コマンドロジックは、リトライコマンドを示すためのコマンドエンコーディングを有するコマンドを生成するようになっている。1つの例において、コマンドロジックは、リトライコマンドを示すために、同じアドレスに一連の連続する読み出しコマンドを生成するようになっている。1つの例において、コマンドロジックは、メモリデバイスのモードレジスタをリトライコマンドモードに設定するために、モードレジスタ書き込みコマンドを生成するようになっており、コマンドロジックは、リトライコマンドを示すために、メモリデバイスがリトライコマンドモードである間に、送信すべき読み出しコマンドを生成するようになっている。1つの例において、コマンドロジックは、前の読み出しコマンドに回答して受信されたエラーデータの受信に回答して、リトライコマンドを生成するようになっている。1つの例において、ハードウェアインタフェースは、前の読み出しコマンドに回答して、メモリデバイスから、検出された訂正不可能エラー（DUE）の指標を受信するようになっており、コマンドロジックは、メモリデバイスからのDUEの指標に回答してリトライコマンドを生成するようになっている。1つの例において、メモリデバイスは、ダブルデータレート（DDR）規格と互換性のあるシンクロナスダイナミックランダムアクセスメモリ（SDRAM）デバイスを備える。

【0147】

概して本明細書の説明に関して、1つの例において、メモリデバイスは、読み出しコマンドに回答して、読み出しデータを受信し、メモリデバイスの内部で読み出しデータにECCを適用し、訂正済みデータを返すエラー検査訂正（ECC）回路と、読み出しコマンドの後にリトライコマンドを受信するハードウェアインタフェースであって、リトライコマンドに回答して、ハードウェアインタフェースは、ECC回路によるECCが適用されていない読み出しデータを返すようになっている、ハードウェアインタフェースと、を備える。

【0148】

1つの例において、リトライコマンドは、リトライコマンドを示すためのコマンドエンコーディングを有するコマンドを含む。1つの例において、リトライコマンドは、他の介在コマンドを伴わない一連の読み出しコマンドを含む。1つの例において、メモリデバイスは、読み出しコマンドのアドレスを記憶するレジスタをさらに備え、メモリデバイスは、レジスタに記憶されている読み出しコマンドのアドレスが、他の介在コマンドを伴わず

10

20

30

40

50

に送信された後続の読み出しコマンドのアドレスと一致する場合、一連の読み出しコマンドをリトライコマンドとしてデコードするようになっている。1つの例において、メモリデバイスは、リトライコマンドモードを示すためのフィールドを記憶するモードレジスタをさらに備え、リトライコマンドは、リトライコマンドモードにおいて受信された読み出しコマンドを含む。1つの例において、ハードウェアインタフェースは、ECCを適用せずに読み出しデータを返すためにECC回路をバイパスするようになっている。1つの例において、ECC回路は、リトライコマンドを送信するようにホストをトリガするために、検出された訂正不可能エラー（DUE）をホストに示すようになっている。1つの例において、メモリデバイスは、ダブルデータレート（DDR）規格と互換性のあるシンクロナスダイナミックランダムアクセスメモリ（SDRAM）デバイスを含む。

10

【0149】

概して本明細書の説明に関して、1つの例において、システムは、読み出しコマンドに応答して読み出しデータにECCを内部で適用し、読み出しコマンドに応答して、訂正済みデータを返すエラー検査訂正（ECC）回路を有するメモリデバイスと、読み出しコマンドを生成し、読み出しコマンドの後にリトライコマンドを生成するコマンドロジックを有するメモリコントローラと、を備え、リトライコマンドは、メモリデバイスのECC回路によるECCを適用せずに読み出しデータを返すようにメモリデバイスをトリガするようになっている。

