

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4815491号  
(P4815491)

(45) 発行日 平成23年11月16日(2011.11.16)

(24) 登録日 平成23年9月2日(2011.9.2)

(51) Int.Cl. F 1  
**G 0 6 F 1 3 / 2 8 ( 2 0 0 6 . 0 1 )** G 0 6 F 1 3 / 2 8 3 1 0 A

請求項の数 14 (全 35 頁)

(21) 出願番号	特願2008-533662 (P2008-533662)	(73) 特許権者	503260918 アップル インコーポレイテッド アメリカ合衆国 95014 カリフォル ニア州 クパチーノ インフィニット ル ープ 1
(86) (22) 出願日	平成18年9月29日 (2006. 9. 29)	(74) 代理人	100064621 弁理士 山川 政樹
(65) 公表番号	特表2009-510630 (P2009-510630A)	(74) 代理人	100098394 弁理士 山川 茂樹
(43) 公表日	平成21年3月12日 (2009. 3. 12)	(72) 発明者	ゴー, ドミニク アメリカ合衆国・95132・カリフォル ニア州・サンノゼ・アラベル ウェイ・1 233
(86) 国際出願番号	PCT/US2006/038081		
(87) 国際公開番号	W02007/041301		
(87) 国際公開日	平成19年4月12日 (2007. 4. 12)		
審査請求日	平成21年8月17日 (2009. 8. 17)		
(31) 優先権主張番号	11/238, 790		
(32) 優先日	平成17年9月29日 (2005. 9. 29)		
(33) 優先権主張国	米国 (US)		

最終頁に続く

(54) 【発明の名称】 統合DMA

(57) 【特許請求の範囲】

【請求項1】

ホスト内のアドレス空間から第1のダイレクト・メモリ・アクセス(DMA)データを読み出し、DMA転送が割り当てられるチャンネルに応じて前記第1のDMAデータをDMAコントローラ内のオフロード・エンジン又はインターフェース回路のいずれかに供給するように構成されたダイレクト・メモリ・アクセス(DMA)コントローラであって、

前記第1のDMAデータは前記ホスト内のメモリ・コントローラと結合されたメモリにおける第1の複数の記憶領域から読み出され、前記第1の複数の記憶領域は前記ホスト内のアドレス空間にマッピングされており、

前記オフロード・エンジンは、前記第1のDMAデータの受け取りにตอบสนองして、前記第1のDMAデータに対する少なくとも第1のオペレーションを実行し、結果を生成するように構成され、かつ前記第1のDMAデータを前記オフロード・エンジンに供給するDMA転送中に前記第1のオペレーションの実行を少なくとも開始するように構成され、前記DMAコントローラは、前記DMA転送を記述するDMA記述子データ構造を用いて前記ホスト内の前記アドレス空間に前記結果を書き込むように構成され、前記結果は、前記ホスト内のメモリ・コントローラと結合されたメモリにおける第2の1つ又は複数の記憶領域に書き込まれ、前記第2の1つ又は複数の記憶領域は前記ホスト内のアドレス空間にマッピングされており、前記DMA記述子データ構造はメモリに格納され、前記DMAコントローラによってメモリから読み出され、

前記オフロード・エンジンは、前記インターフェース回路から受信されたデータを前記

10

20

アドレス空間に書き込むように構成されていることを特徴とするDMAコントローラ。

【請求項2】

前記第1のDMAデータを読み出すように構成された送信制御回路と、

前記第1のDMAデータを前記送信制御回路から受信するために結合された前記オフロード・エンジンと、

前記結果を受信するために前記オフロード・エンジンに結合された受信制御回路と、  
から構成され、

前記受信制御回路は、前記DMA転送を記述するDMA記述子データ構造を用いて、前記  
10 記ホスト内の前記アドレス空間に前記結果を書き込むように構成され、前記インターフェ  
ース回路から受信されたデータを前記アドレス空間に書き込むようにさらに構成されるこ  
とを特徴とする請求項1記載のDMAコントローラ。

【請求項3】

前記第1のオペレーションは前記第1のDMAデータを前記第2のDMAデータに変換  
し、前記オフロード・エンジンは前記受信制御回路に結合され、前記第2のDMAデー  
タを前記受信制御回路に供給するように構成されていることを特徴とする請求項1または2  
に記載のDMAコントローラ。

【請求項4】

前記第1のオペレーションは暗号化を含むことを特徴とする請求項1乃至3のいずれか  
に記載のDMAコントローラ。

【請求項5】

前記DMAコントローラは、前記第1のDMAデータにハッシュ関数を適用するように  
さらに構成され、前記結果は前記ハッシュ関数の出力を有することを特徴とする請求項1  
乃至4のいずれかに記載の装置。

【請求項6】

前記結果は、前記第1のDMAデータに対応するDMA記述子データ構造内に格納され  
ることを特徴とする請求項1乃至5のいずれかに記載の装置。

【請求項7】

前記第1のオペレーションは、ハッシュ関数オペレーション、巡回冗長検査の生成、ま  
たはチェック、チェックサム生成のうちの1つであることを特徴とする請求項6に記載  
の装置。

【請求項8】

排他的OR(XOR)回路をさらに備え、前記送信制御回路は複数のチャネルからデー  
タを読み出し、前記データを前記XOR回路に供給するように構成され、前記XOR回路  
は前記データを排他的論理和して前記結果を生成するように構成されていることを特徴と  
する請求項1乃至7のいずれかに記載のDMAコントローラ。

【請求項9】

ダイレクト・メモリ・アクセス(DMA)コントローラによって実行される方法であっ  
て、当該方法は、

ホスト内のメモリ・コントローラに結合されたメモリにおける第1の複数の記憶  
領域から第1のDMAデータをDMAコントローラ内へ読み出すステップと、

DMA転送が割り当てられるチャネルに応じて前記第1のDMAデータをDMAコント  
ローラ内のオフロード・エンジンまたはインターフェース回路のいずれかに供給するステ  
ップと、

前記オフロード・エンジンによる前記第1のDMAデータの受け取りにตอบสนองして、前記  
第1のDMAデータに少なくとも第1のオペレーションを実行し、前記オフロード・エン  
ジンに結果を生成するステップと、

前記DMA転送を記述するDMA記述子データ構造を用いて、前記ホスト内のメモリ・  
コントローラと結合されたメモリにおける第2の1つ又は複数の記憶領域に前記DMAか  
ら受信した前記結果を書き込むステップと、

前記インターフェース回路から受信したデータを前記アドレス空間に書き込むステップ

10

20

30

40

50

と、  
から成り、

前記第 1 の複数の記憶領域は、前記ホスト内のアドレス空間にマッピングされており、  
前記第 2 の 1 つ又は複数の記憶領域は、前記ホスト内のアドレス空間にマッピングされ  
ており、

前記 D M A 記述子データ構造はメモリに格納され、前記 D M A コントローラによってメ  
モリから読み出され、

前記オフロードエンジンは、前記第 1 の D M A データを前記オフロード・エンジンに供給する D M A 転送中に、前記第 1 のオペレーションの実行を少なくとも開始するように構成されていることを特徴とする方法。

10

【請求項 1 0】

前記第 1 のオペレーションは、前記第 1 の D M A データを第 2 の D M A データに変換し、前記方法は前記アドレス空間に前記第 2 の D M A データを書き込むステップをさらに有することを特徴とする請求項 9 に記載の方法。

【請求項 1 1】

前記第 1 のオペレーションは暗号化を含むことを特徴とする請求項 9 又は 1 0 に記載の方法。

【請求項 1 2】

前記オフロードエンジンに前記第 1 の D M A データにハッシュ関数を適用するステップをさらに有し、前記結果は前記ハッシュ関数の出力を含むことを特徴とする請求項 9 乃至 1 1 のいずれかに記載の装置。

20

【請求項 1 3】

前記結果は、前記第 1 の D M A データに対応する D M A 記述子データ構造内に格納され、前記第 1 のオペレーションはハッシュ関数、巡回冗長検査の生成、或いはチェックサム

の生成のうちの 1 つであることを特徴とする請求項 9 乃至 1 2 のいずれかに記載の装置。

【請求項 1 4】

複数のチャネルからデータを読み出し、前記データを排他的論理和して前記結果を生成するステップをさらに有することを特徴とする請求項 9 乃至 1 3 のいずれかに記載の方法。

【発明の詳細な説明】

30

【技術分野】

【0 0 0 1】

本発明は、集積回路の分野に関し、より具体的には、1 つ又は複数の集積回路を備えるシステム内のダイレクト・メモリ・アクセス ( D M A ) に関する。

【背景技術】

【0 0 0 2】

1 つ又は複数のプロセッサ、メモリ、入力 / 出力 ( I / O ) デバイス、又はインターフェースを含む一般的なシステムでは、I / O とメモリとの間でデータを転送するためダイレクト・メモリ・アクセス ( D M A ) 転送が利用されることが多い。あるシステムでは、D M A を使用する各 I / O デバイス又はインターフェース内に個々の D M A 回路が含まれている。別のシステムでは、1 つ又は複数の I / O デバイスが D M A 回路を共有している。

40

【0 0 0 3】

あるシステムはさらに、1 つの記憶領域から別の記憶領域へとデータをコピーするために使用可能な「データ・ムーバ」をも含んでいる。データ・ムーバは、プロセッサをオフロードできるがそうでない場合は、プロセッサはデータ移動 ( 例えば、一般的には一度に 3 2 ビット又は 6 4 ビットであるプロセッサが使用するビット幅でのデータの読み出しと書き込み ) を行う命令を実行しなければならない。データ・ムーバ用のプログラミング・モデルは一般的には、I / O デバイスとメモリとの間の通信に対応する D M A プログラミング・モデルとは異なっている。

50

## 【発明の開示】

## 【課題を解決するための手段】

## 【0004】

一実施態様において、装置は第1のインターフェース回路と、第1のインターフェース回路に結合されたダイレクト・メモリ・アクセス(DMA)コントローラと、DMAコントローラに結合されたホストとを備えている。第1のインターフェース回路はプロトコルに従ってインターフェースで通信するように構成されている。ホストは、ホストのメモリ・システム内の複数の記憶領域に少なくとも一部がマッピングされる少なくとも1つのアドレス空間を備えている。DMAコントローラは、第1のインターフェース回路とアドレス空間との間でDMA転送を行うように構成され、DMAコントローラはさらに、複数の記憶領域のうちの第1の複数の記憶領域と、複数の記憶領域のうちの第2の複数の記憶領域との間でDMA転送を行うように構成されている。方法も企図される。

10

## 【発明を実施するための最良の形態】

## 【0005】

以下の詳細な説明は、簡単に説明される添付図面を参照する。

## 【0006】

本発明には様々な修正及び代替形態が可能であるが、その特定の実施形態を図面で例示的に示し、本明細書で詳細に説明する。しかし、図面及び、その詳細な説明は本発明を開示された特定の形態に限定することを意図するものではなく、逆に本発明は添付の請求項に定義された本発明の趣旨と範囲に含まれる全ての修正形態、同等物及び代替形態を包含するものであると理解されたい。

20

## 【0007】

ここで図1を参照すると、システム10の一実施形態のブロック図が示されている。図示した実施形態において、システム10はホスト12と、DMAコントローラ14と、インターフェース回路16と、物理的インターフェース層(PHY)36とを含んでいる。DMAコントローラ14はホスト12とインターフェース回路16とに結合される。インターフェース回路16はさらに物理的インターフェース層36に結合されている。図示した実施形態において、ホスト12は、プロセッサ18A、18Bなどの1つ又は複数のプロセッサと、メモリ・コントローラ20A、20Bなどの1つ又は複数のメモリ・コントローラと、I/Oブリッジ(IOB)22と、I/Oメモリ(IOM)24と、I/Oキャッシュ(IOC)26と、レベル2(L2)キャッシュ28と、インターコネクタ30とを含んでいる。プロセッサ18A、18B、メモリ・コントローラ20A、20B、IOB22、L2キャッシュ28はインターコネクタ30に結合されている。IOB22はさらにIOC26とIOM24とに結合されている。DMAコントローラ14もIOB22とIOM24とに結合されている。図示した実施形態において、インターフェース回路16は周辺インターフェース・コントローラ32と、MAC34A、34Bなどの1つ又は複数のメディア・アクセス制御回路(MAC)とを含んでいる。MAC34A、34BはDMAコントローラ14と物理的インターフェース層36とに結合されている。周辺インターフェース・コントローラ32もI/Oブリッジ22とI/Oメモリ34とに結合され、(かつ、ひいては間接的にDMAコントローラ14に結合され)、かつ周辺インターフェース層36に結合されている。周辺インターフェース・コントローラ32及びMAC34A~34Cは各々、構成レジスタ38A~38Cを含んでいる。ある実施形態において、システム10のコンポーネントをチップ上のシステムとして単一の集積回路上に集積している。別の実施形態において、システム10を2つ以上の集積回路として実装している。

30

40

## 【0008】

ホスト12は1つ又は複数のアドレス空間を備えている。ホスト12内のアドレス空間の少なくとも一部はホスト12内の記憶領域にマッピングされている。すなわち、ホスト12はホストのアドレス空間内のアドレスにマッピングされたメモリ・システムを備えている。例えば、メモリ・コントローラ20A、20Bは各々、アドレス空間内にマッピン

50

グされた記憶領域を備えるメモリ（図示せず）に結合されている。ある場合は、アドレス空間全体が記憶領域にマッピングされている。別の場合は、アドレス空間の一部がメモリ・マップドI/Oでよい（例えば、周辺インターフェース・コントローラ32によって制御される周辺インターフェースが幾つかのメモリ・マップドI/Oを含んでいてもよい）。

**【0009】**

DMAコントローラ14は、インターフェース回路16とホストのアドレス空間との間でDMA転送を行うように構成されている。特に、DMA転送は、アドレス空間がマッピングされた記憶領域とインターフェース回路16との間で行われる。加えて、ある実施形態において、DMAコントローラ14を、アドレス空間内の記憶領域のセット間でDMA転送を行うように構成している。すなわち、このようなDMA転送の転送元と転送先が記憶領域である。データ・ムーバの機能はこのようにDMAコントローラ14に組み込まれており、ある実施形態において、別のデータ・ムーバは必要がないこともある。メモリ間のDMA転送のためのプログラミング・モデルは、他のDMA転送（例えば以下に、より詳細に記載するDMA記述子）用のプログラミング・モデルと同様のものでもよい。メモリ間のDMA転送をコピーDMA転送と呼ぶこともある。

10

**【0010】**

ある実施形態において、DMAコントローラ14を、DMAデータが転送される際にDMAデータに1つ又は複数のオペレーション（又は「関数」）を実行するように構成している。オペレーションはアドレス空間とインターフェース回路との間の転送に対して実行してもよく、ある実施形態において、コピーDMA転送に対して実行する。ある実施形態において、DMAコントローラ14が実行するオペレーションをプロセッサが実行する必要がないので、DMAコントローラ14によって実行されるオペレーションがプロセッサ18A、18Bにかかる処理負荷を軽減することがある。一実施形態において、DMAコントローラ14が実行するオペレーションの一部はパケット・データに対してなされるオペレーションである（例えば、暗号化/復号化、巡回冗長検査（CRC）の生成又はチェック、チェックサム生成又はチェックなど）。オペレーションは、例えば安価ディスク冗長アレイ（RAID）処理に使用してもよい排他的OR（XOR）オペレーションをも含んでいる。

