

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第3870717号  
(P3870717)

(45) 発行日 平成19年1月24日(2007.1.24)

(24) 登録日 平成18年10月27日(2006.10.27)

(51) Int. Cl. F I  
**G06F 13/28 (2006.01)** G O 6 F 13/28 3 1 0 J  
**G06F 13/38 (2006.01)** G O 6 F 13/38 3 1 0 D  
 G O 6 F 13/38 3 5 0

請求項の数 11 (全 31 頁)

(21) 出願番号	特願2001-143431 (P2001-143431)	(73) 特許権者	000002369
(22) 出願日	平成13年5月14日 (2001.5.14)		セイコーエプソン株式会社
(65) 公開番号	特開2002-342261 (P2002-342261A)		東京都新宿区西新宿2丁目4番1号
(43) 公開日	平成14年11月29日 (2002.11.29)	(74) 代理人	100090479
審査請求日	平成14年10月3日 (2002.10.3)		弁理士 井上 一
		(74) 代理人	100090387
			弁理士 布施 行夫
		(74) 代理人	100090398
			弁理士 大淵 美千栄
		(72) 発明者	神原 義幸
			長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内
		(72) 発明者	石田 卓也
			長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内

最終頁に続く

(54) 【発明の名称】 データ転送制御装置及び電子機器

(57) 【特許請求の範囲】

【請求項1】

バスを介したデータ転送のためのデータ転送制御装置であって、

第1の方向のデータ転送においては、第1のバスを介して受信され第2のバスを介して送信されるデータを記憶し、第2の方向のデータ転送においては、第2のバスを介して受信され第1のバスを介して送信されるデータを記憶するデータ記憶領域を有するバッファと、

前記バッファの前記データ記憶領域へのデータの書き込み及び前記データ記憶領域からのデータの読み出しの管理を行うバッファ管理回路とを含み、

前記第1の方向のデータ転送においては、第2のバスを介したデータ送信が終了したことを条件に、データ転送を行うデータフェーズが終了したと判断し、前記第2の方向のデータ転送においては、第2のバスを介したデータ受信が終了したと判断された後に、前記バッファの前記データ記憶領域が空になり最後のデータのデータの送受信に対するアクノリッジが前記第1のバスを介して返ってきた場合に、データ転送を行うデータフェーズが終了したと判断することを特徴とするデータ転送制御装置。

10

【請求項2】

請求項1において、

前記第1、第2のバスのうち第2のバス側にのみ、バスを介して送信又は受信されるデータのサイズをカウントするカウンタを設け、前記カウンタを用いて、前記第2のバスを介したデータ送信の終了と、前記第2のバスを介したデータ受信の終了を判断することを

20

特徴とするデータ転送制御装置。

【請求項 3】

請求項 1 又は 2 において、

前記第 2 のバスを介したデータ受信の終了後、データフェーズの終了までの期間において、前記第 2 のバスを介したデータ受信のステータスを取得し、取得したステータスに基づきステータスブロックを作成し、作成されたステータスブロックを、データフェーズの終了後のステータスフェーズにおいて、前記第 1 のバスを介して送信することを特徴とするデータ転送制御装置。

【請求項 4】

請求項 3 において、

1 つのエンドポイントを介して転送される情報として、前記第 2 の方向で転送されるデータと、ステータスブロックとが割り当てられている場合に、データ用のデータ記憶領域とステータス用のステータス記憶領域とが、前記バッファに確保され、

前記データフェーズでは前記データ記憶領域のデータについての処理を行いながら前記ステータス記憶領域のステータスブロックについての処理を行い、

データフェーズからステータスフェーズに切り替わった場合に、前記ステータス記憶領域からステータスブロックを読み出して、前記第 1 のバスを介して送信することを特徴とするデータ転送制御装置。

【請求項 5】

請求項 1 乃至 4 のいずれかにおいて、

前記第 2 のバスを介したデータ受信が終了し、且つ、前記バッファの前記データ記憶領域の残りデータサイズが最大パケットサイズ未満である場合には、前記データ記憶領域のショートパケットを前記第 1 のバスを介して自動送信することを特徴とするデータ転送制御装置。

【請求項 6】

請求項 5 において、

前記データ記憶領域のショートパケットの自動送信を有効又は無効に設定するショートパケット自動送信信号がアクティブである場合に、ショートパケットの自動送信を行うことを特徴とするデータ転送制御装置。

【請求項 7】

請求項 6 において、

前記第 2 のバスを介して受信されたデータに所与の情報が付加されて前記第 1 のバスを介して送信される場合には、前記ショートパケット自動送信信号が非アクティブに設定されることを特徴とするデータ転送制御装置。

【請求項 8】

請求項 1 乃至 7 のいずれかにおいて、

前記第 2 のバスを介したデータ受信が終了し、且つ、前記バッファの前記データ記憶領域の残りデータサイズが最大パケットサイズ未満である場合には、前記データ記憶領域にショートパケットが存在することを処理手段に知らせることを特徴とするデータ転送制御装置。

【請求項 9】

請求項 1 乃至 8 のいずれかにおいて、

前記データ記憶領域が、先に入力された情報が先に出力される記憶領域であることを特徴とするデータ転送制御装置。

【請求項 10】

請求項 1 乃至 9 のいずれかにおいて、

U S B ( Universal Serial Bus ) の規格に準拠したデータ転送を行うことを特徴とするデータ転送制御装置。

【請求項 11】

請求項 1 乃至 10 のいずれかのデータ転送制御装置と、

10

20

30

40

50

前記データ転送制御装置及びバスを介して転送されるデータの出力処理又は取り込み処理又は記憶処理を行う装置と、

を含むことを特徴とする電子機器。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、データ転送制御装置及び電子機器に関する。

【0002】

【背景技術及び発明が解決しようとする課題】

近年、パーソナルコンピュータと周辺機器（広義には電子機器）とを接続するためのインターフェース規格として、USB（Universal Serial Bus）が注目を集めている。このUSBには、従来は別々の規格のコネクタで接続されていたマウスやキーボードやプリンタなどの周辺機器を、同じ規格のコネクタで接続できると共にいわゆるプラグ&プレイやホットプラグも実現できるという利点がある。