【0150】

1つの例において、リトライコマンドは、リトライコマンドを示すためのコマンドエンコーディングを有するコマンドを含む。1つの例において、リトライコマンドは、他の介在コマンドを伴わない一連の読み出しコマンドを含み、メモリデバイスは、一連の読み出しコマンドをリトライコマンドとしてデコードするようになっている。1つの例において、メモリデバイスは、リトライコマンドモードを示すためのフィールドを記憶するモードレジスタをさらに備え、メモリコントローラは、リトライコマンドを示すために、メモリデバイスがリトライコマンドモードである間に、読み出しコマンドを送信するようになっている。1つの例において、メモリデバイスは、ダブルデータレート（DDR）規格と互換性のあるシンクロナスダイナミックランダムアクセスメモリ（SDRAM）デバイスを含む。1つの例において、システムは、メモリコントローラに結合されているホストプロセッサデバイス、ホストプロセッサに通信可能に結合されているディスプレイ、ホストプロセッサに通信可能に結合されているネットワークインタフェース、またはシステムに電力を供給するバッテリーのうちの1つまたは複数をさらに含む。

20

30

【0151】

本明細書に示されるフロー図は、様々なプロセスアクションのシーケンスの例を提供する。フロー図は、ソフトウェアまたはファームウェアルーチン、ならびに物理的な操作によって実行される動作を示すことができる。フロー図は、ハードウェアおよび／またはソフトウェアで実装できる、有限状態機械（FSM）の状態の実装の一例を示すことができる。特定のシーケンスまたは順序で示されているが、特に明記しない限り、動作の順序は変更できる。したがって、図示された図表は単なる例として理解されるべきであり、プロセスは異なる順序で実行でき、いくつかの動作は並列に実行することができる。さらに、1つまたは複数のアクションを省略でき、したがって、すべての実装がすべてのアクションを実行するわけではない。

40

【0152】

様々な操作または機能が本明細書に記載されている限り、それらはソフトウェアコード、命令、コンフィグレーション、および／またはデータとして記載または定義することができる。コンテンツは、直接実行可能（「オブジェクト」または「実行可能」形式）、ソースコード、または差分コード（「デルタ」または「パッチ」コード）とすることができる。本明細書に記載されているソフトウェアコンテンツは、コンテンツが格納された製品を介して、または通信インタフェースを介してデータを送信するための通信インタフェースを動作させる方法を介して提供することができる。機械可読記憶媒体は、記述された機

50

能または操作を機械に実行させることができ、記録可能／記録不可能な媒体（例えば、読み出し専用メモリ（ROM）、ランダムアクセスメモリ（RAM）、磁気ディスク記憶媒体、光記憶媒体、フラッシュメモリデバイスなど）などの機械（例えば、コンピューティングデバイス、電子システムなど）によってアクセス可能な形式で情報を格納する任意の機構を含む。通信インタフェースは、メモリバスインタフェース、プロセッサバスインタフェース、インターネット接続、ディスクコントローラなど、別のデバイスと通信するためのハードワイヤード、無線、光などの媒体のいずれかとインタフェースする任意の機構を含む。通信インタフェースは、ソフトウェアコンテンツを記述するデータ信号を提供するために、コンフィグレーションパラメータを提供すること、および／または通信インタフェースを準備する信号を送信することによって構成することができる。通信インタフェースは、通信インタフェースに送信される1つまたは複数のコマンドまたは信号を介してアクセスすることができる。

10

【0153】

本明細書に記載される様々な構成要素は、記載される操作または機能を実行するための手段とすることができる。本書に記載される各構成要素は、ソフトウェア、ハードウェア、またはこれらの組み合わせを含む。構成要素は、ソフトウェアモジュール、ハードウェアモジュール、専用ハードウェア（例えば、特定用途向けハードウェア、特定用途向け集積回路（ASIC）、デジタル信号プロセッサ（DSP）など）、組み込みコントローラ、ハードワイヤード回路などとして実装することができる。

【0154】

20

本明細書に記載される内容に加えて、開示される内容および本発明の実装に対して、それらの範囲から逸脱することなく様々な修正を行うことができる。したがって、本明細書の説明および例は、限定的な意味ではなく、例示的な意味で解釈されるべきである。本発明の範囲は、以下の特許請求の範囲を参照することによってのみ判断されるべきである。