20

**【0011】**

一般に、DMA転送は、少なくとも1つの転送先がホストのアドレス空間内の記憶領域又はその他のアドレス（1つ又は複数）である、転送元から転送先へのデータ転送である。DMA転送は、転送されるデータがシステム内の1つ又は複数のプロセッサ（例えばプロセッサ18A、18B）を通過せずに達成される。DMAコントローラ14は転送元を読み出し、転送先を書き込むことによってDMA転送を達成できる。例えば、DMAコントローラ14が（図示した実施形態において、データを読み出すためにインターコネクタ30上でコヒーレント読み出しトランザクションを実行するIOB22への）メモリ読み出し要求を生成し、読み出されたデータをDMAデータとしてインターフェース回路16に送信することによって、メモリからインターフェース回路16へのDMA転送を達成する。一実施形態において、DMAコントローラ14は、周辺インターフェース・コントローラ32を経たDMA転送のために、データをIOM24へ読み込む、読み込み要求を生成し、周辺インターフェース・コントローラ32は、IOM24からデータを読み出し、データを送信する。DMAコントローラ14がインターフェース回路16からデータを受け取り、DMAデータをメモリに転送する（図示した実施形態において、IOB22への）メモリ書き込み要求を生成することによって、インターフェース回路16からメモリへのDMA転送を達成する。一実施形態において、周辺インターフェース・コントローラ32がデータをIOM24に書き込み、DMAコントローラ14がデータをメモリに書き込ませる。したがって、DMAコントローラ14が周辺インターフェース・コントローラ32のためのDMAアシストを提供する。転送元の記憶領域へのメモリ読み出し要求、及び（メモリ読み出し要求によるDMAデータを含む）転送先の記憶領域へのメモリ書き込み

30

40

50

要求を生成することによって、コピーDMA転送を達成することができる。

【0012】

ホスト12は一般に、ホスト12のアドレス空間へとマッピングされたメモリとインターフェースするように構成された1つ又は複数のプロセッサとメモリ・コントローラを含んでいる。ホスト12は、ホスト12内でのプロセッサの性能を増強するためにL2キャッシュ28などの他の回路を場合により含んでいる。さらに、ホスト12は様々なI/O回路やDMAコントローラ14にインターフェースするための回路を含んでいる。ホスト12の1つの実装が図1に示されているが、他の実施形態は任意の構造を含み、かつDMAコントローラ14とインターフェース回路16へのインターフェースを含んでいる。

【0013】

プロセッサ18A、18Bは、プロセッサ18A、18Bによって実装される命令セット・アーキテクチャ内で定義される命令を実行するための回路を備えている。様々な実施形態で任意の命令セット・アーキテクチャを実装している。例えば、PowerPC(商標)命令セット・アーキテクチャを実装している。他の例示的な命令セット・アーキテクチャはARM(商標)命令セット、MIPS(商標)命令セット、SPARC(商標)命令セット、(IA-32とも呼ばれる) x86命令セット、IA-64命令セットなどを含んでいる。

【0014】

メモリ・コントローラ20A、20Bはメモリとインターフェースするように構成された回路を備えている。例えば、メモリ・コントローラ20A、20Bを、同期DRAM(SDRAM)、ダブル・データ・レート(DDR)SDRAM、DDR2SDRAM、ラムバスDRAM(RDRAM)などのようなダイナミック・ランダム・アクセス・メモリ(DRAM)とインターフェースするように構成されている。メモリ・コントローラ20A、20Bは、これらがインターコネクタ30から結合されるメモリ用の読み出しと書き込みトランザクションを受け取り、メモリに読み出し/書き込みオペレーションを実行する。読み出し、及び書き込みトランザクションは、DMAコントローラ14及び/又は周辺インターフェース・コントローラ32に代わってIOB22によって開始される読み出し及び書き込みトランザクションを含んでいる。加えて、読み出し及び書き込みトランザクションはプロセッサ18A、18B及び/又はL2キャッシュ28によって生成されるトランザクションを含んでいる。

【0015】

L2キャッシュ28は、プロセッサ18A、18Bによる低潜在アクセス及び/又はインターコネクタ30上の他のエージェントのために、メモリ・コントローラ20A、20Bが結合されているメモリ内の様々な記憶領域に対応するデータのコピーをキャッシュするように構成されたキャッシュ・メモリを備えている。L2キャッシュ28は任意の容量と構成(例えばダイレクト・マップ方式、群連想方式など)を備えている。

【0016】

IOB22は、DMAコントローラ14と周辺インターフェース・コントローラ32の代わりにインターコネクタ30上のトランザクションを通信するように構成された回路を備えている。インターコネクタ30がキャッシュ・コヒーレンスをサポートしてもよく、IOB22がコヒーレンスに関与し、IOB22によって開始されるトランザクションのコヒーレンスを保証している。図示した実施形態において、IOB22は、IOB22によって開始された最近のトランザクションをキャッシュするためのIOC26を使用する。IOC26は様々な実施形態で任意の容量及び構成を有していてもよく、コヒーレントである。例えば、DMAコントローラ14と周辺インターフェース・コントローラ32によって生成される読み出し/書き込みに起因して部分的にしか更新されないデータ・ブロックをキャッシュするためにIOC26を使用することができる。場合によっては、IOC26を使用してインターコネクタ30上の読み出し・修正・書き込みシーケンスを回避する。加えて、IOC26が読み出し/書き込みを完了するためにキャッシュ・ブロックの十分なオーナーシップを有している場合は、DMAコントローラ14又は周辺インタ

10

20

30

40

50

ーフェース・コントローラ 3 2 によって生成される読み出し / 書き込み用の I O C 2 6 内のキャッシュ・ヒットのためのインターコネクト 3 0 上でのトランザクションを回避する。他の実施形態は I O C 2 6 を含んでいない。

**【 0 0 1 7 】**

I O M 2 4 を、周辺インターフェース・コントローラ 3 2 又は D M A コントローラ 1 4 と I O B 2 2 との間で転送されるデータ用のステージング・バッファとして使用している。したがって、I O B 2 2 と D M A コントローラ 1 4 / 周辺インターフェース・コントローラ 3 2 との間のデータ・パスが I O M 2 4 を通っている。(読み出し / 書き込み要求、要求と関連するホストのアドレス空間内のアドレスを含む) 制御パスが直接、I O B 2 2 と D M A コントローラ 1 4 / 周辺インターフェース・コントローラ 3 2 の間にある。他の実施形態において、I O M 2 4 が含まれない。

10

**【 0 0 1 8 】**

インターコネクト 3 0 は、プロセッサ 1 8 A、1 8 B、メモリ・コントローラ 2 0 A、2 0 B、L 2 キャッシュ 2 8、I O B 2 2 の間で通信するための任意の通信媒体を含んでいる。例えば、インターコネクト 3 0 はコヒーレンシ・サポートを備えたバスである。あるいは、インターコネクト 3 0 は上記のエージェント間のポイント・ツー・ポイントインターコネクト、パケットベースのインターコネクト、又はその他のいずれかのインターコネクトでもよい。

**【 0 0 1 9 】**

インターフェース回路 1 6 は一般に、任意のインターフェース・プロトコルに従ってシステム 1 0 へのインターフェースで通信し、インターフェースで送信される通信を受信するためにシステム 1 0 内の他のコンポーネントと通信するか、又はインターフェースから受信された通信を供給するように構成された回路を備えている。インターフェース回路を、システム 1 0 内を転送元とする通信をインターフェース・プロトコルに変換し、システム 1 0 内で転送するためにインターフェースから受信された通信を変換するように構成している。例えば、インターフェース回路 1 6 は、周辺インターフェース・プロトコル(例えば周辺インターフェース・コントローラ 3 2)に従って通信するように構成された回路を備えている。別の例として、インターフェース回路 1 6 は、ネットワーク・インターフェース・プロトコル(例えば M A C 3 4 A、3 4 B)に従って通信するように構成された回路を備えている。

20

30

**【 0 0 2 0 】**

M A C 3 4 A、3 4 B は、ネットワーク・インターフェース用に定義されたメディア・アクセス・コントローラの機能を実装する回路を備えている。例えば、1 つ又は複数の M A C 3 4 A、3 4 B がギガビット・イーサネット標準を実装している。1 つ又は複数の M A C 3 4 A、3 4 B が 1 0 ギガビット・イーサネット・アタッチメント・ユニット・インターフェース(X A U I)標準を実装している。別の実施形態は、1 0 メガビット又は 1 0 0 メガビット標準などの他のイーサネット標準、又は他のいずれかのネットワーク標準を実装している。一実施形態において、6 つの M A C があり、そのうちの 4 つがギガビット・イーサネット M A C であり、2 つが X A U I M A C である。他の実施形態は、それよりも多い、又はそれよりも少ない M A C、及び任意に組み合わせたタイプの M A C を有している。

40

**【 0 0 2 1 】**

特に、イーサネット標準を実装する M A C 3 4 A、3 4 B は、受信されたパケットからインターフレーム・ギャップ(I F G)、プリアンプル、フレーム開始デリミタ(S F D)を取り除いてもよく、D M A が記憶するよう残りのパケット・データを D M A コントローラ 1 4 に提供している。M A C 3 4 A ~ 3 4 D は、D M A から受信されたパケット用の I F G、プリアンプル、S F D を送信 D M A 転送として挿入し、パケットを送信用の P H Y 3 6 に送信する。

**【 0 0 2 2 】**

周辺インターフェース・コントローラ 3 2 は、周辺インターフェースを制御するように

50

構成された回路を備えている。一実施形態において、周辺インターフェース・コントローラ 32 が周辺コンポーネント・インターコネク (PCI) エクスプレス・インターフェースを制御している。別の実施形態は、PCI エクスプレス・インターフェースに加えて、又はその代替として他の周辺インターフェース (例えば PCI、PCI-X、ユニバーサル・シリアル・バス (USB) など) を実装している。

#### 【0023】

PHY 36 は一般に、インターフェース回路 16 に制御されてシステム 10 への外部インターフェースで物理的に通信するように構成された回路を備えている。特定の一実施形態において、PHY 36 は PCI エクスプレス・レーンとして、又はイーサネット接続として使用するように構成されたシリアライザ/デシリアライザ (SERDES) 回路のセットを備えている。PHY 36 は SERDES、及び同期先入れ先出し型 (FIFO) バッファを経た送信用の 8b/10b エンコード/デコードを実行する回路、及びさらに PIC エクスプレス又はイーサネット通信リンクとして使用するために SERDES リンクを論理的に構成する回路を含んでいる。一実装形態では、PHY は、PCI エクスプレス・レーン又はイーサネット接続として構成可能な 24 の SERDES を備えている。所望の任意の数の SERDES を PCI エクスプレスとして構成してもよく、所望の任意の数の SERDES をイーサネット接続として構成している。

10

#### 【0024】

図示した実施形態において、周辺インターフェース・コントローラ 32 及び MAC 34 A、34 B 内の構成レジスタ 38 A ~ 38 C が示されている。各々の周辺インターフェース・コントローラ 32 及び MAC 34 A、34 B 内には 1 つ又は複数の構成レジスタがある。図 1 には示されていないが、システム 10 内には別の構成レジスタが存在している。構成レジスタを、周辺インターフェース・コントローラ 32 及び MAC 34 A、34 B の様々なプログラマブルに選択できるフィーチャを構成し、様々なフィーチャを有効化、無効化し、周辺インターフェース・コントローラ 32 及び MAC 34 A、34 B をオペレーション用に構成するなどを使用している。以下に記載の一実施形態において、構成レジスタを周辺インターフェース・コントローラ 32 及び MAC 34 A、34 B の実行時再構成のための制御記述子内で指定している。

20

#### 【0025】

様々な実施形態で、システム 10 が図 1 に示されたいずれかの要素 (例えばプロセッサ、メモリ・コントローラ、キャッシュ、I/Oブリッジ、DMAコントローラ、及び/又はインターフェース回路など) のうちの 1 つ又は任意の数の要素を含んでいてもよいことに留意されたい。

30

#### 【0026】

次に図 2 を参照すると、DMAコントローラ 14 の一実施形態のブロック図が示されている。図 2 の実施形態については、DMA 転送を行わせる記述子ソフトウェア・モデルを記載する。ある実施形態において、記述子モデルに加えて、又はその代替として、レジスタを基本としたソフトウェア・モデルがサポートされている。レジスタを基本としたモデルでは、各々の DMA 転送が DMA コントローラ 14 内にプログラムされ、DMA コントローラ 14 が DMA 転送を行う。転送の完了時には、DMA コントローラ 14 はプロセッサ 18 A、18 B の 1 つに割り込むか、DMA 転送の完了を判定するためにソフトウェアがポーリングできる (例えば DMA コントローラ 14 内部のレジスタ内の) ステータスを提供している。

40

#### 【0027】

記述子モデル内では、ソフトウェアがメモリ内の記述子データ構造を使用して多重 DMA 転送の実行を設定している。一般に、DMA 記述子は、DMA 転送を記述するデータ構造をメモリ内に備えている。例えば、DMA 記述子内の情報は、DMA 転送の転送元とターゲット、転送のサイズ、転送の様々な属性を特定する。場合によっては、DMA 転送の転送元とターゲットは暗黙的でもよい。メモリ内の記述子データ構造内に (例えば「記述子リング」内に) 複数の記述子が格納されてもよく、DMA コントローラ 14 にはデータ

50

構造内の第1の記述子のアドレスがプログラムされている。DMAコントローラ14は記述子を読み出し、表示されたDMA転送を行う。ソフトウェアとハードウェアとの間の記述子のオーナーシップを制御するため、多様な制御機構を使用している。例えば、記述子は、記述子内に記述されたDMA転送の実行準備ができることをDMAコントローラ14に示す有効ビット、すなわちイネーブル・ビットを含んでいる。所与のDMA転送の終了時にDMAコントローラ14がプロセッサ18A、18Bに割り込むべきことを指示するために、記述子内の割り込みビットを使用してもよく、又は記述子が最後のDMA転送を記述しており、DMAコントローラ14が休止すべきであることを指示するために、転送終了ビットを使用することもできる。あるいは、DMAコントローラ14が処理のために幾つの記述子内を利用可能であることを示すために、DMAコントローラ14が、ソフトウェアで増分できる記述子カウント・レジスタを実装している。DMAコントローラ14は記述子のプリフェッチが生成されたことを示すために、記述子カウント・レジスタを減分する。別の実施形態において、DMAコントローラ14は、記述子の消費(すなわち指定されたDMA転送の実行)を示すために記述子カウント・レジスタを減分する。さらに別の実施形態において、幾つの記述子が処理され、又はプリフェッチされたかを示すために、DMAコントローラ14は別個の記述子処理済みカウント・レジスタを使用する。

10

## 【0028】

DMAコントローラ14は送信(Tx)DMA転送と受信(Rx)DMA転送とを実行できる。Tx DMA転送は転送元としてホスト12内にアドレス空間(例えばメモリ・コントローラ20A、20Bに結合されたメモリ内の記憶領域)を有している。Rx DMA転送はターゲットとしてホスト12内にアドレス空間を有している。Tx DMA転送はターゲットとしてインターフェース回路16を有し、又は(例えばコピーDMA転送用の)ターゲットとしてホスト12のアドレス空間内に別のアドレスを有している。ホスト・アドレス空間のターゲットを有するTx DMA転送は、転送元アドレスから読み出されたDMAデータをターゲット・アドレスに書き込むためにRx DMAデータ・パスを使用する。ループバック回路40がTx DMAデータ・パスとRx DMAデータ・パスとの間のリンクを提供している。すなわち、「ループバック回路」は、送信DMAデータ・パスからTx DMAデータを受信するために、かつ受信DMAデータ・パス上のRx DMAデータを供給するために結合されたDMAコントローラにローカルな回路を備えている。ループバック回路40によって受信DMAデータ・パス上に供給されるデータは、(例えばコピーDMA機能用の)送信DMAデータ・パスから受信されたデータである。ある実施形態において、ループバック回路40によって供給されるデータは、受信されたデータからループバック回路40によって変換されたデータである。ある実施形態において、ループバック回路40によって供給されるデータは、ループバック回路40によって受信され、ループバック回路40によってデータ(例えば、チェックサム、CRCデータなど)上で計算された結果によって増大されたデータである。あるいは、ループバック回路40によって供給されるデータは、ループバック回路40によって受信されたデータでもよく(又はデータが供給されなくてもよく)、その結果がDMA転送用の記述子内に格納されている。変換されたデータ、又は計算され、データと共に含まれ、又はDMA記述子に書き込まれる結果を、本明細書では総称して「結果」と呼んでもよい。