10

【0003】

一方、このUSBには、同じくシリアルバスインターフェース規格として脚光を浴びているIEEE1394に比べて、転送速度が遅いという問題点がある。

【0004】

そこで、従来のUSB1.1の規格に対する互換性を持ちながら、USB1.1に比べて格段に高速な480Mbps（HSモード）のデータ転送速度を実現できるUSB2.0規格が策定され、注目を浴びている。

20

【0005】

さて、このUSB2.0では、HS（High Speed）モード時には480Mbpsでデータ転送が行われる。従って、高速な転送速度が要求されるハードディスクドライブや光ディスクドライブなどのストレージ機器のインターフェースとして用いることができるという利点がある。

【0006】

しかしながら、その一方で、USBバスに接続されるデータ転送制御装置は、480Mbpsという高速で転送されてくるデータを処理しなければならない。従って、データ転送制御装置の処理速度や、データ転送制御装置を制御するファームウェア（CPU）の処理速度が遅いと、実効的な転送速度を確保できず、バス帯域をロスしてしまうという課題がある。

30

【0007】

本発明は、以上のような技術的課題に鑑みてなされたものであり、その目的とするところは、実効的なバスの転送速度を向上できるデータ転送制御装置及び電子機器を提供することにある。

【0008】

また本発明の他の目的は、回路構成や回路制御を簡素化できるデータ転送制御装置及び電子機器を提供することにある。

【0009】

40

【課題を解決するための手段】

上記課題を解決するために本発明は、バスを介したデータ転送のためのデータ転送制御装置であって、第1の方向のデータ転送においては、第1のバスを介して受信され第2のバスを介して送信されるデータを記憶し、第2の方向のデータ転送においては、第2のバスを介して受信され第1のバスを介して送信されるデータを記憶するデータ記憶領域を有するバッファと、前記バッファの前記データ記憶領域へのデータの書き込み及び前記データ記憶領域からのデータの読み出しの管理を行うバッファ管理回路とを含み、前記第1の方向のデータ転送においては、第2のバスを介したデータ送信が終了したことを条件に、データ転送を行うデータフェーズが終了したと判断し、前記第2の方向のデータ転送においては、第2のバスを介したデータ受信が終了し、且つ、前記バッファの前記データ記憶領

50

域が空になったことを条件に、データ転送を行うデータフェーズが終了したと判断することを特徴とする。

【0010】

本発明によれば、第1の方向のデータ転送においては、第2のバスを介したデータ送信（例えばDMA転送）が終了したことを条件に、データフェーズ（データトランスポート。全データの転送）が終了したと判断される。一方、第2の方向のデータ転送においては、第2のバスを介したデータ受信が終了し、その後に、データ記憶領域が空になったことを条件に、データフェーズが終了したと判断される。

【0011】

このようにすれば、第2のバス側での転送データサイズ（データ長）をチェックするだけで、第1、第2の方向のデータ転送の両方において、データフェーズの終了を簡易に判断できるようになる。これにより、回路構成や回路制御を簡素化でき、データ転送制御装置を制御する処理手段の処理負荷を軽減できる。

10

【0012】

また本発明は、前記第2のバスを介して送信又は受信されるデータのサイズをカウントするカウンタを含むことを特徴とする。

【0013】

このようなカウンタを設けることで、第2のバス側での転送データサイズを容易にチェックできるようになる。しかも、第1のバス側にカウンタを設けなくても済むため、カウンタの制御を簡素化できる。

20

【0014】

また本発明は、前記第2のバスを介したデータ受信のステータスを取得し、取得したステータスを、データフェーズの終了後のステータスフェーズにおいて、前記第1のバスを介して送信することを特徴とする。

【0015】

このようにすれば、第2のバスを介したデータ受信の終了後、データフェーズの終了までの期間を有効利用して、ステータスに関する処理を行うことができるようになり、実効的な転送速度を向上できる。

【0016】

また本発明は、前記第2のバスを介したデータ受信が終了し、且つ、前記バッファの前記データ記憶領域の残りデータサイズが最大パケットサイズ未満である場合には、前記データ記憶領域のショートパケットを前記第1のバスを介して自動送信することを特徴とする。

30

【0017】

このようにすれば、処理手段が介在することなく、例えばハードウェアにより、データ記憶領域のショートパケットを自動的に送信できるようになる。これにより、処理手段の処理負荷を軽減でき、実効的な転送速度を向上できる。

【0018】

また本発明は、前記データ記憶領域のショートパケットの自動送信を有効又は無効に設定する手段を含むことを特徴とする。

40

【0019】

このようにすれば、ショートパケットの自動送信が許されるデバイスと、自動送信が許されないデバイスの両方に対応できるようになり、データ転送制御装置の汎用性を高めることができる。

【0020】

また本発明は、前記第2のバスを介して受信されたデータに所与の情報が付加されて前記第1のバスを介して送信される場合には、前記自動送信が無効に設定されることを特徴とする。

【0021】

このようにすれば、第1のバスに接続されるホスト等が、データフェーズの終了ではない

50

のに、データフェーズの終了であると誤認識する事態を、効果的に防止できる。

【0022】

また本発明は、前記第2のバスを介したデータ受信が終了し、且つ、前記バッファの前記データ記憶領域の残りデータサイズが最大パケットサイズ未満である場合には、前記データ記憶領域にショートパケットが存在することを処理手段に知らせることを特徴とする。

【0023】

このようにすれば、状況に応じた適切な処理を処理手段が行うことが可能になり、例えば、ショートパケットの自動送信が許されないようなデバイスにおいても、適切なデータ転送の制御を実現できる。

【0024】

また本発明は、前記データ記憶領域が、先に入力された情報が先に出力される記憶領域であることを特徴とする。

【0025】

このように、データ記憶領域を、先入れ先出しの記憶領域に設定することで、DMA (Direct Memory Access) 転送などにおいて効率的なデータ送受信を実現できる。