（項目1） メモリデバイスに結合されているハードウェアインタフェースと、

上記メモリデバイスのためのリトライコマンドを生成するコマンドロジックと、
を備え、

上記リトライコマンドは、上記メモリデバイスのECC回路によるエラー検査訂正（ECC）を適用せずに読み出しデータを返すように上記メモリデバイスをトリガするようになっている、メモリコントローラ。

30

（項目2） 上記コマンドロジックは、上記リトライコマンドを示すために、コマンドエンコーディングを有するコマンドを生成するようになっている、項目1に記載のメモリコントローラ。

（項目3） 上記コマンドロジックは、上記リトライコマンドを示すために、同じアドレスに一連の連続する読み出しコマンドを生成するようになっている、項目1に記載のメモリコントローラ。

（項目4） 上記コマンドロジックは、上記メモリデバイスのモードレジスタをリトライコマンドモードに設定するために、モードレジスタ書き込みコマンドを生成するようになっている、

上記コマンドロジックは、上記メモリデバイスが上記リトライコマンドを示すために上記リトライコマンドモードである間、送信すべき読み出しコマンドを生成するようになっている、項目1に記載のメモリコントローラ。

40

（項目5） 上記コマンドロジックは、前の読み出しコマンドに応答して受信されたエラーデータの受信に応答して、上記リトライコマンドを生成するようになっている、項目1に記載のメモリコントローラ。

（項目6） 上記ハードウェアインタフェースは、上記前の読み出しコマンドに応答して、上記メモリデバイスから、検出済み訂正不可能エラー（DUE）の指示を受信するようになっている、上記コマンドロジックは、上記メモリデバイスからの上記DUEの上記指示に応答して上記リトライコマンドを生成するようになっている、項目5に記載のメモリコントローラ。

50

(項目 7) 上記メモリデバイスは、ダブルデータレート (DDR) 規格と互換性のあるシンクロナスダイナミックランダムアクセスメモリ (SDRAM) デバイスを備える、項目 1 に記載のメモリコントローラ。

(項目 8) メモリデバイスであって、

読み出しデータを受信し、上記メモリデバイスの内部で上記読み出しデータに ECC を適用し、読み出しコマンドに応答して、訂正済みデータを返すエラー検査訂正 (ECC) 回路と、

上記読み出しコマンドの後にリトライコマンドを受信するハードウェアインタフェースであって、上記リトライコマンドに응答して、上記 ECC 回路による ECC を適用せずに上記読み出しデータを返すようになっている、ハードウェアインタフェースと、

を備える、メモリデバイス。

(項目 9) 上記リトライコマンドは、上記リトライコマンドを示すために、コマンドエンコーディングを有するコマンドを含む、項目 8 に記載のメモリデバイス。

(項目 10) 上記リトライコマンドは、他の介在コマンドを伴わない一連の読み出しコマンドを含む、項目 8 に記載のメモリデバイス。

(項目 11) 上記読み出しコマンドのアドレスを記憶するレジスタをさらに備え、上記メモリデバイスは、上記レジスタに記憶されている上記読み出しコマンドの上記アドレスが他の介在コマンドを伴わずに送信された後続の読み出しコマンドのアドレスと一致する場合、上記一連の読み出しコマンドを上記リトライコマンドとしてデコードするようになっている、項目 10 に記載のメモリデバイス。

(項目 12) リトライコマンドモードを示すためのフィールドを記憶するモードレジスタをさらに備え、

上記リトライコマンドは、上記リトライコマンドモードにおいて受信された読み出しコマンドを含む、項目 8 に記載のメモリデバイス。

(項目 13) 上記ハードウェアインタフェースは、ECC を適用せずに上記読み出しデータを返すために上記 ECC 回路をバイパスするようになっている、項目 8 に記載のメモリデバイス。

(項目 14) 上記 ECC 回路は、上記リトライコマンドを送信するようにホストをトリガするために、検出済み訂正不可能エラー (DUE) を上記ホストに示すようになっている、項目 8 に記載のメモリデバイス。

(項目 15) 上記メモリデバイスは、ダブルデータレート (DDR) 規格と互換性のあるシンクロナスダイナミックランダムアクセスメモリ (SDRAM) デバイスを含む、項目 8 に記載のメモリデバイス。