20

30

40

## 【0029】

したがって、ある実施形態において、結果(例えば変換されたDMAデータ、又はデータから生成された結果)を生成するために、Tx DMAデータに1つ又は複数のオペレーション(又は「関数」)を実行するように、ループバック回路40を構成している。図2の実施形態において、ループバック回路40は、コピーFIFO42、オフロード・エンジン44、送信データ・パスに結合された排他的OR(XOR)回路46を含んでいる。コピーFIFO42は、Rx DMAデータ・パスでの送信用のTx DMAデータ・パスからの送信データを格納する。したがって、コピーFIFO42はコピーDMAオペレーションを実行する。オフロード・エンジン44を、DMAデータに様々なオペレーションを実行し、変換されたデータ、又はデータから分離された結果を生成するように構成

50

している。様々な実施形態で、オフロード・エンジン 44 を、所望の任意のオペレーションセットを実行するように構成している。一実施形態において、オフロード・エンジン 44 を、パケット処理を支援するオペレーションを実行するように構成している。例えば、パケットの暗号化及び/又は認証を行う様々なネットワーク・セキュリティ・プロトコルが開発されている。認証は一般的には、パケットの一部又は全部にわたるハッシュの生成を含んでいる。したがって、オフロード・エンジン 44 を、DMA 転送でパケット・データに暗号化/復号化、及び/又はハッシュ関数を実行するように構成する。加えて、オフロード・エンジン 44 を、チェックサム生成/チェック、及び/又は CRC の生成/チェックを実行するように構成する。チェックサム及び/又は CRC 保護は様々なパケット・プロトコルで使用されている。XOR 回路 46 はビット単位の XOR DMA データ (例えば複数の転送元からの DMA データ) でよい。XOR 回路 46 を、例えば安価ディスク冗長アレイ (RAID) の処理及び他のタイプの処理、又は XOR 関数を使用する処理をサポートするために使用することができる。

10

**【0030】**

ループバック回路 40 (及びより具体的には、ループバック・コンポーネント 42、44 及び 46) は、DMA データをループバック回路 40 に供給する DMA 転送中に DMA データ上で動作している。すなわち、ループバック回路 40 は、Tx DMA 転送が残りの DMA データを供給する間に、DMA データへのオペレーションの実行を少なくとも開始している。一般に、結果をメモリに、より一般的にはホストのアドレス空間に (例えば DMA データに添付された変換済み DMA データとして、又は Tx DMA 転送用の DMA 記述子内のフィールドなどの別個の結果記憶域に) 書き込んでもよい。

20

**【0031】**

ループバック回路 40 はさらに、オフロード・エンジン 44 と XOR 回路 46 (オフロード・エンジン 44 に結合されたオフロード FIFO 48、及び XOR 回路 46 に結合された XOR FIFO 50) を含んでいる。FIFO 48、50 は、DMA データが受信 DMA データ・パスで送信されるまで、オフロード・エンジン 44 と XOR 回路 46 のそれぞれからのデータを一時的に格納する。図示した実施形態において、FIFO 間を調停するためのアービタ 52 が備えられ、FIFO 42、48、50 に結合されている。アービタ 52 はループバック FIFO 54 にも結合され、これがターゲットに書き込まれるべきループバック回路 40 からのデータを一時的に格納する。

30

**【0032】**

図示した実施形態において、DMA コントローラ 14 は Tx DMA データ・パス上の Tx 制御回路 56 と、Rx DMA データ・パス上の Rx 制御回路 58 とを備えている。Tx 制御回路 56 は送信 DMA 転送のためのホスト 12 からのデータをプリフェッチする。特に、Tx 制御回路 56 は DMA 記述子をプリフェッチしてもよく、DMA データ用の転送元アドレスを決定するために DMA 記述子処理する。Tx 制御回路 56 は次いで DMA データをプリフェッチする。Tx 制御回路 56 のオペレーションを指すためにプリフェッチという用語を用いるが、プリフェッチは一般に、記述子や DMA データをホストのアドレス空間から読み出すために生成されるオペレーションである。

40

**【0033】**

Tx 制御回路 56 は DMA データをターゲットに送信する。この実施形態において、ターゲットはインターフェース回路 16 又はループバック回路 40 のいずれか 1 つ (より具体的には、図示した実施形態において、コピー FIFO 42、オフロード・エンジン 44、XOR 回路 46 の 1 つ) である。Tx 制御回路 56 は (例えばターゲット識別子を送信することにより) 送信済みデータのターゲットを特定する。あるいは、Tx 制御回路 56 とインターフェース回路 16 との間、及び Tx 制御回路 56 とループバック・コンポーネント 42、44、46 との間に物理的に分離されたパスを設けている。Tx 制御回路 56 は送信されるデータを一時的に格納するためバッファ 62 のセットを含んでいる。Tx 制御回路 56 はさらに、様々な制御情報にデータを供給する。制御情報は DMA 記述子からの情報を含んでいる。制御情報は、ループバック回路 40 用に、データをターゲット・ア

50

ドレス空間に格納するためのバッファ・ポインタ（１つ又は複数）を含んでいる。制御情報はさらに、DMA記述子内に含まれ、かつインターフェース回路１６又はループバック回路４０によって使用される他のいずれかの制御情報を含んでいる。それらの例は、DMA記述子の説明に関連して以下に、より詳細に記載する。

#### 【 0 0 3 4 】

R×制御回路５８はホスト１２のアドレス空間に書き込まれるDMAデータを受信し、かつデータをメモリに格納する書き込みを行う。一実施形態において、ソフトウェアは、受信されたDMAデータを格納するためメモリ内にバッファを割り当てる。R×制御回路５８は、バッファ・ポインタ（バッファを識別するホストのアドレス空間のアドレス）を備えている。R×制御回路５８は、データを格納するための書き込み用のアドレスを生成するためにバッファ・ポインタを使用する。R×制御回路５８用のバッファ・ポインタをプリフェッチするため、R×プリフェッチ・エンジン６０を備えている。R×プリフェッチ・エンジン６０は、バッファ・ポインタをR×制御回路５８に提供するために結合される。R×プリフェッチ・エンジン６０は、R×プリフェッチ・エンジン６０によって使用されるために、プリフェッチされたバッファ・ポインタを一時的に格納するためのバッファ６４のセットを含んでいる。同様に、R×制御回路５８は、メモリに書き込まれる受信済みのDMAデータを一時的に格納するためのバッファ６８のセットを含んでいる。

#### 【 0 0 3 5 】

一実施形態において、R×制御回路５８を、受信されたDMAデータ用の記述子を生成するように構成する。すなわち、受信されたDMAデータ用のDMA記述子をソフトウェアに作成させるのではなく、ソフトウェアがDMAデータを格納するためのバッファを割り当て、バッファ・ポインタを提供している。R×制御回路５８は、割り当てられたバッファに受信されたDMAデータを格納し、DMA転送用の記述子を作成している。R×制御回路５８によって作成された記述子は、受信されたDMAデータを格納する１つ又は複数のバッファへの１つ又は複数のバッファ・ポインタ、及びDMA転送を記述する別の情報を含んでいる。受信DMA記述子の例示的实施形態が図１２に示されており、以下に、より詳細に説明する。R×制御回路５８が、受信されたDMAデータ用の記述子を作成するので、記述子はソフトウェアによって作成された記述子よりも効果的である。例えば、ソフトウェアは最大の可能性があるDMA転送を受信することができるDMA記述子を作成する必要がある場合があり（又は、より大容量の転送用の複数の記述子が必要であり）、最大の可能性があるDMA転送を格納するための十分なバッファを割り当てる必要がある場合がある。一方、R×制御回路５８によって作成された記述子は、受信された実際の転送にとって十分大きい場合があり（かつ、受信されたデータを格納するのに十分なバッファを消費することがあるが）、必ずしもそれ以上大きい必要はない。

#### 【 0 0 3 6 】

図示した実施形態において、R×制御回路５８は、ループバックFIFO５４に結合され、インターフェース回路１６からのDMAデータをも受信するためのアービタ６６からDMAデータを受信する。アービタ６６は、データをR×制御回路５８に転送するため、ループバックFIFO５４と、インターフェース回路１６から受信されたDMAデータとを調停する。

#### 【 0 0 3 7 】

アービタ５２、６６は所望のいずれかの調停方式を実装している。例えば、優先度に基づく方式、ラウンドロビン方式、重み付きラウンドロビン方式、又はこれらの方式の組み合わせを用いる。ある実施形態において、調停方式がプログラマブルである。アービタ５２によって実装される調停方式（１つ又は複数）は、アービタ６６によって実装される方式（１つ又は複数）とは異なっている。

#### 【 0 0 3 8 】

T×制御回路５６、R×プリフェッチ・エンジン６０、R×制御回路５８は、図示した実施形態において、IOM/IOBインターフェース・ユニット７０に結合される。IOM/IOBインターフェース・ユニット５６は、T×制御回路５６、R×プリフェッチ・

エンジン 60、R×制御回路 58 の代わりに、IOB 22 及び IOM 24 と通信する。IOM / IOB インターフェース・ユニット 70 は、T×制御回路 56、R×プリフェッチ・エンジン 60、R×制御回路 58 からの読み出しと書き込みの要求を受け取り、これらの要求を満たすために IOB 22 及び IOM 24 と通信する。

#### 【0039】

特に、IOM / IOB インターフェース・ユニット 70 は、記述子、及び T×制御回路 56 からの DMA データの読み出し要求、及び R×プリフェッチ・エンジン 60 からのバッファ・ポインタを格納するメモリへの読み出し要求を受け取り、かつこの要求を IOB 22 に伝達する。IOB 22 は IOM 24 のどのエントリが要求されたデータ（例えば、ホストのアドレス空間又は IOC 26 からのデータの読み出し後、又は以前の要求からデータが IOB 24 内に既に存在することもある）を含むデータのキャッシュ・ラインを格納するか否かを示し、また IOM / IOB インターフェース・ユニット 70 は IOM 24 からデータを読み込み、そのデータを T×制御回路 56 又は R×プリフェッチ・エンジン 60 に供給する。IOM / IOB インターフェース・ユニット 70 は R×制御回路 58 からの書き込み要求も受け取ってもよく、IOM 24 に（IOB 22 によって書き込みデータ用に割り当てられたエントリで）書き込みデータを格納する。IOM 24 内にデータのキャッシュ・ラインが蓄積された後（又は、DMA 転送の完了のいずれか先の方）、IOM / IOB インターフェース・ユニット 70 はその旨を IOB 22 に通知し、（バッファ・ポインタから書き込まれるバッファへと導き出された）キャッシュ・ラインが書き込まれるアドレスを提供する。

#### 【0040】

一実施形態において、DMA コントローラ 14 は DMA 転送を送信し、DMA 転送を受信するための様々なチャネルをサポートしている。様々な実施形態で、任意の数のチャネルがサポートされている。例えば、一実装形態では、20 の送信 DMA チャネルを備え、64 の受信 DMA チャネルを備えている。各チャネルは転送元から転送先までの独立した論理データ・パスでよい。チャネルは、ソフトウェアにより所望通りに割り当てられる。

#### 【0041】

より具体的には、各送信チャネルをインターフェース回路 16 の 1 つ、又はループバック・コンポーネント回路 42、44、又は 46 の 1 つに割り当てる。全ての送信チャネルが使用される必要はない（すなわち、送信チャネルの一部は無効化されてもよい）。T×制御回路 56 は、チャネルごとに DMA 記述子と DMA データをプリフェッチする。すなわち、T×制御回路 56 は、処理に利用できる DMA 記述子を有する各チャネルごとに独立してプリフェッチを生成している。T×制御回路 56 は、生成されたプリフェッチから IOM / IOB インターフェース・ユニット 70 への読み出し要求を送信するためのプリフェッチを選択している。

#### 【0042】

インターフェース回路 16 の 1 つに各受信チャネルを割り当てる。必ずしも全ての受信チャネルが使用される必要はない（すなわち、受信チャネルの一部は無効化されてもよい）。R×制御回路 58 は、受信されたデータと共にチャネル番号を受信する。ループバック回路 40 は DMA 記述子から DMA 用のバッファ・ポインタを供給してもよく、R×制御回路 58 は、DMA データをホストのアドレス空間に書き込むためにバッファ・ポインタを使用する。割り当てられたチャネルでインターフェース回路 16 がプログラム可能であってもよく、チャネルを決定するためにパケット・フィルタリングを使用してもよい。インターフェース回路 16 はチャネル番号に DMA データを供給してもよく、R×制御回路 58 は、DMA データをホストのアドレス空間に書き込むためのチャネル用に R×プリフェッチ・エンジン 60 から提供されたバッファ・ポインタを使用してもよい。

#### 【0043】

DMA コントローラ 14 は、図 2 に示されているような様々な構成レジスタ 38D ~ 38H を含んでいる。構成レジスタ 38A ~ 38H は、DMA コントローラ 14 の様々なプログラム可能なフィーチャを有効化 / 無効化するために、かつ / 又は前述のように、プロ

10

20

30

40

50

グラム可能フィーチャを構成するためにプログラム可能である。例えば、T×制御回路56内の構成レジスタ38Dは、各チャンネルごとの記述子リングのアドレス、及び利用可能な記述子の数を示す記述子カウントを含んでいる。構成レジスタ38Dはさらに、インターフェース回路16への送信チャンネルの割り当て、及びコンポーネント・ループバック機能を含んでいる。チャンネルごとの様々な他の構成、及びチャンネルに関連しない構成が構成レジスタ38Dに格納されている。同様に、構成レジスタ38Eは、各インターフェース回路16、バッファ・リング・カウントなどのためのバッファ・ポインタ・リングのアドレス、及びチャンネルに関連しない様々な構成を格納する。構成レジスタ38Fは様々な受信DMA構成を格納する。構成レジスタ38Gは、ループバック回路40用の構成全体、及び所望の各コンポーネント回路用の構成を格納する。構成レジスタ38Gはアービタ52用の構成（例えば、調停方式の選択、選択された調停方式の構成のプログラミング）をも格納する。構成レジスタ38Hはアービタ66用の調停の構成を格納する。

#### 【0044】

T×制御回路56は記述子とDMAデータを取得するためにプリフェッチを実装するが、他の実施形態はプリフェッチを実装しない場合もあることに留意されたい。したがって、一般に、送信DMA転送（及びループバック回路40へのDMA転送）を行うように構成されたT×エンジン56、又はT×制御回路56がある。

#### 【0045】

この説明はDMA転送のためのバッファとバッファ・ポインタに言及したものであることに留意されたい。（62、64、68などのハードウェア記憶バッファではなく）バッファ・ポインタによって指されるバッファは連続記憶領域を備えている。（送信のために、又はDMAデータを受信するための領域として）DMAデータを格納するためにソフトウェアが記憶領域を割り当てる。バッファ・ポインタは、ホストのアドレス空間内の記憶領域のアドレスを備えている。例えば、バッファ・ポインタは記憶領域のベース、又は記憶領域の境界を指している。

#### 【0046】

次に図3を参照すると、オフロード・エンジン44の一実施形態のブロック図が示されている。図示した実施形態において、オフロード・エンジン44は入力バッファ80と、出力バッファ82と、セキュリティ回路のセット84A～84Dと、CRC生成器86と、チェックサム生成器88とを含んでいる。入力バッファ80はT×制御回路56と、セキュリティ回路84A～84Dと、CRC生成器86と、チェックサム生成器88とに結合される。出力バッファ82はセキュリティ回路84A～84Dと、CRC生成器86と、チェックサム生成器88とに結合される。出力バッファ82はオフロードFIFO48にも結合される。一実施形態のセキュリティ回路84Aが図3に、より詳細に示されており、セキュリティ回路84B～84Dは同様のものでよい。セキュリティ回路84Aはハッシュ回路90と暗号回路92とを含んでいる。ハッシュ回路90及び暗号回路92は双方とも入力バッファ80と出力バッファ82とに結合される。加えて、ハッシュ回路90の出力は、入力として暗号回路92に結合されており、暗号回路92の出力は、入力として「バタフライ」構造でハッシュ回路90に結合される。