【0026】

また本発明は、USB (Universal Serial Bus)の規格に準拠したデータ転送を行うことを特徴とする。

【0027】

但し、本発明はUSB以外の規格 (USBの思想を受け継ぐ規格) に準拠したデータ転送に適用することも可能である。

【0028】

また本発明に係る電子機器は、上記のいずれかのデータ転送制御装置と、前記データ転送制御装置及びバスを介して転送されるデータの出力処理又は取り込み処理又は記憶処理を行う装置とを含むことを特徴とする。

【0029】

本発明によれば、データ転送制御装置のデータ転送を制御する処理手段 (ファームウェア等) の処理負荷を軽減できるため、電子機器の低コスト化、小規模化などを図れる。

【0030】

【発明の実施の形態】

以下、本実施形態について図面を用いて詳細に説明する。

【0031】

なお、以下に説明する本実施形態は、特許請求の範囲に記載された本発明の内容を何ら限定するものではない。また本実施形態で説明される構成の全てが本発明の解決手段として必須であるとは限らない。

【0032】

1. USB

1.1 データ転送手法

まず、USB (USB 2.0) のデータ転送手法について簡単に説明する。

【0033】

USBでは、IEEE 1394等とは異なり、データ転送の主導権をホストが有している。即ち、データ転送のトランザクションを起動するのはホスト側であり、データ転送に関する大部分の制御はホストが行う。このため、ホストの処理負荷は重くなるが、ホストとなるPC (パーソナルコンピュータ)等は高速で高性能なCPU (プロセッサ)を有しているため、このように負荷の重い処理もそれほど問題にはならない。

【0034】

一方、USBにおいては、デバイス (ターゲット) はホストからの要求に対して単に応えるだけでよい。そのため、デバイス側の処理、構成については簡素化できる。従って、デバイス側ではホストのような高性能で高速なCPUを使用する必要が無く、低価格なCPU (マイコン)を使用できるようになり、低コスト化を図れる。

10

20

30

40

50

## 【 0 0 3 5 】

さて、USBではこのようなホスト主導のデータ転送を実現するために、図1(A)に示すようなエンドポイント(E P 0 ~ 1 5 )がデバイス側に用意される。ここで、エンドポイントは、ホストとデバイスとの間でデータ転送を行うためのバッファ(F I F O )の入りに相当するものであり、USBでのデータ転送は、全て、このエンドポイントを経由して行われる。

## 【 0 0 3 6 】

そして、このエンドポイントは、デバイスアドレスとエンドポイント番号により一義的にアドレス指定できるようになっている。即ちホストは、デバイスアドレスとエンドポイント番号を指定することで、所望のエンドポイントへのデータ送信や、所望のエンドポイントからのデータ受信を自由に行うことができる。

10

## 【 0 0 3 7 】

また、エンドポイントの設定はデバイス側の任意であり、エンドポイント番号の割り当てや、各エンドポイントに割り当てられる記憶領域のデータサイズ等は、エニミュレーション(enumeration)処理時にホストは知ることができる。

## 【 0 0 3 8 】

なお、USBにおいては、データ転送のタイプとして、コントロール転送、アイソクロナス転送、インタラプト転送、バルク転送が用意されている。

## 【 0 0 3 9 】

ここで、コントロール転送は、ホストとデバイス(ターゲット)との間でコントロールエンドポイントを介して行われる制御用の転送モードである。このコントロール転送により、デバイスの初期化のためのコンフィグレーション情報等が転送される。

20

## 【 0 0 4 0 】

アイソクロナス転送は、画像データや音声データのように、データの正当性よりもバンド幅の確保が優先されるデータの転送のために用意された転送モードである。このアイソクロナス転送では、一定周期に一定量のデータを転送できることが保証されるため、データのリアルタイム性が重要なアプリケーションに有効な転送モードとなる。

## 【 0 0 4 1 】

インタラプト転送は、比較的低い転送速度で少量のデータを転送するために用意された転送モードである。

30

## 【 0 0 4 2 】

バルク転送は、不定期に発生する大量のデータを転送するために用意された転送モードである。このバルク転送では、アイソクロナス転送やインタラプト転送により使用された時間以外の空いた時間においてデータ転送が行われると共に、データの正当性がチェックされる。従って、リアルタイム性はあまり重要ではないが、データの信頼性は確保したいデータの転送に有効な転送モードである。

## 【 0 0 4 3 】

## 1 . 2 トランザクション構成

さて、図1(B)に示すように、USBのバルク転送におけるトランザクションは基本的に、トークンパケット、データパケット、ハンドシェイクパケットという3つのパケットにより構成される。なお、アイソクロナス転送の場合は、ハンドシェイクパケットは不要になる。

40

## 【 0 0 4 4 】

ここでトークンパケットは、ホストがデバイス(ターゲット)のエンドポイントのリードやライトを要求する場合等に使用されるパケットである。このトークンパケットは、例えば、P I D ( O U T , I N , S O F , S E T U P などのパケット I D )、A D D R ( デバイスアドレス)、E N D P ( エンドポイント番号)、C R C ( C y c l i c R e d u n d a n c y C h e c k ) のフィールドを有する。

## 【 0 0 4 5 】

また、データパケットは、データの実体を送るためのパケットであり、P I D ( D A T A

50

0、DATA 1)、DATA (データの実体)、CRCのフィールドを有する。

【0046】

また、ハンドシェイクパケットは、データ受信に成功したか否かを受信側が送信側に伝えるためのパケットであり、PID (ACK、NAK、STALL)のフィールドを有する。

【0047】

OUTトランザクション(ホストがデバイスに情報を出力するトランザクション)では、図1(C)に示すように、まず、ホストがデバイスに対してOUTトークンパケットを転送する。次に、ホストはデバイスに対してOUTのデータパケットを転送する。そして、デバイスは、OUTのデータパケットの受信に成功すると、ホストに対してACKのハンドシェイクパケットを転送する。

10

【0048】

一方、INTトランザクション(ホストがデバイスから情報を入力するトランザクション)では、図1(D)に示すように、まず、ホストがデバイスに対してINTトークンパケットを転送する。そして、INTトークンパケットを受信したデバイスは、ホストに対してINのデータパケットを転送する。そして、ホストは、INのデータパケットの受信に成功すると、デバイスに対してACKのハンドシェイクパケットを転送する。