(項目 16) 読み出しコマンドに응答して読み出しデータに ECC を内部で適用し、上記読み出しコマンドに응答して、訂正済みデータを返すエラー検査訂正 (ECC) 回路を有するメモリデバイスと、

上記読み出しコマンドを生成し、上記読み出しコマンドの後にリトライコマンドを生成するコマンドロジックを有するメモリコントローラと、

を備え、

上記リトライコマンドは、上記メモリデバイスの上記 ECC 回路による ECC を適用せずに読み出しデータを返すように上記メモリデバイスをトリガするようになっている、データを記憶するシステム。

(項目 17) 上記リトライコマンドは、上記リトライコマンドを示すために、コマンドエンコーディングを有するコマンドを含む、項目 16 に記載のシステム。

(項目 18) 上記リトライコマンドは、他の介在コマンドを伴わない一連の読み出しコマンドを含み、上記メモリデバイスは、上記一連の読み出しコマンドを上記リトライコマンドとしてデコードするようになっている、項目 16 に記載のシステム。

(項目 19) 上記メモリデバイスは、リトライコマンドモードを示すためのフィールドを記憶するモードレジスタをさらに備え、

上記メモリコントローラは、上記メモリデバイスが上記リトライコマンドを示すように

10

20

30

40

50

上記リトライコマンドモードである間、読み出しコマンドを送信するようになっている、項目 16 に記載のシステム。

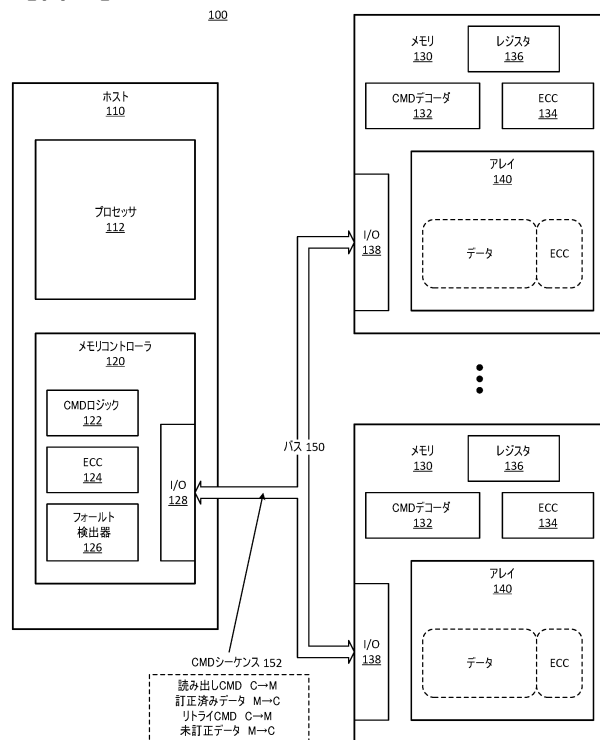
(項目 20) 上記メモリデバイスは、ダブルデータレート (DDR) 規格と互換性のあるシンクロナスダイナミックランダムアクセスメモリ (SDRAM) デバイスを含む、項目 16 に記載のシステム。

(項目 21) 上記メモリコントローラに結合されているホストプロセッサデバイス、ホストプロセッサに通信可能に結合されているディスプレイ、ホストプロセッサに通信可能に結合されているネットワークインタフェース、または、上記システムに電力を供給するバッテリー、
 のうちの 1 つまたは複数さをさらに備える、項目 16 に記載のシステム。