#### 【0047】

セキュリティ回路84A～84Dは、パケット処理のセキュリティ機能をオフロードするために様々なオペレーションを行うように構成される。特に、セキュリティ回路84A～84Dは、暗号化/復号化（集合的に暗号化、又は暗号機能と呼ばれる）、及び様々なセキュア・パケット仕様（例えばセキュア・インターネット・プロトコル（IPSec）、又はセキュア・ソケット・レイヤー（SSL））に含まれるハッシュ関数を実行するように構成される。

#### 【0048】

一般的には、セキュア・パケット・プロトコルを使用する通信には、エンドポイントがそれが使用することができるプロトコルと通信する交渉セッション、エンドポイントがサポートするセキュリティ方式、暗号化とハッシュのタイプ、キーの交換又は認証などが含

10

20

30

40

50

まれる。その場合、合意に基づくプロトコル、暗号化などを使用したバルク転送段階がある。バルク転送中、(例えばインターフェース回路16の1つから受信DMAパスを経て)パケットがホスト12内に受信される。ソフトウェアは、キー、暗号化アルゴリズムなどを得るためにメモリ内のデータ構造を調査し、パケットを復号化、及び/又は認証するためにオフロード・エンジン44を介してDMA転送を準備する。同様に、ソフトウェアはセキュア送信のためのパケットを準備し、パケットを暗号化及び/又は認証するためにオフロード・エンジン44を介してDMA転送を使用する。

#### 【0049】

ハッシュ回路90は、パケットの認証に使用できる様々なハッシュ関数を実装している。一般的には、ハッシュはパケットの少なくとも一部にわたって計算され、ハッシュ結果がパケット内に含まれる。パケットが転送先で受信されると、パケット内のいずれかのフィールドが変更されたか否かを検知するために(ひいてはパケットがその転送元からの転送で修正されたか否かを検知するために)、ハッシュがチェックされる。一実施形態において、ハッシュ回路90内で以下のハッシュ関数がサポートされている:メッセージ・ダイジェスト5(MD-5)/セキュア・ハッシュ・アルゴリズム-1(SHA-1)、及びハッシュ・メッセージ認証コード(HMAC)。他の実施形態はSHA-2を実装している。他の実施形態は、上記の関数、及び他の関数のサブセット又はスーパーセットを含む他のいずれかのハッシュ関数のセットを実装している。

10

#### 【0050】

暗号回路92を、暗号化関数を実行するように構成する。セキュア・パケット仕様に応じて、場合によってはハッシュ・データを含むパケットの少なくとも一部に暗号化関数が適用される。暗号化関数のいずれものセットは、様々な実施形態でサポートされている。例えば、一実施形態において、以下の暗号化/復号化アルゴリズムが暗号回路92に実装される。すなわち、データ暗号化標準(DES)、トリプル・データ暗号化標準(3DES)、先進データ暗号化標準(AES)、Kasumi、主張されているロンのコード4(ARC4)、及び/又はロンのコード4(RC4)である。

20

#### 【0051】

ある場合には、認証及び暗号化の双方の関数が使用される場合は、送信用にパケットが準備されるときに最初に暗号化が行われ、次いで暗号化されたデータ(例えばIPSec)に認証ハッシュが実行される。別の場合は、認証ハッシュが最初に実行され、二番目に(ハッシュ・データを含む)パケットの暗号化が実行される(例えばSSL)。いずれの場合も、認証ハッシュと復号化は受信されたパケットでは逆の順序で行われる。

30

#### 【0052】

セキュリティ回路84A~84Dは、回路90と92との間のパタフライ接続を介して、単一のDMA転送でのどの順序でのデータの暗号化及びハッシュもサポートできる。すなわち、暗号化が最初に実行される場合は、暗号化(又は復号化される)データでハッシュ関数を計算するために、セキュリティ回路84Aに供給されるデータが暗号回路92に送られ、暗号回路92の出力がハッシュ回路90の入力に送られる。最初にハッシュが実行される場合は、セキュリティ回路84Aに供給されるデータがハッシュ回路90に送られ、ハッシュ回路90の出力が暗号回路92の入力に送られる。セキュリティ回路84A~84Dは、所定のDMA転送でハッシュ関数のみ又は暗号化関数のみの実行をもサポートする。セキュリティ回路84A~84Dに向けられるDMA転送用のDMA記述子からの制御情報が、セキュリティ回路84A~84Dを経たデータの伝送を制御する。

40

#### 【0053】

図示した実施形態は、4つのセキュリティ回路84A~84Dを示している。他の実施形態は、1つのセキュリティ回路を含む任意の数のセキュリティ回路を含んでもよい。一実施形態において、セキュリティ回路84A~84Dは、システム10で使用されるシステム・クロックの周波数の2倍の周波数でクロックされ、システムのクロック周期ごとに、(1つはシステム周期クロックの最初の半期で実行され、もう1つはシステムのクロック周期の第2の半期で実行される)2つのオペレーションを受け取る。したがって、セキ

50

セキュリティ機能を実行するために、ソフトウェアによって選択される8つのセキュリティ論理回路がある。

【0054】

CRC生成器86を、CRC生成を指定するDMA転送で供給されるデータ上にCRCデータを生成するように構成する。CRC生成を、受信されたパケットからのCRCデータをチェックするために使用する。例えば、CRC生成器86内で生成されたデータが、受信されたパケット内の対応するCRCデータと比較される。あるいは、受信されたパケット内のCRCデータを、CRC生成器86を経たDMA転送に含めてもよく、受信されたパケット内のエラーを検知するために、その結果を所定値と比較してチェックする。ある実施形態において、2つ以上のCRC生成器86がある。さらに、オフロード・エンジン44内に物理的に備えられるものよりも多くの論理的CRC生成器を備えるため、セキュリティ回路84A~84Dと同様に、CRC生成器86(1つ又は複数)がシステムのクロック周期数の2倍の周波数でクロックされる。特定の一実施形態において、等しい数の論理ユニット(8)をセキュリティ回路84A~84Dに備えるため、システムのクロック周波数の2倍の周波数でクロックされる4つのCRC生成器86がある。

10

【0055】

チェックサム生成器88を、チェックサム生成を指定するDMA転送で供給されるデータ上にチェックサムを生成するように構成する。チェックサム生成を、受信されたパケットからのチェックサム・データをチェックするために使用する。例えば、チェックサム生成器88内で生成されたデータが、受信されたパケット内の対応するチェックサムと比較される。あるいは、受信されたパケット内のチェックサム・データを、チェックサム生成器88を経たDMA転送に含めてもよく、受信されたパケット内のエラーを検知するために、その結果を所定値と比較してチェックする。ある実施形態において、2つ以上のチェックサム生成器88がある。

20

【0056】

入力バッファ80は、ターゲット回路84A~84D、86、又は88がデータに対してオペレーションを行うまで、Tx制御回路56によって供給されるデータを一時的に格納する。ターゲット回路84A~84D、86、88は、オフロードFIFO48に書き込まれる出力データを出力バッファ82に出力する。別の実施形態において、入力バッファ80、及び/又は出力バッファ82を含めなくてもよい。

30

【0057】

図4は、システム10の一実施形態で実装されてもよいDMAのモデルの一実施形態を示すブロック図である。前述のように、DMA転送は、転送元のアドレス空間又はインターフェース(ブロック100)から転送先のアドレス空間又はインターフェース(ブロック102)へと行われる。DMA転送は図4のブロック104で表わされる。場合によっては、DMA転送は、結果を生成するためにDMAデータに対する1つ又は複数のオペレーション、又は関数の実行(ブロック106)を含む。結果はDMAブロック104に戻され、ターゲットのアドレス空間に供給される。ある場合は、結果を、ターゲットのアドレス空間に書き込むことができるDMAデータに変換する。別の場合は、結果をDMAデータから分離し、DMAデータを増大して(例えばDMAデータの終端に添付)、又は別個の位置内に(例えば、DMA転送用のDMA記述子内に)格納する。

40

【0058】

図4は、ある場合は単一のDMA転送を示す。別の場合は、図4のモデルを完成させるために複数のDMA転送を使用する。例えば、関数106をループバック回路40(又は、図2の実施形態の場合はそのコンポーネント)によって実行する。したがって、図示した実施形態において、インターフェース回路16をターゲットとするDMA転送で関数を実行したい場合は、2つのDMA転送を行う。第1のDMA転送は、転送元のアドレス空間からターゲットのアドレス空間に対して行われ、実行される所望の関数を指定する。引き続き、転送元として第1のDMA転送のターゲットのアドレス空間、及びターゲットとして所望のインターフェース回路を使用して第2のDMA転送を行う。

50

## 【 0 0 5 9 】

例えば、M A C 3 4 A、3 4 Bを介して転送用のソフトウェアを使用してパケットを準備してもよく、パケットをホスト1 2内のメモリに格納する。暗号化及び/又はハッシュを使用したセキュア転送が望ましいことがあり、したがって、ソフトウェアがホスト1 2内の第1の記憶領域から第2の記憶領域への第1のDMA転送を構築してもよく、第1のDMA転送がオフロード・エンジン4 4内で暗号化及び/又はハッシュを指定している。ソフトウェアはさらに、第2の記憶領域からターゲットM A C 3 4 A、3 4 Bへの第2のDMA転送を準備する。同様に、パケットは、DMAとしてホスト1 2内のメモリ内に受信されて、パケットのハッシュ及び/又は復号化をチェックするため、ソフトウェアがオフロード・エンジン4 4を介したDMA転送を行う。

10

## 【 0 0 6 0 】

別の実施形態は、データがインターフェース回路1 6に送信される際に、関数の適用を許容している。例えば一実施形態において、TCPパケットのTCPヘッダ内に含まれるチェックサムを生成するため、IOB 2 2内で部分的チェックサムの生成がサポートされる。このようなパケット向けにDMA転送が指定されると、IOB 2 2がこのパケット・データをIOM 2 4に累積し、各キャッシュ・ラインごとに部分的チェックサムを生成する。IOB 2 2からの各キャッシュ・ライン及び、部分的チェックサムが計算されなかったデータに、部分的チェックサムを使用してDMAコントローラ1 4内で最終チェックサムが生成されてもよく、DMAコントローラ1 4が、計算されたチェックサムをTCPヘッダに挿入する。別の実施形態は、関数が実行され、データが送信される単一のDMA転送の一部として、ループバック回路4 0の出力が直接インターフェース回路1 6に送られることを許容している。さらに、別の実施形態は、受信DMA転送中に受信されたDMAデータ上で関数を実行するため、Rx DMAデータ・パスからのデータがループバック回路4 0に入力されることを許容している。

20

## 【 0 0 6 1 】

次に図5を参照すると、記述子のデータ構造とバッファ・ポインタのデータ構造を格納する記憶領域1 1 0のブロック図が示されている。図5の実施形態において、記述子のデータ構造は記述子リングのセット1 1 2 A ~ 1 1 2 Nを含んでいる。DMAコントローラ1 4によってサポートされる各DMAチャネル(例えば図5のチャネル0からチャネルN)ごとに1つの記述子がある。すなわち、DMAチャネルと記述子リングとの間には1対1の対応関係があり、所定のDMAチャネル用のDMA転送は、チャネルに割り当てられた記述子リング1 1 2 A ~ 1 1 2 N内に対応する記述子を有している。加えて、図5に示された実施形態において、バッファ・ポインタのデータ構造はバッファ・ポインタ・リングのセット1 1 4 A ~ 1 1 4 Mを含んでいる。インターフェース回路1 6ごとに(例えば図5では、インターフェース回路0からM、ただしM + 1はインターフェース回路1 6の数でよい)、バッファ・ポインタ・リングがある。すなわち、インターフェース回路と記述子リングとの間には1対1の対応関係があり、そのインターフェースで受信されるDMA用に使用されるバッファ・ポインタが、そのインターフェース回路に割り当てられたバッファ・ポインタ・リング1 1 4 Aから1 1 4 Mから取り出される。

30

## 【 0 0 6 2 】

各記述子リング1 1 2 A ~ 1 1 2 Nは対応するDMAチャネル用の記述子のセットを備えている。送信DMAチャネル用には、記述子はリング内の最初の記述子から最後の記述子までリング内に含まれる順序で処理され、次いで最後の記述子が処理された後、リング内の最初の記述子に折り返される。したがって、任意の時点で、リング内のどのリングも、次に処理されるべき「現在の記述子」であると見られる。前述のように、ソフトウェアが、多様な方式でのDMAチャネルでの処理に利用できる記述子の数を制御している。したがって、(対応する記述子リング内に)所定の送信DMAチャネルで利用できる記述子があれば、DMAコントローラ1 4は(他のDMAチャネルを有するリソースと調停して)指定のDMA転送を実行する。本実施形態の受信DMA転送の場合は、DMA転送がそのチャネルで受信されると、対応する記述子リング内の記述子が消費される。DMAコン

40

50

トローラ 14 は、受信された DMA データを格納するために使用されるバッファ・ポインタ (1 つ又は複数) で現在の記述子、及び転送ステータス情報などの DMA 転送に関連する他の情報を書き込んでよい。

【 0 0 6 3 】

別の実施形態は他のデータ構造 (例えば記述子のリンクされたリスト) を使用する。各記述子リング 1 1 2 A ~ 1 1 2 N のベース・アドレスが (例えば、チャンネルが送信チャンネルであるか受信チャンネルであるかに応じて、構成レジスタ 3 8 D 又は 3 8 F 内にプログラムされて) DMA コントローラ 14 に提供される。記述子リング 1 1 2 A ~ 1 1 2 N の他の属性 (例えば範囲) もプログラムされる。ある実施形態において、所定のリング内の記述子は固定サイズのものでよいので、所定の記述子はリングのベース・アドレスから固定的にオフセットされる。別の実施形態において、記述子は可変サイズのものでよい。さらに別の実施形態において、記述子は、構成レジスタ 3 8 D 又は 3 8 F 内のプログラム可能な属性に応じて、固定サイズのもので可変サイズのものでよい。属性はチャンネルごとにプログラム可能なものでもよく、又はチャンネル全体としてプログラムされる。

10

【 0 0 6 4 】

各バッファ・ポインタ・リング 1 1 4 A ~ 1 1 4 M は、対応するインターフェースからの R x DMA 転送からの DMA データを格納するために使用されるソフトウェアによって割り当てられたメモリ内のバッファを指すバッファ・ポインタを備えている。記述子リング 1 1 2 A ~ 1 1 2 N と同様に、ソフトウェアは、バッファ・ポインタ・リング 1 1 4 A ~ 1 1 4 M 内のバッファ・ポインタを任意の所望の方式で DMA コントローラ 14 に利用できるようにしている。各インターフェース用のバッファ・ポインタ・リングのベース・アドレスが、(例えば、図 2 の実施形態の R x プリフェッチ・エンジン 6 0 内の構成レジスタ 3 8 E 内で) DMA コントローラ 14 内にプログラムされてもよく、所定の任意の時点でバッファ・ポインタ・リング内のバッファ・ポインタの 1 つが、対応するインターフェース用に次に消費されるバッファ・ポインタである。

20

【 0 0 6 5 】

DMA チャンネルではなく、インターフェース回路に関連するバッファ・ポインタ・リング 1 1 4 A ~ 1 1 4 M を備えることによって、ある実施形態において、ソフトウェアはより多数の DMA チャンネルではなくより少数のインターフェース回路にバッファを割り当てられる。ある場合は、このメモリの割り当ての方が効率的である。そのトラヒックがどのチャンネルで受信されるかをソフトウェアが前もって知る必要なく、より多くのトラヒックを扱うインターフェース回路により多くのバッファを割り当てる。DMA データが所定のインターフェースから受信されると、そのインターフェースに割り当てられたバッファ内にデータが格納され、DMA データが受信されるチャンネル用の記述子にバッファ・ポインタが書き込まれる。どの受信 DMA チャンネルが DMA 転送に関連しているかに応じて、記述子は記述子リング 1 1 2 A ~ 1 1 2 N の 1 つにある。