【0049】

なお、図1(C)、(D)において、"DH"は、ホストからデバイスに対して情報が転送されることを意味し、"HD"は、デバイスからホストに対して情報が転送されることを意味する(以下の説明及び図でも同様)。

20

【0050】

1.3 Bulk-Only

さて、USBのデバイスは種々のクラスに分類される。そして、ハードディスクドライブや光ディスクドライブなどのデバイスは、マストレージと呼ばれるクラスに属し、このマストレージクラスには、電子機器のベンダ等により作成されたCBI(Control/Bulk/Interrupt)やBulk-Onlyなどの仕様がある。

【0051】

そして、CBI仕様では図2(A)に示すように、デバイスは、コントロール、バルクアウト、バルクイン、インタラプトのエンドポイントEP0、1、2、3を用意する。ここで、エンドポイントEP0では、USB層のコントロールの packets や、コマンドの packets が転送される。また、EP1ではOUTデータ(ホストからデバイスに転送されるデータ)、EP2ではINデータ(デバイスからホストに転送されるデータ)、EP3ではインタラプトINの packets が転送される。なお、EP1~15のいずれを、バルクアウト、バルクイン、インタラプトINのエンドポイントに割り当てるかは、デバイス側の任意である。

30

【0052】

一方、Bulk-Only仕様では図2(B)に示すように、デバイスは、コントロール、バルクアウト、バルクインのエンドポイントEP0、1、2を用意する。ここで、エンドポイントEP0では、USB層のコントロールの packets が転送される。また、EP1ではコマンド(CBW)及びOUTデータ、EP2ではステータス(CSW)及びINデータの packets が転送される。なお、EP1~15のいずれを、バルクアウト、バルクインのエンドポイントに設定するかは、デバイス側の任意である。

40

【0053】

ここで、CBW(Command Block Wrapper)は、コマンドブロック及びこれに関連する情報を含む packets であり、図3にそのフォーマットが示される。また、CSW(Command Status Wrapper)は、コマンドブロックのステータスを含む packets であり、図4にそのフォーマットが示される。

【0054】

図3において、dCBWSignatureは、当該 packets をCBWとして識別するための情報であ

50

り、dCBWTagは、コマンドブロックのタグであり、dCBWDataTransferLengthはデータフェーズにおいて転送されるデータの長さを指定する。また、bmCBWFlagsは転送方向等を指定するためのフラグであり、dCBWLUNはロジカルユニット番号であり、bCBWCBLengthはコマンド長であり、CBWCBは、ATA / ATAPI や SCSI などのコマンドがカプセル化されて記述されるコマンドブロックである。

**【 0 0 5 5 】**

また図 4 において、dCSWSignatureは、当該パケットをCSWとして識別するための情報である。また、dCSWTagは、ステータスブロックのタグであり、当該CSWに対応するCBWのdCBWTagの値が書き込まれる。また、CSWDataResidueは、CBWのdCBWDataTransferLengthにより指定されたデータの長さとしてデバイスが実際に処理したデータの長さの差であり、bCSWStatusはステータスブロックである。

10

**【 0 0 5 6 】**

次に、図 2 ( B ) の Bulk - Only 仕様におけるデータの書き込み処理、読み出し処理について、図 5 ( A )、( B ) を用いて説明する。

**【 0 0 5 7 】**

ホストがデバイスにデータを書き込む場合には、図 5 ( A ) に示すように、まず、ホストがCBWをデバイスに転送するコマンドフェーズ(コマンドトランスポート)が行われる。具体的には、ホストが、エンドポイントEP1を指定するトークンパケットをデバイスに転送し、次に、CBW(図 2 ( B ) の A 1、図 3 参照)をデバイスのエンドポイントEP1に転送する。このCBWにはライトコマンドが含まれる。そして、デバイスからホストにACKのハンドシェイク(H.S)が返却されると、コマンドフェーズが終了する。

20

**【 0 0 5 8 】**

コマンドフェーズ(コマンドトランスポート)が終了するとデータフェーズ(データトランスポート)に移行する。このデータフェーズでは、まず、ホストが、エンドポイントEP1を指定するトークンパケットをデバイスに転送し、次に、OUTデータ(図 2 ( B ) の A 2 参照)をデバイスのエンドポイントEP1に転送する。そして、デバイスからホストにACKのハンドシェイクが返却されると、1つのトランザクションが終了する。そして、このようなトランザクションが繰り返され、CBWのdCBWDataTransferLength(図 3 参照)で指定されるデータ長の分だけデータが転送されると、データフェーズが終了する。

30

**【 0 0 5 9 】**

データフェーズ(データトランスポート)が終了するとステータスフェーズ(ステータストランスポート)に移行する。このステータスフェーズでは、まず、ホストが、エンドポイントEP2を指定するトークンパケットをデバイスに転送する。すると、デバイスが、エンドポイントEP2にあるCSW(図 2 ( B ) の A 3、図 4 参照)をホストに転送する。そして、ホストからデバイスにACKのハンドシェイクが返却されると、ステータスフェーズが終了する。

**【 0 0 6 0 】**

ホストがデータを読み出す場合は、図 5 ( B ) に示すように、まず、ホストが、エンドポイントEP1を指定するトークンパケットをデバイスに転送し、次に、CBWをデバイスのエンドポイントEP1に転送する。このCBWはリードコマンドを含む。そして、デバイスからホストにACKのハンドシェイクが返却されると、コマンドフェーズが終了する。

40

**【 0 0 6 1 】**

コマンドフェーズが終了するとデータフェーズに移行する。このデータフェーズでは、まず、ホストが、エンドポイントEP2を指定するトークンパケットをデバイスに転送する。すると、デバイスが、エンドポイントEP2にあるINデータ(図 2 ( B ) の A 4 参照)をホストに転送し、ホストからデバイスにACKのハンドシェイクが返却されると、1つのトランザクションが終了する。そして、このようなトランザクションが繰り返され、CBWのdCBWDataTransferLengthで指定されるデータ長の分だけデータが転送されると、