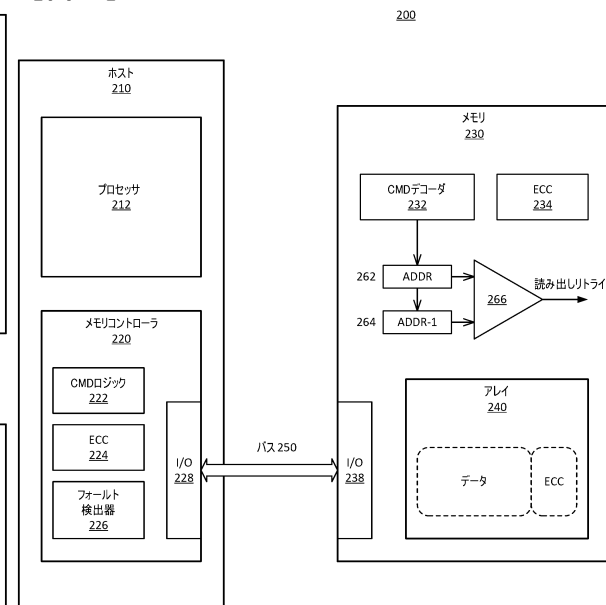
10

【図面】

【図 1】



【図 2】



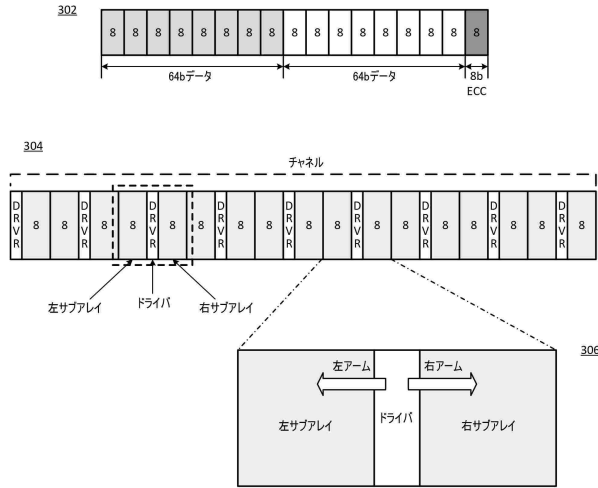
20

30

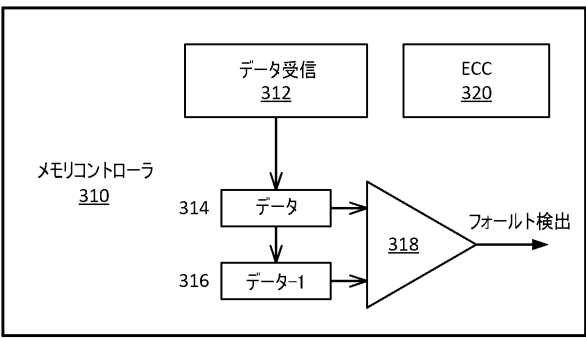
40

50

【図 3 A】

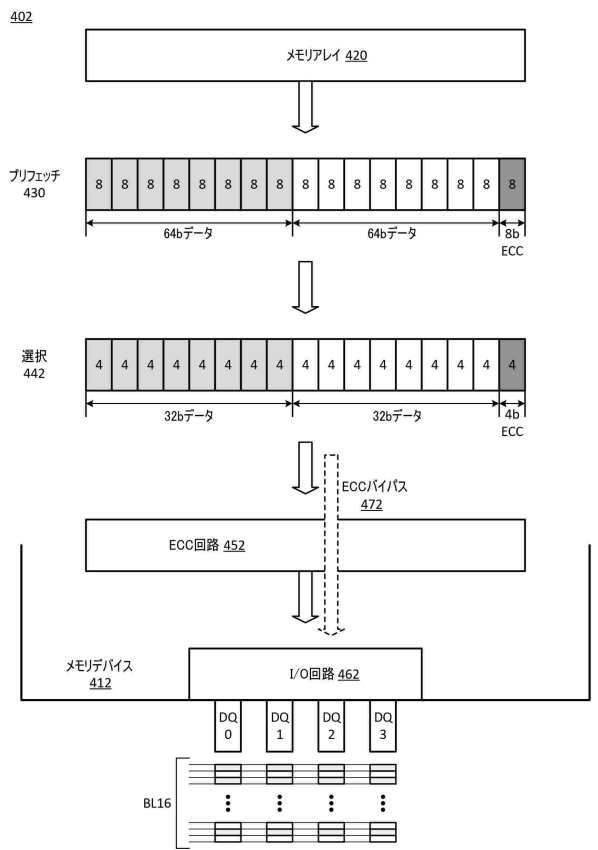


【図 3 B】

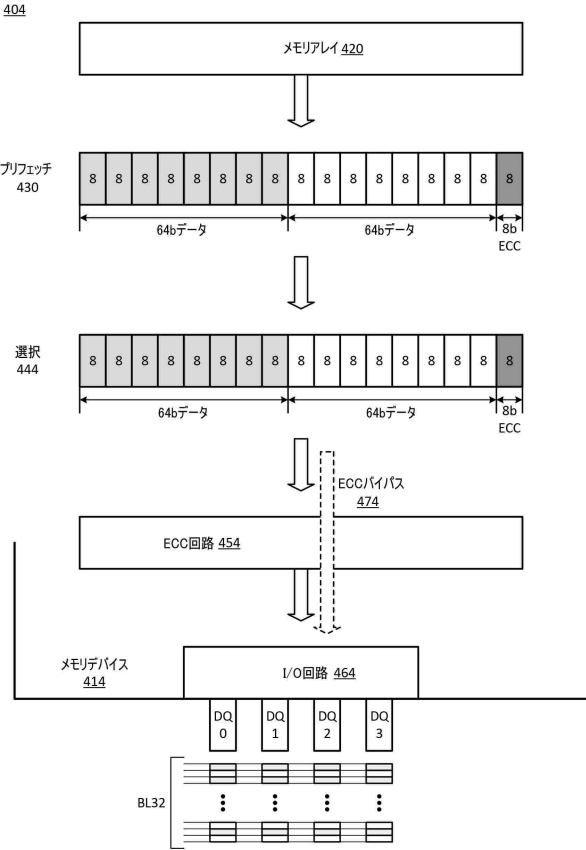