30

【 0 0 6 6 】

バッファ・ポインタ・リング 1 1 4 A ~ 1 1 4 M は、バッファ・ポインタごとのサイズ・フィールド (図 5 の Sz) をも含んでいる。サイズ・フィールドが、対応するバッファ・ポインタによって指されるバッファのサイズを示している。したがって、例えば利用可能なメモリ量、所定のインターフェース上の予測される DMA 転送のサイズなどに基づいて、ソフトウェアが異なるサイズのバッファを割り当てる。

40

【 0 0 6 7 】

次に図 6 を参照すると、所定のインターフェース回路について R x プリフェッチ・エンジン 6 0 の一実施形態のオペレーションを示すフローチャートが示されている。R x プリフェッチ・エンジン 6 0 は、並行して、かつ独立して動作する各インターフェース回路ごとに図 6 に示されたオペレーションを実行する回路を含んでいる。容易に理解できるように、図 6 には特定の順序でブロックが示されているが、ブロックは図 6 に示されたオペレーションを実行する組み合わせの論理回路で並行して実行されてもよい。ある実施形態において、1 つ又は複数のブロック又はフローチャート全体が複数のクロック周期にわたっ

50

てパイプライン接続される。

【0068】

R×プリフェッチ・エンジン60は、(インターフェース回路に対応するバッファ・ポインタ・リング114A~114M内の)インターフェース回路用のバッファ・ポインタを利用可能であるか否か(決定ブロック120)、かつ、インターフェース回路にバッファ・ポインタが必要であるか否か(決定ブロック122)を判定する。少なくとも1つのバッファ・ポインタが利用可能であり、必要である場合は(決定ブロック120、122の「yes」区間)、R×プリフェッチ・エンジン60はホスト12内のメモリ内のバッファ・ポインタ・リング114A~114Mからバッファ・ポインタ(1つ又は複数)を読み出すプリフェッチ要求を生成する(ブロック124)。

10

【0069】

バッファ・ポインタは一般に、R×プリフェッチ・エンジン60によって未だプリフェッチされていない対応するバッファ・ポインタ・リング114A~114M内にバッファ・ポインタがあれば「利用可能」である。ソフトウェアによってバッファ・ポインタがバッファ・ポインタ・リング114A~114M内に挿入されてもよく、ソフトウェアは、前述の方式のいずれか(例えばバッファ・ポインタ・リングのエントリ内の有効ビットを使用し、前述のDMA記述子カウントと同様のバッファ・ポインタ・リングのカウントを増分して)バッファ・ポインタが利用可能であることを指示している。同様に、バッファ・ポインタは多様な方式で「必要である」と見られる。例えば、受信DMAチャンネルが有効化され、そのチャンネル用にプリフェッチされたバッファ・ポインタがない場合は、バッファ・ポインタは「必要である」可能性がある。ある実施形態において、プリフェッチされるべきいくつかのバッファ・ポインタを示し、又はプリフェッチされるべき最小数と最大数のバッファ・ポインタを示すようにR×プリフェッチ・エンジン60がプログラムされる。R×プリフェッチ・エンジン60は、プログラムされた数のバッファ・ポインタのプリフェッチを試みるために、バッファ・ポインタのためのプリフェッチ要求を生成する。

20

【0070】

前述のように、図6のオペレーションが、有効化された各々のインターフェース回路ごとに並行して行われる。2つ以上のプリフェッチ要求が同時に生成される場合は、R×プリフェッチ・エンジン60はプリフェッチ要求の中から選択するための回路も含む。例えば、一実施形態において、R×プリフェッチ・エンジン60はプリフェッチ要求の中から固定優先度方式を実装する。別の実施形態において、R×プリフェッチ・エンジン60は、最少数のバッファ・ポインタが現在選択され、準備ができるインターフェース回路に対応するプリフェッチ要求を選択する。別の実施例として、R×プリフェッチ・エンジン60は、現在プリフェッチされるバッファ・ポインタと、そのインターフェース回路用の所望のバッファ・ポインタ数との最大の差をどのインターフェース回路が有しているか否かに基づいて要求に重み付けする。さらにラウンドロビン又は優先度に基づく選択機構も利用してもよく、必要ならば、これらの機構はインターフェース回路間のプログラム可能な重みを含む。ポインタが各インターフェース回路ごとにプリフェッチされることを保証するため、インターフェースごとのタイムアウトなどの飢餓状態防止機構をも使用できる。

30

40

【0071】

プリフェッチされたバッファ・ポインタが読み出されるべきIOM24内で利用可能である場合は、IOM/IOBインターフェース回路70によってその旨がR×プリフェッチ・エンジン60に通知される。ある実施形態において、R×プリフェッチ・エンジン60はIOM24からポインタの一部又は全部を読み出し、プリフェッチされたバッファ・ポインタを必要に応じてR×制御回路58に供給する。

【0072】

次に図7を参照すると、R×制御回路58の一実施形態の、データの受信に回答したオペレーションを示したフローチャートが示されている。データはループバック回路40、又はインターフェース回路16のどちらから受信される。図7では理解し易いように、ブ

50

ロックが特定の順序で示されているが、ブロックは図7に示されたオペレーションを実行する組み合わせの論理回路で並行して実行される。ある実施形態において、1つ又は複数のブロック又はフローチャート全体が複数のクロック周期にわたってパイプライン接続される。

#### 【0073】

Rx制御回路58は、DMAデータを受信するためにバッファが既に使用中であるか否かを判定する(決定ブロック130)。バッファがDMAデータを格納するために既に選択されており、未だフル状態になっていない場合は、バッファは使用中である。Rx制御回路58を、ループバック回路40用の各Rxチャンネル及び1つ又は複数のバッファ・ポインタ用の能動バッファを保持するように構成する。あるいは、ループバック回路40はバッファ・ポインタに各データ転送を供給してもよく、Rx制御回路58がループバック回路40用にポインタを保持しない。バッファが使用中ではない場合は(決定ブロック130、「no」区間)、Rx制御回路48は、読み出されるバッファを識別するために次のバッファ・ポインタを選択する。ループバック回路40からのDMA転送用には、ループバック回路40によって次のバッファ・ポインタが提供される。インターフェース回路16からのDMA転送用には、Rxプリフェッチ・エンジン60から次のバッファ・ポインタが提供され、Rx制御回路58はRxプリフェッチ・エンジン60にポインタの消費を指示している。

#### 【0074】

いずれの場合も、Rx制御回路58は受信されたデータをバッファに書き込んでもよい(ブロック134)。すなわち、Rx制御回路58はIOM/IOBインターフェース回路70に対してDMAデータをメモリに書き込むための書き込み要求を生成している。Rx制御回路58は、少なくともインターフェース回路16から受信されたデータについて、バッファがフル状態である場合にその旨を判定するため、バッファに書き込まれたバイト数を監視している。ある実施形態において、ループバック回路40によってバッファのサイズが提供され、Rx制御回路58がループバック転送についてもバッファのフル状態を監視している。バッファがフル状態である場合、又は転送元(ループバック回路40、又はインターフェース回路16)によってデータ転送がDMA転送の終了であるものと示された場合は(決定ブロック136、「yes」区間)、Rx制御回路58はDMA転送用の記述子にバッファ・ポインタを書き込む(ブロック138)。ループバック転送の場合は、バッファ・ポインタが転送元のDMA記述子からの転送先バッファ・ポインタであるので、受信されたデータ用のDMA記述子はなくてもよく、このような転送にはブロック138は実行されない。データ転送がDMA転送の終了であることが示された場合は(決定ブロック140、「yes」区間)、Rx制御回路58は、転送処理の様々な終了を実行する(ブロック142)。例えば、インターフェース回路16からのDMA転送の場合、Rx制御回路58はDMA転送用の様々なステータスを生成し、転送用のDMA記述子のヘッダを生成し、かつデータが受信されたRx DMAチャンネルに対応する記述子リング112A~112NにDMA記述子を書き込む。加えて、Rx制御回路58は、IOM/IOB回路70に転送終了を信号通知する。転送終了の信号通知はIOM/IOB回路70に対して、そのチャンネルにはそれ以上のデータは供給されないことを通知する。未完了のキャッシュ・ラインがDMAによって更新されると、IOM/IOB回路70は更新されたデータがメモリに(例えば、ある実施形態において、インターコネクタ30での読み出し-修正-書き込みを避けるため、IOC26内に)格納される。

#### 【0075】

次に図8を参照すると、所定のTx DMAチャンネル用のTx制御回路56の一実施形態のオペレーションを示すフローチャートが示されている。Tx制御回路56は、並行して、かつ独立して動作する各Tx DMAチャンネルごとに図8に示されたオペレーションを実行する回路を含んでいる。容易に理解できるように、図8には特定の順序でブロックが示されているが、ブロックは図8に示されたオペレーションを実行する組み合わせの論理回路で並行して実行されてもよい。ある実施形態において、1つ又は複数のブロック又

10

20

30

40

50

はフローチャート全体が複数のクロック周期にわたってパイプライン接続される。

【 0 0 7 6 】

T x 制御回路 5 6 は、(チャンネルに対応する記述子リング 1 1 2 A ~ 1 1 2 N 内の)プリフェッチ用のチャンネルを利用可能であるか否か(決定ブロック 1 5 0)、かつ、チャンネルに記述子が必要であるか否か(決定ブロック 1 5 2)を判定する。少なくとも1つの記述子が利用可能であり、必要である場合は(決定ブロック 1 5 0、1 5 2の「yes」区間)、T x 制御回路 5 6 はホスト 1 2 内のメモリ内の記述子リング 1 1 2 A ~ 1 1 2 N から記述子を読み出す要求を生成する(ブロック 1 5 4)。

【 0 0 7 7 】

図 6 に関して上述したバッファ・ポインタが「利用可能」であるのと同様に、記述子は一般に、T x 制御回路 5 6 によって未だプリフェッチされていない対応する記述子リング 1 1 2 A ~ 1 1 2 N 内に記述子があれば「利用可能」である。ソフトウェアによって T x

DMA チャンネル用の記述子が記述子リング 1 1 2 A ~ 1 1 2 N 内に挿入され、ソフトウェアは、前述の方式のいずれかで(例えば記述子リングのエントリ内の有効ビットを使用し、記述子リングのカウントを増分するなどして)記述子が利用可能であることを指示する。同様に、記述子は多様な方式で「必要である」と見られる。例えば、T x DMA チャンネルが有効化され、そのチャンネル用にプリフェッチされた記述子がない場合は、記述子は「必要である」可能性がある。ある実施形態において、記述子を格納するスペースが I O M 2 4 及び / 又はバッファ 6 2 内にある限りは、T x 制御回路 5 6 は記述子をプリフェッチする。別の実施形態において、プリフェッチされるべきいくつかの記述子を示し、又はプリフェッチされるべき最小数と最大数の記述子を示すように T x 制御回路 5 6 がプログラムされる。T x 制御回路 5 6 は、プログラムされた数の記述子のプリフェッチを試みるために、記述子のためのプリフェッチ要求を生成する。

【 0 0 7 8 】

プリフェッチされた記述子が、読み出される I O M 2 4 内で利用可能である場合は、T x 制御回路 5 6 に I O M / I O B インターフェース回路 7 0 によってその旨が通知される。T x 制御回路 5 6 は、ある実施形態において、記述子の一部又は全部を I O M 2 4 からバッファ 6 2 へと読み込む。

【 0 0 7 9 】

T x 制御回路 5 6 は、DMA データが(チャンネル上で送信されるべき)プリフェッチ用に利用可能であるか否か(決定ブロック 1 5 6)、かつ、チャンネル用に DMA データが必要であるか否か(決定ブロック 1 5 8)を判定する。DMA データが利用可能であり、必要である場合は(決定ブロック 1 5 6 及び 1 5 8の「yes」区間)、T x 制御回路 5 6 はホスト 1 2 内のアドレス空間から(例えばホスト 1 2 内の記憶領域から)DMA データを読み出す要求を生成する(ブロック 1 6 0)。

【 0 0 8 0 】

T x 制御回路 5 6 が、処理されるべき記述子を有しており(例えば、記述子がチャンネル用の記述子リングから次に処理される予定の記述子である場合)、記述子データがバッファ 6 2 又は I O M 2 4 内にあり、かつ記述子データが実行される有効 DMA 転送を記述する場合は、DMA データをプリフェッチ用に利用可能であるとする。チャンネル上の以前の DMA データが既に送信される場合(又は間もなく送信される場合)は、DMA データが必要である。ある実施形態において、T x 制御回路 5 6 は、所定の任意の時点で幾つもの DMA データがプリフェッチされるべきかをプログラム可能であり、所望の量よりも少ない DMA データがプリフェッチされ、未だ送信されていない場合は、DMA データが必要である。ある実施形態において、DMA データが必要な場合は、送信チャンネル間の調停方式も影響を及ぼす。(例えば、チャンネルが比較的長時間にわたって調停に勝たない場合は、それが調停に勝つまで DMA データは送信できないので、DMA データは未だ必要ない。)

【 0 0 8 1 】

ブロック 1 5 6、1 5 8、1 6 0 により図示されるオペレーションは、ブロック 1 5 0

、152、154のオペレーションから（プリフェッチされた記述子がDMAデータが利用可能であるかどうかを決定するために使用されること以外では）独立していてもよいことに留意されたい。したがって、ブロック156、158、160を実行する回路は、ブロック150、152、154を実行する回路から独立していてもよく、それらの回路と並行して評価することができる。

#### 【0082】

前述のように、図8のオペレーションは、有効化されたTx DMAチャンネルごとに並行して行われもよい。2つ以上のプリフェッチ要求が同時に生成される場合は、Tx制御回路56はプリフェッチ要求の中から選択するための回路も含んでいる。例えば、Tx制御回路56は、最少数の記述子又は最少量のDMAデータが現在プリフェッチされ、準備ができるTx DMAチャンネルに対応するプリフェッチ要求を選択する。別の例として、Tx制御回路56は、現在プリフェッチされる記述子/DMAデータと、そのチャンネル用の所望の記述子数/DMAデータ量との最大の差をどのTx DMAチャンネルが有しているかに基づいて要求に重み付けする。さらにラウンドロビン又は優先度に基づく選択機構も利用してもよく、必要ならば、これらの機構はチャンネル間のプログラム可能な重みを含む。記述子とDMAデータとが、有効化されたチャンネルごとにプリフェッチされることを保証するため、チャンネルごとのタイムアウトなどの飢餓状態防止機構をも使用する。

#### 【0083】

プリフェッチされたDMAデータを読み出されるべきIOM24内で利用可能である場合は、IOM/IOBインターフェース回路70によってその旨がTx制御回路56に通知される。ある実施形態において、Tx制御回路56は、IOM24からのDMAデータの一部又は全部を読み出バッファ62へと読み込む。加えて、Tx制御回路56は、図9の一実施形態について図示されているように、プリフェッチされたDMAデータをターゲットに送信する。図9では理解し易いように、ブロックが特定の順序で示されているが、ブロックは図9に示されたオペレーションを実行する組み合わせの論理回路で並行して実行される。ある実施形態において、1つ又は複数のブロック又はフローチャート全体が複数のクロック周期にわたってパイプライン接続される。

#### 【0084】

Tx DMAチャンネルがループバック回路40（すなわち、より具体的には、ループバック回路40内のループバック・コンポーネント）に割り当てられると（決定ブロック162、「yes」区間）、Tx制御回路56は転送用に、かつDMAデータと共に送信されるようにDMA記述子から転送先バッファ・ポインタ（1つ又は複数）を抽出する（ブロック164）。Rx制御回路58にバッファのサイズが通知されるように、各転送先バッファ・ポインタのサイズ情報も提供される。ループバック回路40は転送先バッファ・ポインタ（1つ又は複数）を受け取り、メモリに書き込まれるべきデータ（例えばコピーDMA転送の場合はオリジナルのDMAデータ、又は、例えば暗号化が行われる場合は、変換されたDMAデータ）をバッファ・ポインタに提供している。あるいは、ループバックの結果がDMA記述子に書き込まれる場合（例えばハッシュ結果、CRC結果、又はチェックサム結果）は、DMA記述子へのポインタが提供される。Tx制御回路56は、チャンネル番号、（それがあある場合は）ポインタ、及び（それがあある場合は）制御情報と共にデータをターゲット・インターフェース回路16又はループバック・コンポーネント（ブロック166）に送信する。制御情報はDMA記述子からも抽出されてもよい（例えば、所望のループバック・オペレーション（1つ又は複数）を選択する情報、インターフェース回路用の制御情報など）。