50



データフェーズが終了する。

【0062】

データフェーズが終了するとステータスフェーズに移行する。このステータスフェーズの処理は図5(A)のデータの書き込み処理の場合と同様である。

【0063】

2. 本実施形態の特徴

2.1 構成例

図6に本実施形態のデータ転送制御装置の構成例を示す。

【0064】

本実施形態のデータ転送制御装置は、トランシーバマクロ20、SIE30、エンドポイント管理回路40、バッファ管理回路50、バッファ60、バルク転送管理回路70、DMA80を含む。なお、本発明のデータ転送制御装置は、図6に示す回路ブロックの全てを含む必要はなく、それらの一部を省略する構成としてもよい。

10

【0065】

ここで、トランシーバマクロ20は、USB(第1のバス)のFSモードやHSモードでのデータ転送を実現するための回路である。このトランシーバマクロ20としては、例えばUSB2.0の物理層回路や、論理層回路の一部についてのインターフェースを定義したUTMI(USB2.0 Transceiver Macrocell Interface)に準拠したマクロセルを用いることができる。このトランシーバマクロ20は、トランシーバ回路22、クロック生成回路24を含む。

20

【0066】

トランシーバ回路22は、差動信号DP、DMを用いてUSB(第1のバス)上のデータを送受信するアナログフロントエンド回路(受信回路、送信回路)を含む。また、ビットスタッフィング、ビットアンスタッフィング、シリアル・パラレル変換、パラレル・シリアル変換、NRZIデコード、NRZIエンコード、サンプリングクロック生成などの処理を行う回路を含む。

【0067】

またクロック生成回路24は、データ転送制御装置が使用する動作クロックや、サンプリングクロックの生成に使用されるクロックなどを生成する回路であり、480MHzや60MHzのクロックを生成するPLLや発振回路などを含む。

30

【0068】

SIE(Serial Interface Engine)は、USBのパケット転送処理などの種々の処理を行う回路であり、パケットハンドラ回路32、サスペンド&レジューム制御回路34、トランザクション管理回路36を含む。

【0069】

パケットハンドラ回路32は、ヘッダ及びデータからなるパケットの組み立て(生成)や分解などを行う回路であり、CRCの生成や解読を行うCRC処理回路33を含む。

【0070】

サスペンド&レジューム制御回路34は、サスペンドやレジューム時のシーケンス制御を行う回路である。

40

【0071】

トランザクション管理回路36は、トークン、データ、ハンドシェークなどのパケットにより構成されるトランザクションを管理する回路である。具体的には、トークンパケットを受信した場合には、自分宛か否かを確認し、自分宛の場合には、ホストとの間でデータパケットの転送処理を行い、その後、ハンドシェークパケットの転送処理を行う。

【0072】

エンドポイント管理回路40は、バッファ60の各記憶領域の入り口となるエンドポイントを管理する回路であり、エンドポイントの属性情報を記憶するレジスタ(レジスタセット)などを含む。

【0073】

50

バッファ管理回路50は、例えばRAMなどで構成されるバッファ60を管理する回路である。より具体的には、書き込みアドレスや読み出しアドレスを生成し、バッファ60へのデータの書き込み処理やバッファ60からのデータの読み出し処理を行う。

【0074】

バッファ60(パケット記憶手段)は、USBを介して転送されるデータ(パケット)を一時的に記憶するものであり、USB(第1のバス)でのデータ転送速度と、EBS(第2のバス。DMAバス)でのデータ転送速度との速度差を補償する機能などを有する。なお、EBSは、ハードディスクドライブやCDドライブやスキャナなどのデバイスに接続される外部バスである。

【0075】

そして本実施形態では、第1の方向のデータ転送(OUTトランザクション)においては、USB(第1のバス)を介して受信されEBS(第2のバス)を介して送信されるデータ(OUTデータ)を記憶し、第2の方向のデータ転送(INトランザクション)においては、EBSを介して受信されUSBを介して送信されるデータ(INデータ)を記憶するデータ記憶領域62(FIFO。FIFO設定の記憶領域)を、バッファ60が有する。

【0076】

そして本実施形態のデータ転送制御装置は、第1の方向のデータ転送においては、EBSを介したデータ送信が終了したことを条件に、データフェーズ(データトランスポート。全データの転送)が終了したと判断する。一方、第2の方向のデータ転送においては、EBSを介したデータ受信が終了し、バッファ60のデータ記憶領域62が空になったこと(最後のデータの packets についてのアクリッジがUSBを介して返ってきたこと)を条件に、データフェーズが終了したと判断する。

【0077】

また本実施形態のデータ転送制御装置は、EBSを介したデータ受信が終了し、バッファ60のデータ記憶領域62(FIFO。FIFO設定の記憶領域)の残りデータサイズ(データ長、転送数)が最大パケットサイズ未満である場合には、データ記憶領域62のショートパケット(最大パケットサイズ未満のサイズの packets)をUSBを介して自動送信したり、ショートパケットの存在を、割り込み信号等を用いてCPU(ファームウェア、処理手段)に知らせるようにしている。

【0078】

更に本実施形態では、1つのエンドポイント(図2(B)のEP1)を介して転送される情報として、コマンドブロック(第1の情報)とデータ(第2の情報。OUTデータ)とが割り当てられている場合に(図2(B)のA1、A2参照)、コマンドブロック用のコマンド記憶領域(第1の記憶領域)とデータ用のデータ記憶領域(第2の記憶領域)とが、バッファ60上に用意(確保)される。

【0079】

そしてバッファ管理回路60は、USB(第1のバス)を介してコマンドブロックが転送されるコマンドフェーズ(第1のフェーズ)では、ホストからエンドポイント(EP1)に向けて転送される情報を、コマンドブロック用のコマンド記憶領域に書き込み、USBを介してデータ(OUTデータ)が転送されるデータフェーズ(第2のフェーズ)では、ホストからエンドポイントに向けて転送される情報を、データ用のデータ記憶領域に書き込む。即ち、フェーズが切り替わったことを条件に、情報(コマンドブロック、データ)の書き込み領域を切り替えている。