10

【図 4 A】



【図 4 B】



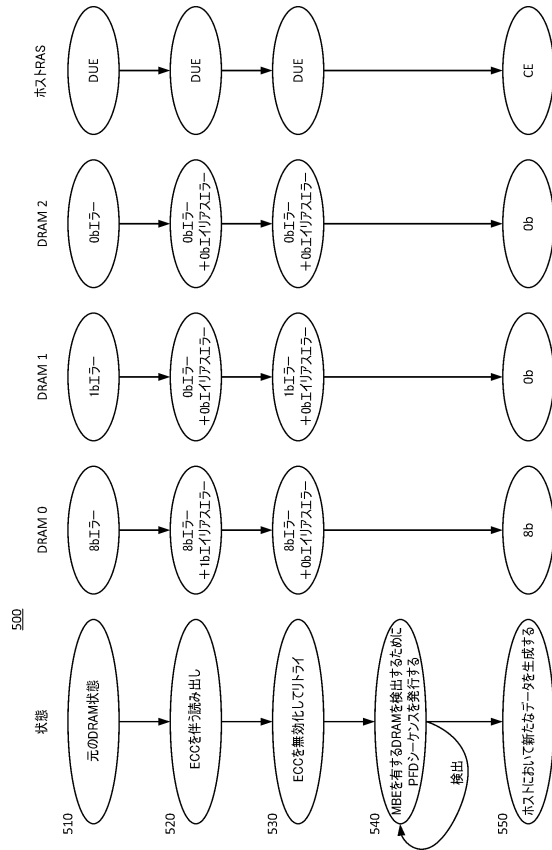
20

30

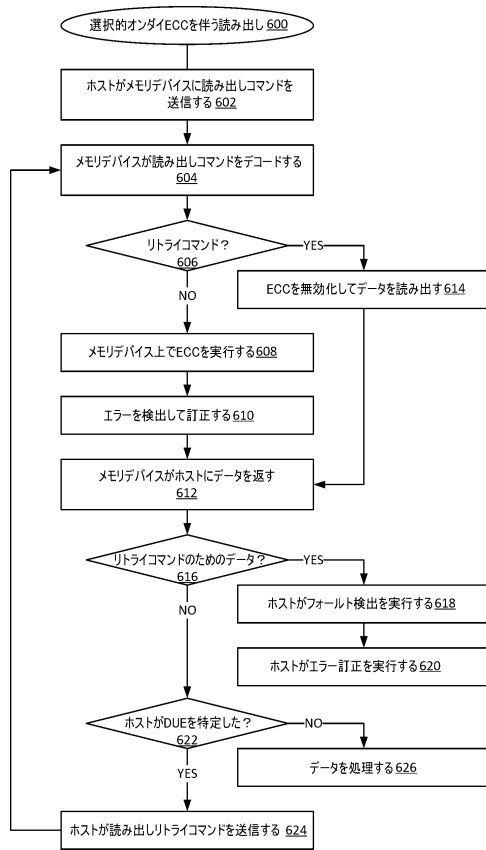
40

50

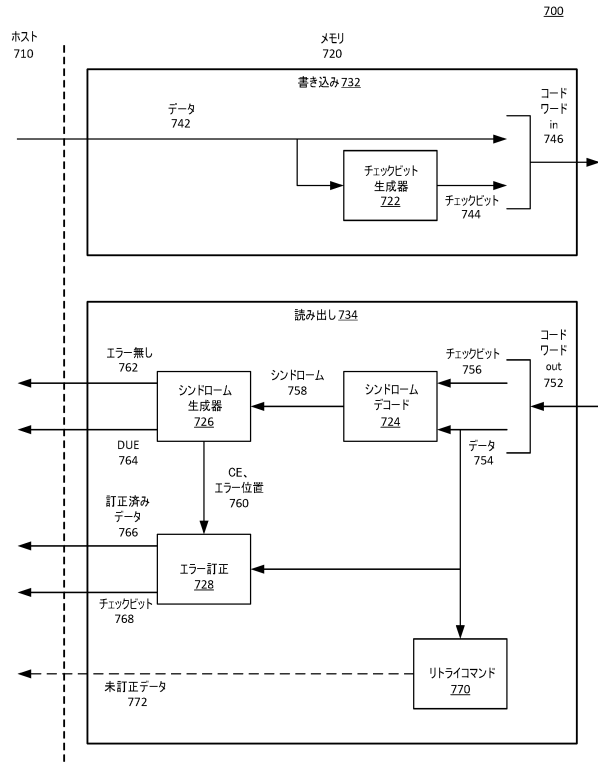
【図 5】



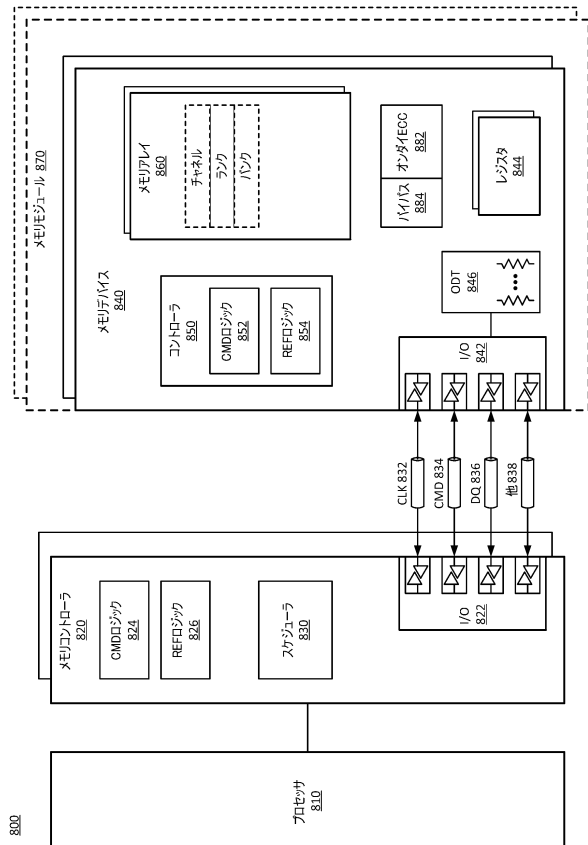
【図 6】



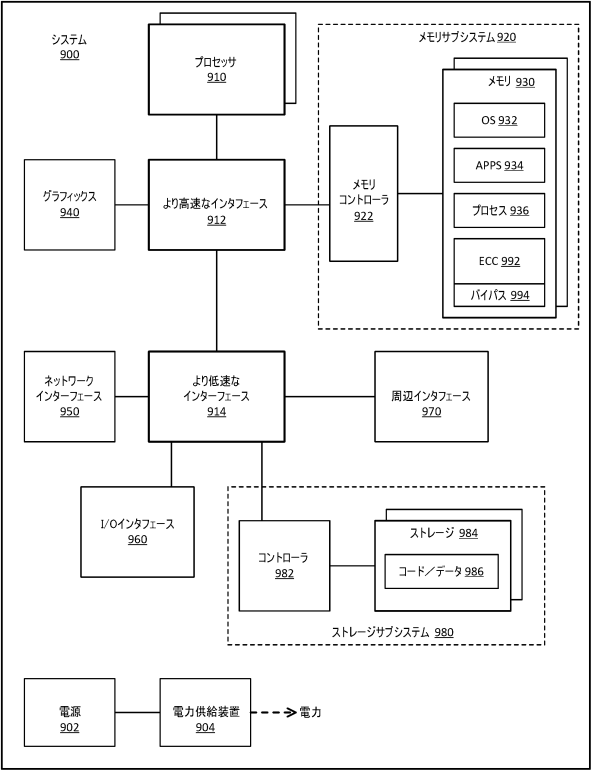
【図 7】