#### 【0085】

図9のオペレーションはTx DMAチャンネルごとに並行して実行される。すなわち、Tx制御回路56はTx DMAチャンネルごとに並行して図9のオペレーションを実行する独立した回路を含んでいる。送信データ・パス上でインターフェース回路16とループバック回路40とに送信するチャンネルを選択するため、DMAデータを送信する必要があるチャンネル間で調停するための調停機構を使用する。

10

20

30

40

50

## 【 0 0 8 6 】

記述子リング 1 1 2 A ~ 1 1 2 N は、様々な記述子を格納することが前述されている。一実施形態において、記述子リング 1 1 2 A ~ 1 1 2 N は、（転送元から転送先への DMA 転送を記述する）転送記述子と、制御記述子の双方を格納する。制御記述子は DMA 転送を指定できないが、その代わりにチャンネルからの制御情報をエンコードできる。例えば、構造（例えば 1 つ又は複数の構造レジスタ 3 8 A ~ 3 8 G のコンテンツ）を指定するために制御記述子を使用する。したがって、DMA コントローラ 1 4、そのコンポーネント、又は実行中のシステム 0 の他のコンポーネントを DMA 転送間で再構成するために制御記述子を使用する。したがって、ソフトウェアは例えば、第 1 の構成で DMA 転送を実行するために 1 つ又は複数の転送記述子を構築し、構成を変更するために 1 つ又は複数の制

10

## 【 0 0 8 7 】

明確に DMA 転送を行わずに DMA コントローラ 1 4 を制御する、他のタイプの制御記述子も考えられる。例えば、タイム・トリガ記述子が処理される場合に DMA チャンネル上で時間遅延を生じさせるタイム・トリガ記述子が考えられる。例えば、システム 1 0 は、遅延値がプログラムされるタイマを含み、タイマが満了するまで記述子リングの次の記述子の処理を遅延させる。タイマの遅延が、DMA コントローラ 1 4 を「ウェイクアップ」させるため DMA コントローラ 1 4 に対するトリガ事象を生成する。様々な実施形態で、ユーザーはタイマに遅延値をプログラムしてもよく、又はタイム・トリガ記述子に遅延値を含める。別の実施形態において、DMA コントローラ 1 4 がタイマを実装する。タイム・トリガ記述子は多様な用途を有している。例えば、タイム・トリガ記述子に転送元のアドレス空間から転送先のアドレス空間へとコピーするコピー DMA 記述子をインターリーブしている。転送元のアドレス空間は 1 つ又は複数のメモリ・マップド I/O レジスタ、又は他のシステム・ステータス・レジスタを備えている。したがって、タイム・トリガ記述子によって遅延されて、レジスタを読み出し、結果を転送先に書き込むために反復コピー DMA 転送を使用する。コピー DMA 転送はプロセッサ 1 8 A、1 8 B 上のソフトウェア・ポーリング・ループに代わって、プロセッサを他のオペレーション用に解放する。

20

30

## 【 0 0 8 8 】

図 1 0 はチャンネル 1 0 に対応する記述子リング 1 1 2 A の一実施形態のブロック図であり、記述子リング内の転送記述子内の制御記述子の埋め込みの例を示している。必要ならば図 1 0 の方式で、（連続的に、又はリング内の異なるポイントで）2 つ以上の制御記述子を埋め込んでもよい。

## 【 0 0 8 9 】

図 1 0 では、転送記述子 1 7 0 A、1 7 0 B が示され、続いて制御記述子 1 7 2 が、続いてさらに 2 つの転送記述子 1 7 0 C、1 7 0 D が示されている。したがって、例えば記述子 1 7 0 A、1 7 0 B によって指定される 2 つの DMA 転送は、T x 制御回路 5 6 によって実行され、続いて制御記述子 1 7 2 の処理が行われよう。制御記述子 1 7 2 の処理に

40

## 【 0 0 9 0 】

図 1 1 は、T x DMA チャンネルに対応する記述子リング内に転送記述子が埋め込まれる制御記述子用に T x 制御回路 5 6 の一実施形態を使用する、追加処理を示すフローチャートである。T x 制御回路 5 6 は、並行してかつ独立して動作する T x DMA チャンネルごとに図 1 1 に示されたオペレーションを実行する回路を含んでいる。容易に理解できるように、図 1 1 には特定の順序でブロックが示されているが、ブロックは図 1 1 に示されたオペレーションを実行する組み合わせの論理回路で並行して実行されてもよい。ある実施形態において、1 つ又は複数のブロック又はフローチャート全体が複数のクロック周期にわたってパイプライン接続される。

50

## 【 0 0 9 1 】

T x 制御回路 5 6 は、(例えば図 8 に関して記載したように)記述子がチャンネル用に利用可能であり、必要であると想定して、記述子リングから記述子をプリフェッチする(ブロック 1 7 4)。T x 制御回路 5 6 は、記述子が T x 制御回路 5 6 に戻るとこれを処理して、記述子が制御記述子であるか否かを判定する(決定ブロック 1 7 6)。例えば、記述子ヘッダは、記述子が転送記述子であるか、制御記述子であるかを特定するタイプ・フィールドを含んでいる。記述子が転送記述子である場合は(決定ブロック 1 7 6、「no」区間)、T x 制御回路 5 6 は転送記述子を処理する(ブロック 1 7 7)。例えば、転送記述子の処理は、図 8 及び 9 に関する上記の記述子と同様でよい。

## 【 0 0 9 2 】

記述子が制御記述子である場合は(決定ブロック 1 7 6、「yes」区間)、T x 制御回路は、制御記述子がタイム・トリガ記述子であるか否かを判定する(決定ブロック 1 7 8)。制御記述子のヘッダ内でのエンコードを介して、タイム・トリガ記述子が表示される。あるいは、別の構造記述子と同様に、タイム・トリガ記述子が構成レジスタ 3 8 A ~ 3 8 G にロードされる値を指定する。ロードされる値は所望の遅延を指定する。T x 制御回路 5 6 は、ロードされる構成レジスタのレジスタ・アドレスをデコードすることによってタイム・トリガを検知する。制御記述子がタイム・トリガ記述子である場合は(決定ブロック 1 7 8、「yes」区間)、T x 制御回路 5 6 は、タイム・トリガ記述子によって指定された遅延が満了するまで、記述子リングからの次の記述子の処理を遅延させる(ブロック 1 8 0)。制御記述子がタイム・トリガ記述子ではない場合は(決定ブロック 1 7 8、「no」区間)、T x 制御回路 5 6 は制御記述子の値を使用して再構成する(ブロック 1 8 2)。例えば、制御記述子は、構成レジスタ 3 8 A ~ 3 8 G を特定するレジスタ・アドレスと、構成レジスタに書き込む値とを備えている。T x 制御回路 5 6 が書き込みを行わせる。あるいは、制御記述子は、レジスタ・アドレスのリストを格納するメモリ・バッファへのバッファ・ポインタ、及びこれらのレジスタに書き込まれる値を含んでいる。したがって、制御記述子を使用して比較的大量の構成が実行される。ある実施形態において、制御記述子を使用することによって、プロセッサ 1 8 A、1 8 B で実行される、各構成レジスタへの書き込み命令を無くす。

## 【 0 0 9 3 】

図 1 2 から 1 6 は、DMA コントローラ 1 4 の一実施形態による様々なタイプの記述子の例を示す。一般に、記述子はヘッダ、場合によって、結果(例えばループバック回路 4 0 によって生成される結果)を格納するデータ・フィールド、及び DMA データを格納するバッファを指す 1 つ又は複数のバッファ・ポインタ(転送元バッファ・ポインタ)又は DMA データを格納するために使用されてもよいバッファを指す 1 つ又は複数のバッファ・ポインタ(転送先バッファ・ポインタ)を備えている。

## 【 0 0 9 4 】

本実施形態において、記述子は、それらが DMA を受信するのか送信するのかに基づいて、又は、選択された場合にループバック回路 4 0 によって実行される関数に基づいて変化する。受信記述子は R x DMA 転送用に使用され、他のタイプの記述子は T x DMA 転送及びループバック機能によって使用される。DMA コントローラ 1 4 (及び、より具体的には、一実施形態では T x 制御回路 5 6)は、所定の T x DMA チャンネルのインターフェース回路 1 6 への、又はループバック回路 4 0 内の機能への割り当てに基づいて、そのチャンネル用の記述子リング内の記述子のフォーマットを決定する。

## 【 0 0 9 5 】

図 1 2 から 1 6 には様々なフィールド(例えば、特にヘッダ・フィールド)が詳細に示されている。図 1 2 から 1 6 にはある一定の情報が示されているが、図示された情報に加えた、又は図示された情報のサブセットに加えた、又は図示された情報の代替としての他の情報を除外することを意図するものではない。様々な実施形態で、所望に応じて様々な追加の情報が含まれる。

## 【 0 0 9 6 】

10

20

30

40

50

図12は、受信記述子190の一実施形態のブロック図である。受信記述子190は、R×制御回路58によって生成され、R×DMAチャネルに対応する記述子リング112A~112Nに書き込まれる記述子のフォーマットである。図12の実施形態において、受信記述子190は、(一実施形態について図12に分解図で示されている)ヘッダ・フィールド192と、場合によるハッシュ結果フィールド194と、1つ又は複数のバッファ・ポインタ・フィールド196A~196Nとを含んでいる。各バッファ・ポインタ・フィールド196A~196Nは、バッファのサイズによってエンコードされるサイズ・フィールドと、バッファへのポインタでエンコードされるポインタ・フィールドとを含む。R×制御回路58は、受信されたDMAデータを格納するためにバッファ・ポインタ・リング114A~114Mから使用されるバッファ・ポインタでバッファ・ポインタ・フィールド196A~196Nを書き込むように構成される。

10

## 【0097】

受信ヘッダ・フィールド192の分解図は、タイプ・フィールド192A、スタイル・フィールド192B、ハッシュ受信フィールド192C、バッファ・カウント・フィールド192D、パケット長フィールド192E、ステータス・フィールド192Fを含んでいる。タイプ・フィールド192Aは、例えば制御記述子又は転送記述子のような記述子のタイプをエンコードする。タイプ・フィールド192A(及び他の記述子の場合の、下記の他の同様のタイプ・フィールド)は、図16に示された制御記述子を除く記述子を転送記述子として特定する。スタイル・フィールド192Bは、バッファ・ポインタ・フィールド196A~196N内のバッファ・ポインタが転送先と転送先の双方のポインタを含むのか、又は(転送元がR×DMA転送用のインターフェース回路であるため)転送先のポインタだけを含むのかを意味する記述子のスタイルをエンコードする。ハッシュ受信フィールド192Cは、場合によるハッシュ結果フィールド194が記述子内に含まれるか否かを示すために使用される。バッファ・カウント・フィールド192Dは、受信されたDMAデータを格納するために使用されるバッファ数のカウントを、ひいてはバッファ・ポインタ・フィールド196A~196Nの数でエンコードされる。パケット長フィールド192Eは、バッファ内に(例えばバイトで)格納されるパケットの長さでエンコードされる。ステータス・フィールド192Fは、R×制御回路58によって生成されるステータス(例えばエラー表示)、及び/又は転送元インターフェース回路16によって提供されるステータスを含む、転送の様々なステータスを含んでいる。R×制御回路58は、DMA転送の終了時に受信ヘッダ192を書き込む。

20

30

## 【0098】

図13は、送信記述子200の一実施形態のブロック図である。送信記述子200は、インターフェース回路16への、特にMAC34A、34BへのTx DMA転送用にTx制御回路56によって使用される記述子フォーマットであり、インターフェース回路16に割り当てられるTx DMAチャネルに対応する記述子リング112A~112Nにソフトウェアによって書き込まれる。図13の実施形態において、送信記述子200は、(一実施形態について図13に分解図で示されている)ヘッダ・フィールド202、及び1つ又は複数のバッファ・ポインタ・フィールド204A~204Nを含んでいる。各バッファ・ポインタ・フィールド204A~204Nは、バッファのサイズでエンコードされるサイズ・フィールドと、バッファへのポインタでエンコードされるポインタ・フィールドとを含む。Tx制御回路56を、DMAデータを送信用にバッファからプリフェッチするためにバッファ・ポインタ・フィールド204A~204Nを読み出すように構成する。

40

## 【0099】

送信ヘッダ・フィールド202の分解図は、タイプ・フィールド202A、スタイル・フィールド202B、MAC構成フィールド202C、パケット長フィールド202D、パケット情報フィールド202Eを含む。タイプ・フィールド202Aとスタイル・フィールド202Bは上記のタイプ・フィールド192Aとスタイル・フィールド192Bと同様のものでよい。MAC構成フィールド202Cは、DMA転送によってターゲットに

50

されるMAC34A、34B用の様々なパケット特有のMAC構成情報でエンコードされる。例えば、MAC構成フィールド202Cは、仮想ローカル・エリア・ネットワーク(VLAN)構成(例えば、無し、挿入、除去、又は修正)、CRC構成(例えば、無し、挿入CRC、パッドCRC、修正CRC)、及びMAC転送元アドレスを修正するか否かを含む。パケット長フィールド202Dは、(例えばバイトで)バッファ内に格納されるパケットの長さでエンコードされる。パケット情報フィールド202Eは、パケットを記述する様々な情報(例えばIPヘッダ長、イーサネットのヘッダ長、パケットのタイプ(TCP/UDP)、チェックサム有効化など)でエンコードされる。

#### 【0100】

図14はコピー記述子210の一実施形態のブロック図である。コピー記述子210は、コピーFIFO42を使用する(ホスト12内の1つの記憶領域からホスト12内の別の記憶領域への)コピーDMA転送用のTx制御回路56によって使用される記述子のフォーマットである。したがって、コピー記述子210は、コピーFIFO42に割り当てられるTx DMAチャネルに対応する記述子リング112A~112Nで使用される。図14の実施形態において、コピー記述子210は、(一実施形態について図14に分解図で示されている)ヘッダ・フィールド212、及び1つ又は複数のバッファ・ポインタ・フィールド214A~214Nを含んでいる。各バッファ・ポインタ・フィールド214A~214Nは、バッファのサイズでエンコードされるサイズ・フィールドと、バッファへのポインタでエンコードされるポインタ・フィールドとを含んでいる。加えて、この実施形態において、各バッファ・ポインタ・フィールド214A~214Nは、(転送元DMAデータを格納するバッファの位置を特定する)転送元ポインタ、又は(DMAデータが格納されるべきバッファの位置を特定する)転送先ポインタのいずれかとしてポインタを特定する転送元/転送先(S/D)フィールドを含んでいる。送信用の転送元バッファからDMAデータをプリフェッチし、Rx制御回路58への送信用にコピー記述子FIFO42に転送先ポインタを提供するために、バッファ・ポインタ・フィールド214A~214Nを読み出すように、Tx制御回路56を構成する。

#### 【0101】

一実施形態において、コピー記述子210内の所定の転送先ポインタに2つ以上の転送元ポインタがある。DMAコントローラ14は、転送元バッファからのデータをコピー記述子210内にリストされた順序で転送先バッファにコピーする。したがって、DMAコントローラ14は複数の記憶領域から拡散したデータをコピー・オペレーションで転送先の記憶領域へと収集することをサポートしている。同様に、一実施形態において、コピー記述子210内の所定の転送元ポインタに2つ以上の転送先ポインタがある。このような実施形態において、転送元バッファからのデータの拡散がサポートされる。