【0080】

或いは本実施形態では、1つのエンドポイント(図2(B)のEP2)を介して転送される情報として、データ(第3の情報。INデータ)とステータスブロック(第4の情報)が割り当てられている場合に(図2(B)のA3、A4参照)、データ用のデータ記憶領域(第3の記憶領域)とステータスブロック用のステータス記憶領域(第4の記憶領域)とが、バッファ60上に用意される。

10

20

30

40

50

## 【 0 0 8 1 】

そしてバッファ管理回路 6 0 は、U S B を介してデータ ( I N データ ) が転送されるデータフェーズ ( 第 3 のフェーズ ) では、エンドポイント ( E P 2 ) からホストに向けて転送される情報を、データ用のデータ記憶領域から読み出し、U S B を介してステータスブロックが転送されるステータスフェーズ ( 第 4 のフェーズ ) では、エンドポイントからホストに向けて転送される情報を、ステータス用のステータス記憶領域から読み出す。即ち、フェーズが切り替わったことを条件に、情報 ( データ、ステータスブロック ) の読み出し領域を切り替えている。

## 【 0 0 8 2 】

このようにすることで、1つのエンドポイントに複数種類の情報 ( コマンドブロック、O U T データ、I N データ、ステータスブロック ) が割り当てられている場合にも、コマンド記憶領域のコマンドブロックについての処理を行いながら、データ記憶領域の O U T データについての処理を行うことが可能になる。或いは、データ記憶領域の I N データについての処理を行いながら、ステータス記憶領域のステータスブロックについての処理を行うことが可能になる。従って、データ転送制御装置の処理速度の向上や、ファームウェア ( 処理手段 ) の負荷を軽減することが可能になり、実効的なバスの転送速度を向上できる。

10

## 【 0 0 8 3 】

バルク転送管理回路 7 0 は、U S B におけるバルク転送を管理するための回路である。

## 【 0 0 8 4 】

D M A C 8 0 は、E B U S を介して D M A 転送を行うための D M A コントローラであり、D M A カウンタ 8 2 を含む。そして D M A カウンタ 8 2 は、E B U S ( 第 2 のバス ) を介して送信又は受信されるデータのサイズ ( 転送数、データ長 ) をカウントする回路である。

20

## 【 0 0 8 5 】

## 2 . 2 詳細な接続例

図 7 に、S I E 3 0、エンドポイント管理回路 4 0、バッファ管理回路 5 0、バッファ 6 0、バルク転送管理回路 7 0、D M A C 8 0 の詳細な接続例を示す。

## 【 0 0 8 6 】

図 7 において、S I E R e a d D a t a は、バッファ 6 0 ( データ記憶領域 6 2 ) から読み出され、U S B ( 第 1 のバス ) を介してホストに転送されるデータ ( I N データ ) である。S I E W r i t e D a t a は、U S B を介してホストから転送され、バッファ 6 0 に書き込まれるデータ ( O U T データ ) である。D M A R e a d D a t a は、バッファ 6 0 から読み出され、E B U S ( 第 2 のバス ) を介してハードディスクドライブ H D D などのデバイスに転送 ( D M A 転送 ) されるデータである。D M A W r i t e D a t a は、E B U S を介して H D D 等から転送され、バッファ 6 0 に書き込まれるデータである。

30

## 【 0 0 8 7 】

また、S I E 3 0 が出力する S I E R e a d R e q は、S I E R e a d D a t a の読み出しを要求する信号であり、エンドポイント管理回路 4 0 が出力する S I E R e a d A c k は、S I E R e a d R e q に対するアクリッジ信号である。同様に、S I E W r i t e R e q は、S I E W r i t e D a t a の書き込みを要求する信号であり、S I E W r i t e A c k は、S I E W r i t e R e q に対するアクリッジ信号である。

40

## 【 0 0 8 8 】

また、D M A C 8 0 が出力する D M A R e a d R e q は、D M A R e a d D a t a の読み出しを要求する信号であり、エンドポイント管理回路 4 0 が出力する D M A R e a d A c k は、D M A R e a d R e q に対するアクリッジ信号である。同様に、D M A W r i t e R e q は、D M A W r i t e D a t a の書き込みを要求する信号であり、D M A W r i t e A c k は、D M A W r i t e R e q に対するアクリッジ信号である。

## 【 0 0 8 9 】

F u l l は、バッファ 6 0 のデータ記憶領域 6 2 がフルか否かを示す信号であり、E m p

50

ty は、データ記憶領域 6 2 が空 ( エンプティ ) か否かを示す信号である。この場合に、データ記憶領域 6 2 がフルか否か、或いは空か否かは、データ記憶領域 6 2 の残りデータサイズ ( 残りデータエリア ) をカウントする残りデータサイズカウンタ 4 1 からのカウント値等に基づいて判断される。

【 0 0 9 0 】

なお、本実施形態のデータ記憶領域 6 2 は、先に入力された情報が先に出力される領域 ( F I F O 。 F I F O 設定の領域 ) になっている。この場合、データ記憶領域 6 2 を、直列接続のレジスタ、メモリ等で構成することで、先入れ先出しの領域に設定してもよいし、R A M のアドレス制御を工夫することで、先入れ先出しの領域に設定してもよい。

【 0 0 9 1 】

エンドポイント管理回路 4 0 が出力する H a v e M P S は、バッファ 6 0 のデータ記憶領域 6 2 の残りデータサイズ ( データ長 ) が、最大パケットサイズか否かを示す信号である。最大パケットサイズが 5 1 2 バイトの場合を例にとれば、残りデータサイズが 5 1 2 バイト ( 以上 ) の時には H a v e M P S がアクティブ ( H レベル ) になり、残りデータサイズが 0 ~ 5 1 1 バイトの時には H a v e M P S が非アクティブ ( L レベル ) になる。この場合の残りデータサイズは、残りデータサイズカウンタ 4 1 により求められる。