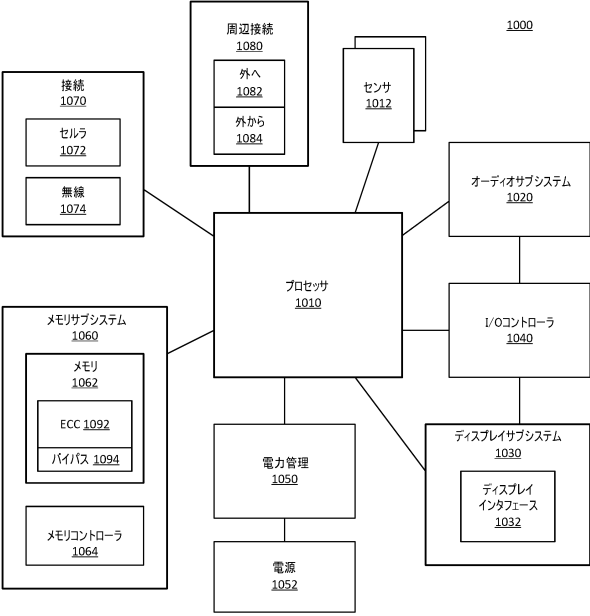
【図 8】



【図 9】



【図 10】



10

20

30

40

50

フロントページの続き

- レッジ ブレーバード・２２００ インテル・コーポレーション内
(72)発明者 ジョンウォン リー
アメリカ合衆国 ９５０５４ カリフォルニア州・サンタクララ・ミッション カレッジ ブレーバ
ード・２２００ インテル・コーポレーション内
- 審査官 田名網 忠雄
- (56)参考文献 特開２００６－２８５９７６（ＪＰ，Ａ）
米国特許出願公開第２０１９／０３４７１５８（ＵＳ，Ａ１）
特開平１１－２４９８２１（ＪＰ，Ａ）
特開２００６－１９１１８８（ＪＰ，Ａ）
特開２００６－２２７９５３（ＪＰ，Ａ）
米国特許出願公開第２０１７／００６０６８０（ＵＳ，Ａ１）
特開平０１－２０９５５２（ＪＰ，Ａ）
特表２０１７－５２９５９９（ＪＰ，Ａ）
特開２０１３－０６５３４７（ＪＰ，Ａ）
- (58)調査した分野 (Int.Cl., DB名)
G 0 6 F 1 3 / 1 6
G 0 6 F 1 2 / 0 0
G 0 6 F 1 1 / 1 0
G 1 1 C 2 9 / 4 2