#### 【0102】

送信ヘッダ・フィールド212の分解図は、タイプ・フィールド212A、スタイル・フィールド212B、転送元タイプ・フィールド212C、転送先タイプ・フィールド212D、論理ブロック長フィールド212Eを含んでいる。タイプ・フィールド212Aとスタイル・フィールド212Bは、上記のタイプ・フィールド192Aとスタイル・フィールド192Bと同様のものでよい。転送元タイプ・フィールド212Cと転送先タイプ・フィールド212Dは、DMA転送の進展と共に転送元バッファ・ポインタ(1つ又は複数)と転送先バッファ・ポインタ(1つ又は複数)をいかにして修正するべきかを指示するためにエンコードされる。例えば、各バッファ・ポインタは、一実施形態では以下のタイプのうちの1つである。すなわち、逐次増分、逐次減分、又は(例えば1、2、4、8、又は16バイトなどの様々な固定幅を有する)固定タイプである。逐次増分は、各データ送信後に、送信されたデータ量だけ増分されることを示している。逐次減分は同様であるが、アドレスが減分される。逐次増分又は逐次減分は、データが逐次記憶領域に書き込まれる記憶領域用に使用される。固定タイプのオプションは、アドレスがレジスタ又はデバイスのポートにマッピングされるメモリである場合に使用され、幅はレジスタ/デバイスへの各送信幅である。転送元タイプ・フィールド212Cもゼロ用のエンコーデ

10

20

30

40

50

ングを有し、ゼロ・ブロックを転送先に書き込むために使用される。転送先タイプ・フィールド 2 1 2 D も、転送元 DMA データはプリフェッチされるが、転送先には書き込まれない、プリフェッチ用のみのエンコーディングを有している。ある実施形態において、論理ブロック長フィールドが複数の DMA 記述子に亘る論理 DMA ブロックの長さを示すために使用される。すなわち、論理 DMA オペレーションは実際に複数の記述子を使用して指定され、論理 DMA ブロックの長さは、論理 DMA オペレーションの長さ（例えば複数の記述子に亘るデータ転送の総計）である。

#### 【 0 1 0 3 】

XOR 回路 4 6 は、送信 DMA 記述子 2 0 0 と同様の記述子を使用する。複数のチャンネルが XOR 回路 4 6 に割り当てられ、各チャンネル内の記述子が XOR 転送元の 1 つを指定する。第 1 のチャンネルが、XOR 結果の転送先（1 つ又は複数の転送先バッファ）をも指定する。

10

#### 【 0 1 0 4 】

図 1 5 はオフロード記述子 2 2 0 の一実施形態のブロック図である。オフロード記述子 2 2 0 は、オフロード・エンジン 4 4 を指定する DMA 転送用に T x 制御回路 5 6 によって使用される記述子のフォーマットである。したがって、オフロード記述子 2 2 0 は、オフロード・エンジン 4 4 に割り当てられる T x DMA チャンネルに対応する記述子リング 1 1 2 A ~ 1 1 2 N 内で使用される。図 1 5 の実施形態において、オフロード記述子 2 2 0 は、（一実施形態について図 1 5 に分解図で示されている）ヘッダ・フィールド 2 2 2 、オプションとしての結果保存フィールド 2 2 4 、及び 1 つ又は複数のバッファ・ポインタ・フィールド 2 2 6 A ~ 2 2 6 N を含む。各バッファ・ポインタ・フィールド 2 2 6 A ~ 2 2 6 N は、バッファのサイズによってエンコードされるサイズ・フィールドと、バッファへのポインタでエンコードされるポインタ・フィールドとを含んでいる。加えて、この実施形態において、各バッファ・ポインタ・フィールド 2 2 6 A ~ 2 2 6 N は、（転送元 DMA データを格納するバッファの位置を特定する）転送元ポインタ、又は（DMA データが格納されるべきバッファの位置を特定する）転送先ポインタのいずれかとしてポインタを特定する転送元 / 転送先（S / D）フィールドを含んでいる。T x 制御回路 5 6 を、転送元バッファから DMA データをプリフェッチし、それがあ場合は転送先バッファを特定するために、バッファ・ポインタ・フィールド 2 2 6 A ~ 2 2 6 N を読み出すように構成する。変換された DMA データがオフロード・エンジン 4 4 の結果である場合は、変換された DMA データ用の転送先ポインタがある。（例えば結果保存フィールド 2 2 4 に格納するために）結果が DMA データとは別個に生成される場合は、場合によっては転送先ポインタがなくてもよく、DMA データが転送先に書き込まれない。

20

30

#### 【 0 1 0 5 】

オフロード・ヘッダ・フィールド 2 2 2 の分解図は、タイプ・フィールド 2 2 2 A 、スタイル・フィールド 2 2 2 B 、結果保存フィールド 2 2 2 C 、暗号化モード・フィールド 2 2 2 D 、機能数フィールド 2 2 2 E 、論理ブロック長フィールド 2 2 2 F 、オフロード制御フィールド 2 2 2 G を含んでいる。タイプ・フィールド 2 2 2 A とスタイル・フィールド 2 2 2 B は、上記のタイプ・フィールド 1 9 2 A とスタイル・フィールド 1 9 2 B と同様のものでよく、論理ブロック長フィールド 2 2 2 F は上記の論理ブロック長フィールド 2 1 2 E と同様のものでよい。結果保存フィールド 2 2 2 C は、結果保存フィールド 2 2 4 がオフロード・ヘッダ記述子 2 2 0 内に含まれているか否かを示すためにエンコードしてもよく、さらに結果保存フィールド 2 2 4 のサイズ（例えば一実施形態において、6 4 ビット、1 9 2 ビット、又は 3 2 0 ビット）をも表示している。結果保存フィールド 2 2 4 は、結果が変換された DMA データではない場合、又は変換された DMA データに加えて結果が生成された場合は、オフロード・エンジン 4 4 によって生成された結果を格納するために使用されている。暗号化モード・フィールド 2 2 2 D は、データがある場合はそれを処理するため、オフロード・エンジン 4 4 用に必要なモードでエンコードされる。暗号化モードは、無し、シグニチャのみ（例えば生成器 8 6 又は 8 8 内の CRC 又はチェックサム）、暗号化のみ、復号化のみ、（ハッシュが先、又は暗号化が先のオプションを

40

50

伴う)暗号化とハッシュ、又は(ハッシュが先、又は復号化が先のオプションを伴う)復号化とハッシュを含んでいる。機能数フィールド222Eは、セキュリティ回路84A~84Dを使用するそれらのモードの機能数(例えば、前述のようにセキュリティ回路84A~84Dから形成された8つの論理機能の1つ)でエンコードされる。オフロード制御フィールド222Gは、DMA転送用の追加の制御情報を含んでいる。例えば、各々のシグニチャ・ヘッダ、暗号化ヘッダ、暗号化トレーラ、ハッシュ・サイズの長さがオフロード制御フィールド222Gに含まれている。同様に、選択された暗号化/復号化(暗号)アルゴリズム、ハッシュ・アルゴリズム、ブロック暗号モードがオフロード制御フィールド222G内でエンコードされる。オフロード制御フィールド222G内に様々な他の制御ビットも含まれる。

10

## 【0106】

図16は制御記述子230の一実施形態のブロック図である。制御記述子230は、いずれかのDMAチャネル内にTx制御回路56によって使用される制御記述子のフォーマットでよい。図16の実施形態において、制御記述子230は(一実施形態について図16に分解図で示されている)ヘッダ・フィールド232と、データ又はサイズ/ポインタ・フィールド234とを含んでいる。データ又はサイズ/ポインタ・フィールド234は、構成レジスタ38A~38Gに書き込まれるインライン・データを格納してもよく、又はレジスタ・アドレスのリスト、及びレジスタに書き込まれる構成データを格納するバッファを指すバッファ・ポインタ(及びバッファ・サイズ)を格納する。

## 【0107】

20

制御ヘッダ・フィールド232の分解図は、タイプ・フィールド232A、インライン/ポインタ・フィールド232B、インターフェース/機能特定(ID)フィールド232C、レジスタ・アドレス・フィールド232Dを含んでいる。タイプ・フィールド232Aは、上記のタイプ・フィールド192Aと同様のものによく、記述子230を制御記述子として特定する。インライン/ポインタ・フィールド232Bは、データ又はサイズ/ポインタ・フィールド234が構成レジスタに書き込まれるインライン・データ、及びレジスタ・アドレスと構成データのバッファへのポインタ、又は事象を格納するか否かを表示するためにエンコードされる。インターフェース/機能IDフィールド232Cは、構成データ(例えばインターフェース回路、オフロード・エンジン44内の機能など)のターゲットを特定するためにエンコードされる。この実施形態において、レジスタ・アドレスはインターフェース/機能IDにローカルであってよい。レジスタ・アドレス・フィールド232Dは、制御記述子230がインライン・データを有している場合はレジスタ・アドレスでエンコードされる。

30

## 【0108】

次に図17を参照すると、チェックサム生成器88の一実施形態のブロック図が示されている。図17の実施形態において、チェックサム生成器88は、複数の16ビットの3:2桁上げ保存加算器(CSA)240A~240G、全加算器242、16ビットのアクキュレータ(ACC)244を含んでいる。チェックサム生成器88は、(例えばTx制御回路56からの)128ビット(16バイト)入力を受けると結合されている。128ビット入力は、CSA240A~240Cへの入力として供給される16ビット部分に分割される。加えて、アクキュレータ244の出力はCSA240Cへの入力として結合されている。各CSA240A~240Cは、図17では「S」と「C」の記号が付された和と桁上げ項を出力する。CSA240A~240Cの和と桁上げ出力はCSA240Dと240Eに入力される。CSA240Dの和と桁上げ出力、及びCSA240Eの桁上げ出力はCSA240Fに入力される。CSA240Eの和出力、及びCSA240Fの和と桁上げ出力はCSA240Gに入力される。CSA240Gの和と桁上げ出力は全加算器242に入力され、これがアクキュレータ244に格納される総和を生成する。

40

## 【0109】

CSAはNビット入力を受け取り、桁上げを位置から位置に波及させない和と桁上げ出

50

力項を生成する。すなわち、総和出力ビットは、他のビット位置のビットに依存しない、そのビット位置での入力の総和である。各総和ビットはそのビット位置での入力ビットの排他的論理和 (XOR) である。桁上げ出力ビットは、別のビット位置の桁上げに依存しない所定のビット位置の総和からの桁上げである。オペレーション上は、桁上げ項は次の最上位ビット位置への桁上げ (キャリー・イン) であるとされる。したがって、最下位ビット位置での桁上げビットはゼロであり、NビットのCSAからの論理的N+1の桁上げビットがある。

【0110】

矢印246で示されるように、CSAからの桁上げ項出力が別のCSA又は全加算器242への入力として供給される各ポイントで、最上位桁上げビットは、桁上げ項の最下位ビットへと折り返される。すなわち、桁上げ項の最上位桁上げビットは最上位ビット位置から削除され、常にゼロである最下位ビットの代わりに最下位ビット位置に挿入される。このようにして、各CSAと全加算器242への入力は常に16ビットであり、全ての桁上げビットが入力に表わされる。図17に示された16ビットCSAの場合は、桁上げ項出力の値は論理的に、最下位ビットが0である17ビットである。桁上げ項出力は以下のように次のCSA (又は全加算器242) に入力される。

$$\text{In}[15:0] = \text{桁上げ}[15:1] \parallel \text{桁上げ}[16] \text{ (及び桁上げ}[0]=0 \text{は切り捨て)}$$

【0111】

最上位桁上げビットの最下位桁上げビットへの折り返しは、1つのCSAの桁上げ項出力と次のCSAへの入力との間での線配線を介して達成される。

【0112】

ある実施形態において、図17に示されているようなチェックサム生成器88は、1回のパスで16ビットのチェックサムを生成でき、チェックサムをアキュムレータ244内に格納する。この実施形態において、桁上げが折り返され、各々128ビットの入力がチェックサム内に累積されるとCSA240A~240G、及び全加算器242によって加算されるので、アキュムレータ244も16ビットである。

【0113】

チェックサムの生成のため、1の補数和が生成される。CSAによる最上位ビットから最下位ビットへの桁上げビット出力の折り返しが、正しい1の補数和を生成するためであることが示される。

【0114】

CSA240A~240Gは各レベルのCSAのセットであるとしている。第1のレベルのCSA (CSA240A~240C) は、チェックサム生成器への入力に結合された入力と、アキュムレータの出力に結合された入力とを有している。別のレベルでは、入力は先行するレベル (又はCSA240Gの場合は複数のレベル) でCSAの出力に結合される。各レベルで、先行レベルからの桁上げ出力は、そのレベルに入力されるべき最下位ビットに折り返される最上位ビットを有している。

【0115】

この実施形態は、TCP、IP、及びICMPで使用される16ビットのチェックサムを生成するための16ビットCSAを使用する。別の実施形態は、容量がより大きい、又はより小さいチェックサムを使用し、このような実施形態でより大型の、又は小型のCSAを使用する。したがって、一般にNビット3:2CSAが使用されている。同様に、図示した実施形態では128ビットの入力が使用されているが、別の実施形態は任意のビット数の入力を使用する。CSAのレベルの数はそれに従って変更されている。

【0116】

図17の実施形態はチェックサム生成器88で実装されるものとして示されているが、部分的チェックサムを完全なものにするため、(前述のように)部分的チェックサムの生成用のIOB22で、またインターフェース回路16で同様の回路を使用できる。

【0117】

次に図18を参照すると、全加算器242の一実施形態のブロック図が示されている。

図示された実施形態において、全加算器 242 は XOR 回路 250 と、一对の桁上げ生成回路 252 A、252 B とを含んでいる。XOR 回路 250 と桁上げ生成回路 252 A、252 B は、(図 17 の CSA 240 G の和と桁上げ出力であり、桁上げ出力の最上位ビットが除去され、最下位ビットの位置に挿入される) 全加算器への「a」及び「b」入力を受け取るように結合されている。桁上げ生成器 252 A は、ゼロである桁上げ(キャリー・イン)( $C_{in}$ ) 入力を受け取るように結合され、桁上げ生成器 252 B は、1 の桁上げ入力を受け取るように結合されている。桁上げ生成器 252 A の桁上げ(キャリー・アウト)( $C_{out}$ ) 出力は、入力として桁上げ生成器 252 A、252 B の桁上げ出力を有するマルチプレクサ(mux) 254 の選択制御として提供される。mux 254 の出力は、これも入力として XOR 回路 250 の出力を有する第 2 の XOR 回路 256 への入力として結合されている。XOR 回路 256 の出力は全加算器 242 の出力である。

10

#### 【0118】

XOR 回路 250 は、a 及び b 入力をビットごとに排他的論理和し、CSA 加算器 240 と同様の別の総和項の生成を効率的に行う。桁上げ生成器 252 A、252 B は a 及び b 入力上で真の桁上げ生成を実行し、桁上げ生成器 252 A の桁上げ(キャリー・アウト) がゼロである場合は、桁上げ生成器 252 A の出力は XOR 回路 250 からの総和項と排他的論理和するための mux 254 を介して選択されることによって、最終的な 1 の補数和を生成する。これに対して、桁上げ生成器 252 A の桁上げ(キャリー・アウト) が 1 である場合は、桁上げ生成器 252 B の出力は mux 254 を介して選択される。桁上げ生成器 252 B への桁上げ(キャリー・イン) は 1 であるので、桁上げ生成器 252 B は、最上位ビットからの桁上げ(キャリー・アウト) を効果的に再び折り返す。別の観点から見ると、桁上げ生成器 252 A、252 B は、桁上げが生成されるか否かに基づいてもう一つの桁上げビットを選択的に折り返している。

20

#### 【0119】

上述の開示が完全に理解されれば、当業者には多くの変化形態及び修正形態が明らかである。以下の請求項はこのような変化形態及び修正形態を全て包含することを意図するものである。