【 0 0 9 2 】

H D D A T A は、E B U S ( 第 2 のバス ) を介して D M A 転送されるデータであり、x D M A C K は、この D M A 転送のアクノリッジ信号であり、x D M A R Q は、D M A 転送の要求信号である。また、x H I O R は、読み出しパルス信号であり、x H I O W は、書き込みパルス信号である。

【 0 0 9 3 】

D M A \_ \_ S t a r t は、D M A 転送の開始を C P U ( ファームウェア ) が指示するための信号である。D M A \_ \_ R u n n i n g は、E B U S を介した D M A 転送が行われていることを示す信号であり、E B U S による D M A 転送の開始によりアクティブ ( H レベル ) になり、D M A 転送の終了により非アクティブ ( L レベル ) になる。即ち、E B U S でのデータ転送 ( データフェーズ ) 期間において、D M A \_ \_ R u n n i n g はアクティブになる。

【 0 0 9 4 】

A u t o E n S h o r t は、バッファ 6 0 のデータ記憶領域 6 2 に存在するショートパケットの自動送信を有効又は無効に設定するための信号である。また、E n S h o r t P k t は、ショートパケットの転送を S I E 3 0 に指示するための信号である。

【 0 0 9 5 】

例えば、A u t o E n S h o r t が H レベルに設定されている場合には、E B U S を介したデータ受信が終了し、且つ、データ記憶領域 6 2 の残りデータサイズが最大パケットサイズ未満になると、E n S h o r t P k t がアクティブになる。これにより、データ記憶領域 6 2 に存在するショートパケット ( 最大パケットサイズ未満のパケット ) が、S I E 3 0 により U B S を介して自動送信される。

【 0 0 9 6 】

より具体的には、D M A \_ \_ R u n n i n g が非アクティブ ( データ受信終了 ) になった後に、H a v e M P S が非アクティブ ( 最大パケットサイズ未満 ) になると、E n S h o r t P k t がアクティブになり、データ記憶領域 6 2 のショートパケットが S I E 3 0 により自動送信される。

【 0 0 9 7 】

I N T r a n A C K は、I N トランザクションに対するアクノリッジが、ホストから適正に返された場合にアクティブになる信号である。B u l k D M A C m p は、バルク転送におけるデータフェーズ ( 全データの転送 ) が終了したことを示すための信号である。

【 0 0 9 8 】

例えば、E B U S を介したデータ受信が終了した後、データ記憶領域 6 2 が空になり、最後のデータ ( I N データ ) のパケットの送信に対するアクノリッジがホストから返ってく

10

20

30

40

50

ると、INT r a n A C Kがアクティブになる。すると、B u l k D M A C m pがアクティブになり、データフェーズ（全データの転送）が終了したことが、ファームウェア（CPU）に伝えられる。これにより、ファームウェアは、次のステータスフェーズの処理に移行できる。

【0099】

### 2.3 動作

次に本実施形態の動作について説明する。

【0100】

図8（A）は、O U Tトランザクション（ホストからデバイスにデータが転送される第1の方向のデータ転送）において、バッファ60のデータ記憶領域62（F I F O）の残りエリア（空きエリア）のサイズが、ホストから転送されてくるデータのサイズ（データ長）以上である場合のタイミング波形図である。この場合には、ホストからのデータの全てをデータ記憶領域62に書き込むことができるため、図8（A）のC1に示すようにホストに対してA C Kを返すことができる。

10

【0101】

なお、H Sモード時において、データ記憶領域62が、ダブルバッファ構造ではなく、シングルバッファ構造である場合には、データ記憶領域62に空きが生じるまでは、ホストからの次のパケットを受信できない。従って、この場合には、A C Kの代わりにN Y E Tがホストに返される。また、ダブルバッファ構造であっても、両方のバッファが使用中であり、次のデータを受け取ることができない場合には、同様に、A C Kの代わりにN Y E T

20

【0102】

図8（B）は、O U Tトランザクションにおいて、データ記憶領域62の残りエリアサイズがデータサイズよりも小さい場合のタイミング波形図である。

【0103】

この場合には、ホストからのデータの全てをデータ記憶領域62に書き込むことができないため、図8（B）のC2に示すようにホストに対してN A Kを返すことになる。そして、今回のO U Tトークンに対するデータ転送により受け取ったデータを破棄するために、C3に示すようにR e w i n d信号がアクティブになる。

【0104】

なお、図8（A）、（B）のC4、C5において、S I E 3 0がS I E W r i t e R e qをアクティブにすると、エンドポイント管理回路40の指示により、バッファ管理回路50がS I E W r i t e D a t aの書き込みアドレスを生成して、バッファ60に出力する。これにより、例えば4バイトのS I E W r i t e D a t aがバッファ60のデータ記憶領域62に書き込まれ、S I E 3 0に対してS I E W r i t e A c kが返される。そして、このような4バイトずつの書き込み処理を繰り返すことで、ホストからのデータ（例えば512バイトのデータ）がデータ記憶領域62に順次書き込まれる。

30

【0105】

図8（C）、（D）は、データ記憶領域62に空きがあるか否かをホストがデバイスに問い合わせるP I N Gトランザクション（P I N Gプロトコル）についてのタイミング波形図である。このP I N GトランザクションはU S BのH Sモードにおいてサポートされているものである。このプロトコルは、ホストからのO U TトークンやO U Tのデータに対して、デバイスからN Y E Tを受け取った場合に使用される。

40

【0106】

そして、このP I N Gトランザクションにおいてはデータ記憶領域62の残りエリアサイズ（空きサイズ）がデータサイズ以上である場合には、図8（C）のC6に示すようにホストに対してA C Kが返される。一方、データ記憶領域62の残りエリアサイズがデータサイズよりも小さい場合には、図8（D）のC7に示すようにホストに対してN A Kが返されることになる。

【0107】

50

図9(A)は、INトランザクション(デバイスからホストにデータが転送される第2の方向のデータ転送)において、データ記憶領域62の残りデータ(有効データ)のサイズが、最大パケットサイズ以上である場合のタイミング波形図である。この場合には、最大パケットサイズ分のデータがホストに転送されると、図9(A)のD1に示すようにホストからデバイスにACKが返される。