#### 【図面の簡単な説明】

#### 【0120】

【図 1】システムの一実施形態のブロック図である。

30

【図 2】図 1 に示された DMA コントローラの一実施形態のブロック図である。

【図 3】図 2 に示されたオフロード・エンジンの一実施形態のブロック図である。

【図 4】図 1 のシステムにおける DMA の一実施形態のブロック図である。

【図 5】記述子リングとバッファ・ポインタ・リングの一実施形態のオペレーションを示すブロック図である。

【図 6】図 2 に示された受信プリフェッチ・エンジンの一実施形態のオペレーションを示すフローチャートである。

【図 7】図 2 に示された受信制御回路の一実施形態のオペレーションを示すフローチャートである。

【図 8】図 2 に示された送信制御回路の一実施形態のプリフェッチ・オペレーションを示すフローチャートである。

40

【図 9】図 2 に示された送信制御回路の一実施形態の送信オペレーションを示すフローチャートである。

【図 10】送信記述子に含まれる制御記述子を有する記述子リングを示すブロック図である。

【図 11】制御記述子の処理の一実施形態を示すフローチャートである。

【図 12】受信 DMA 記述子の一実施形態を示すブロック図である。

【図 13】送信 DMA 記述子の一実施形態を示すブロック図である。

【図 14】コピー DMA 記述子の一実施形態を示すブロック図である。

【図 15】オフロード DMA 記述子の一実施形態を示すブロック図である。

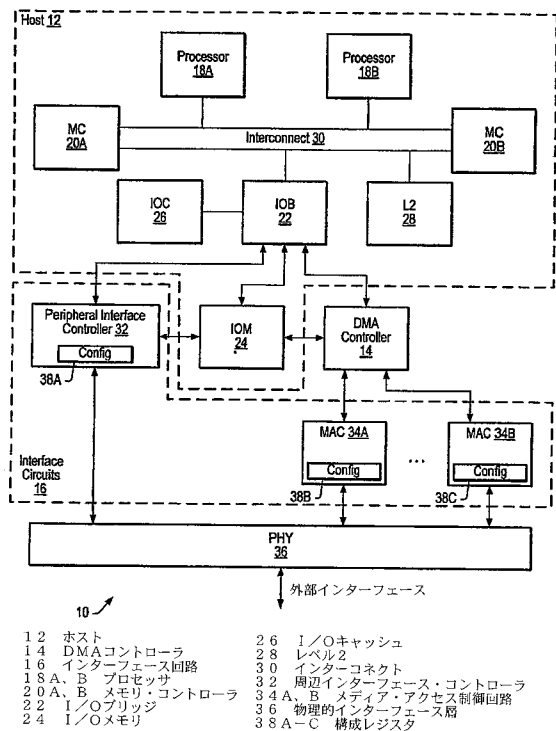
50

【図16】制御記述子の一実施形態を示すブロック図である。

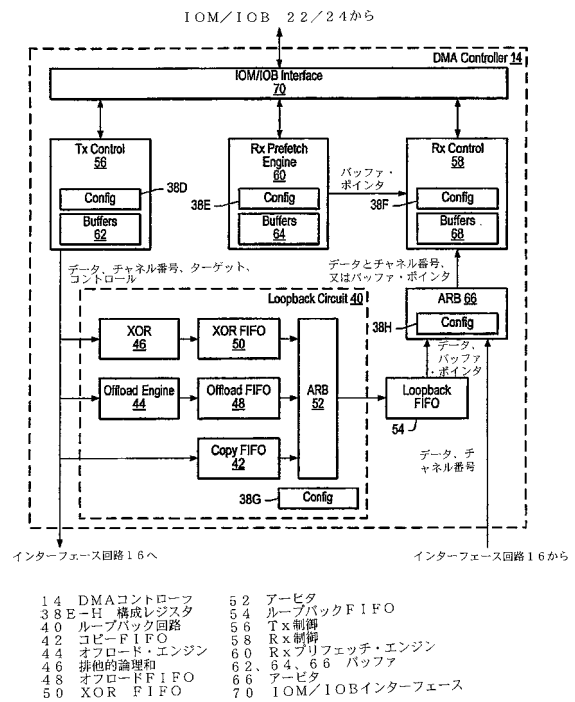
【図17】図3に示されたチェックサム生成器の一実施形態を示すブロック図である。

【図18】図17に示された全加算器の一実施形態を示すブロック図である。

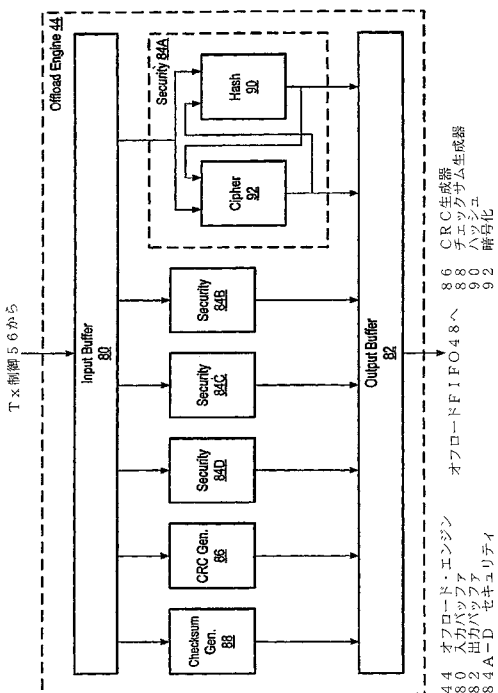
【図1】



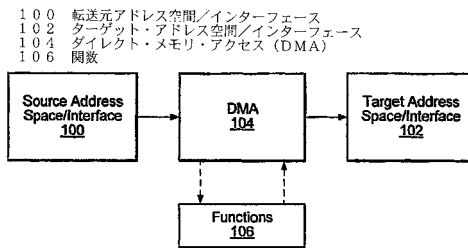
【図2】



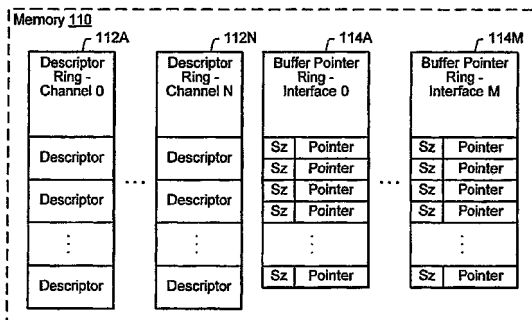
【図 3】



【図 4】



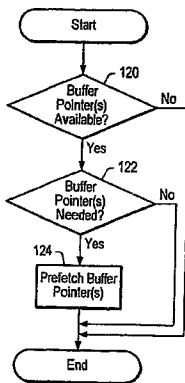
【図 5】



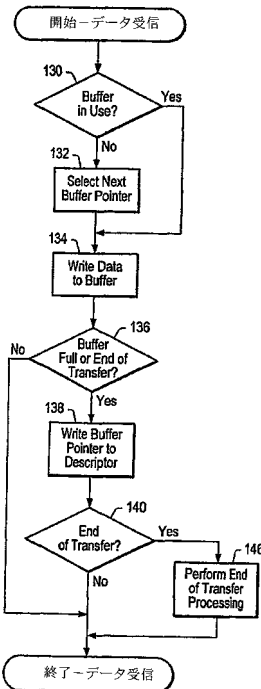
110 メモリ  
 112A 記述子リング チャンネル0  
 112N 記述子リング チャンネルN  
 114A バッファ・ポインタ・リング インターフェース0  
 114M バッファ・ポインタ・リング インターフェースM  
 Descriptor: 記述子  
 Pointer: ポインタ

【図 6】

120 バッファ・ポインタ (1つ又は複数) は利用可能か?  
 122 バッファ・ポインタ (1つ又は複数) が必要か?  
 124 バッファ・ポインタ (1つ又は複数) をプリフェッチ

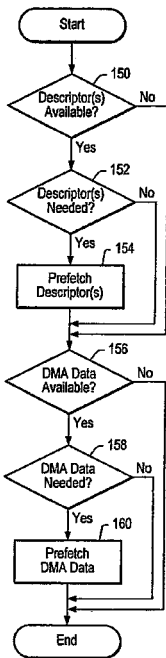


【図 7】



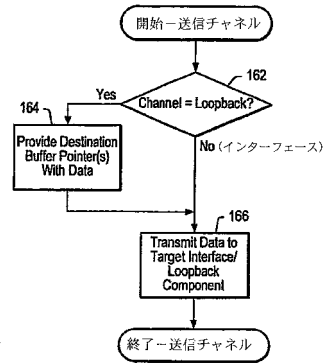
130 バッファを使用中か?  
 132 次のバッファ・ポインタを選択  
 134 バッファにデータを書き込む  
 136 バッファはフル状態か、又は転送終了か?  
 138 記述子にバッファ・ポインタを書き込む  
 140 転送終了か?  
 146 転送終了の処理を実行

【 図 8 】



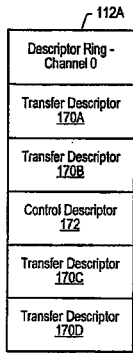
- 1 5 0 記述子 (1つ又は複数) は利用可能か?
- 1 5 2 記述子 (1つ又は複数) が必要か?
- 1 5 4 記述子 (1つ又は複数) をプリフェッチ
- 1 5 6 DMAデータを利用可能か?
- 1 5 8 DMAデータが必要か?
- 1 6 0 DMAデータをプリフェッチ

【 図 9 】



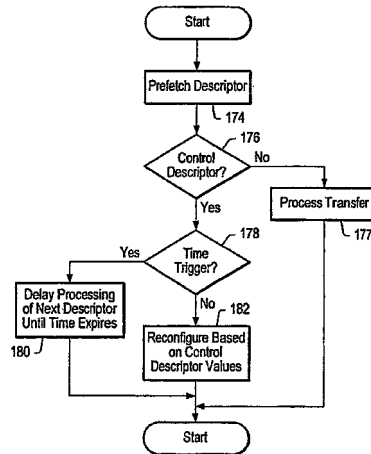
- 1 6 2 チャンネル=ループバックか?
- 1 6 4 転送先バッファ・ポインタ (1つ又は複数) にデータを供給
- 1 6 6 ターゲット・インターフェース/ループバック・コンポーネントにデータを送信

【 図 1 0 】



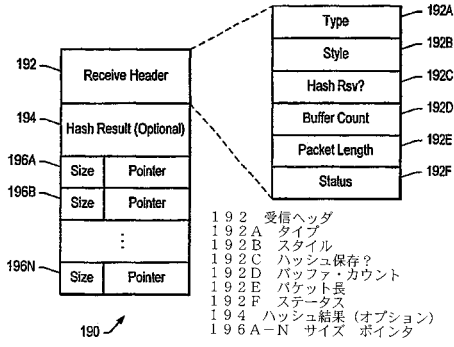
- 1 1 2 A 記述子リング-チャンネル 0
- 1 7 0 A-D 転送記述子
- 1 7 2 制御記述子

【 図 1 1 】

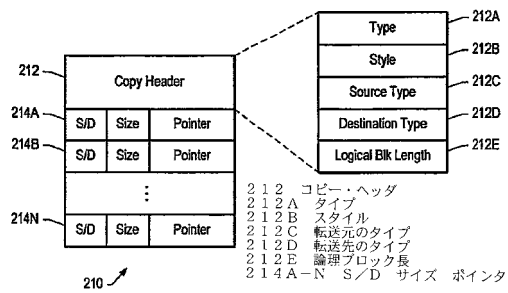


- 1 7 4 記述子をプリフェッチ
- 1 7 6 制御記述子か?
- 1 7 7 転送処理
- 1 7 8 制御記述子か?
- 1 8 0 時間が満了するまで次の記述子の処理を遅延
- 1 8 2 制御記述子の値に基づいて再構成

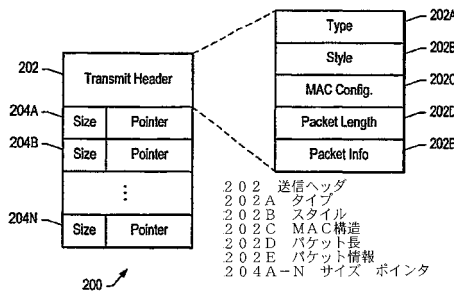
【図12】



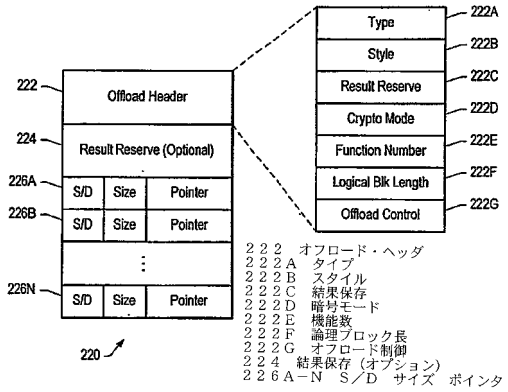
【図14】



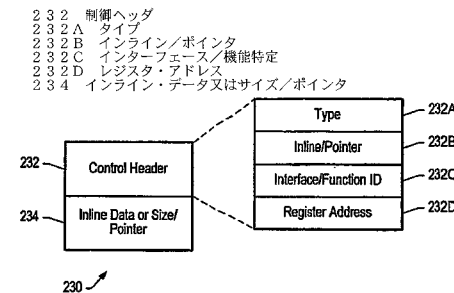
【図13】



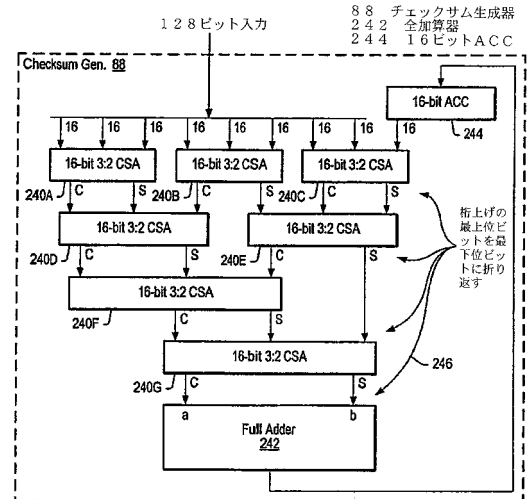
【図15】



【図16】

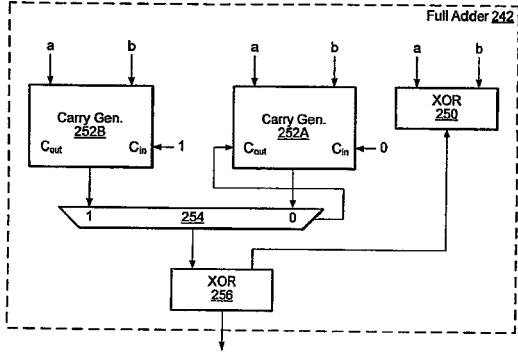


【図17】



【 図 18 】

242 全加算器  
250 排他的論理和  
252A 桁上り生成器  
252B 桁上り生成器  
254 排他的論理和



---

フロントページの続き

- (72)発明者 ハイター, マーク・ディ  
アメリカ合衆国・94025・カリフォルニア州・メンロパーク・ローレル ストリート・123  
5・ナンバー 4
- (72)発明者 チェン, ゾンジャン  
アメリカ合衆国・94303・カリフォルニア州・パロアルト・ローズウッド ドライブ・719
- (72)発明者 ワドハワン, ルチ  
アメリカ合衆国・94806・カリフォルニア州・サニベイイル・アシロマ テラス・977・ナ  
ンバー 2
- (72)発明者 ク, ウェイチュン  
アメリカ合衆国・95129・カリフォルニア州・サンノゼ・デボンシャー ドライブ・6387

審査官 坂東 博司

- (56)参考文献 特開平10-187593(JP, A)  
特開平10-334040(JP, A)  
特開昭63-137351(JP, A)  
特表2001-516925(JP, A)

- (58)調査した分野(Int.Cl., DB名)  
G06F 13/28