【0108】

なお、図9(A)のD2において、SIE30がSIEReadReqをアクティブにすると、エンドポイント管理回路40の指示により、バッファ管理回路50がSIEReadDataの読み出しアドレスを生成して、バッファ60に出力する。これにより、例えば4バイトのSIEReadDataがデータ記憶領域62から読み出され、SIE30に対してSIEReadAckが返される。そして、このような4バイトずつの読み出し処理を繰り返すことで、ホストに対して転送すべきデータがデータ記憶領域62から順次読み出される。

10

【0109】

図9(B)は、INトランザクションにおいて、データ記憶領域62の残りデータのサイズが最大パケットサイズよりも小さく、且つ、EnShortPkt(図7参照)がLレベル(非アクティブ)である場合のタイミング波形図である。この場合には、図9(B)のD3、D4、D5に示すようにEnShortPktがLレベルであるため、ショートパケット(最大パケットサイズ未満のパケット)の転送が禁止される。従って、ホストからのINTトークンに対して、D6、D7、D8に示すようにNAKが返されることになる。

20

【0110】

図9(C)は、INトランザクションにおいて、データ記憶領域62の残りデータのサイズが最大パケットサイズよりも小さく、且つ、EnShortPktがHレベル(アクティブ)になった場合のタイミング波形図である。この場合には、図9(C)のD9に示すように、EnShortPktがHレベルになることでショートパケットの転送が許可され、D10に示すように、データ記憶領域62に存在するショートパケットがホストに適正に転送されることになる。

【0111】

なお、バッファ60のデータ記憶領域62は、OUTトランザクションとINトランザクションとで共用の領域にしてもよいし、別領域にしてもよい。

30

【0112】

図10は、ハードディスクドライブHDDなどのデバイスへのデータ送信時におけるDMA転送のタイミング波形図である。

【0113】

図10のE1、E2に示すように、ハードディスクドライブHDDからのHDMARQ(図7参照)、CPUからのDMA\_\_Startがアクティブになると、DMA転送が開始し、E3に示すように、DMA\_\_Runningがアクティブになる。

【0114】

そして、E4、E5に示すように、DMAReadReq、DMAReadAckを用いたハンドシェイクが行われ、バッファ60のデータ記憶領域62から例えば4バイト(32ビット)のデータが読み出される。読み出された4バイトのデータは、E6、E7に示すように、書き込みパルス信号xHIOWを用いて、16ビットのバスEBUSを介して例えば2バイトずつHDDに送信される。

40

【0115】

そして、データが送信される毎に、E8に示すように、DMAカウンタ82のカウント値Count(データ転送数)がデクリメントされる。なお、Countの初期値Nは、コマンドフェーズにおいて、図3のCBWのdCBWTransferLengthに基づき設定される。

【0116】

E9に示すように、Countが0になり、EBUSにおける全データの転送が終了する

50

と、E10に示すように、DMA\_\_Runningが非アクティブになる。

【0117】

図11は、ハードディスクドライブHDDなどのデバイスからのデータ受信時におけるDMA転送のタイミング波形図である。

【0118】

図11のF1、F2に示すようにHDMAReq、DMA\_\_Startがアクティブになると、DMA転送が開始し、F3に示すように、DMA\_\_Runningがアクティブになる。

【0119】

そして、F4、F5に示すように、読み出しパルス信号xHIORを用いてEBUSを介しHDDからデータが受信される。受信されたデータは、F6、F7に示すように、DMAWriteReq、DMAWriteAckを用いたハンドシェイクにより、バッファ60のデータ記憶領域62に書き込まれる。

【0120】

F8に示すように、データが受信される毎に、Count(データ転送数)がデクリメントされる。そして、F9に示すように、Countが0になり、EBUSにおける全データの転送が終了すると、F10に示すように、DMA\_\_Runningが非アクティブになる。

【0121】

2.4 データフェーズ(データトランスポート)の終了の検出

さて、USBにおいては、図5(A)、(B)で説明したように、ホスト側とデバイス側とで、常にフェーズを整合させながらデータ転送を行う必要がある。

【0122】

例えば図5(B)において、ホストは、現在のフェーズがデータフェーズであると認識しているのに対して、デバイスは、現在のフェーズがステータスフェーズであると認識していたとする。すると、ホストから転送されたINTトークン(図5(B)のB4のトークン)を、デバイスはCSW(B6)に対するINTトークンであると誤認識してしまい、データ転送にエラーが生じてしまう。

【0123】

また、図2(B)に示すBulk-Only仕様では、1つのエンドポイントを介して転送される情報として、複数種類の情報が割り当てられている。具体的には、図2(B)では、バルクアウトエンドポイントEP1を介して転送される情報として、CBW(コマンド)及びOUTデータが割り当てられ、バルクインエンドポイントEP2を介して転送される情報として、CSW(ステータス)及びINデータが割り当てられている。従って、ホスト及びデバイスは、各エンドポイントを介して転送される情報がどの情報なのかを判別する必要があり、Bulk-Only仕様では、現在のフェーズがどのフェーズなのかをホスト及びデバイスが判断して、この情報の判別を行っている。

【0124】

例えば、図5(A)、(B)のB1、B2では、現在のフェーズがコマンドフェーズであるため、エンドポイントEP1を介して転送される情報はCBWであると判断される。また、B3、B4では、現在のフェーズがデータフェーズであるため、エンドポイントEP1を介して転送される情報はOUTデータであると判断され、エンドポイントEP2を介して転送される情報はINデータであると判断される。また、B5、B6では、現在のフェーズがステータスフェーズであるため、エンドポイントEP2を介して転送される情報はCSWであると判断される。

【0125】

このようにUSBにおいては、現在のフェーズがどのフェーズなのかをホスト及びデバイスは正確に認識する必要がある。従って、例えば、データフェーズからステータスフェーズへの切り替わりのタイミングについても、正確に判断できることが望まれる。

【0126】

10

20

30

40

50

そこで、本実施形態では、データフェーズからステータスフェーズへの切り替わりの判断、即ちデータフェーズの終了の判断を、以下に説明するような手法で実現している。

【0127】

例えばホストからデバイスにOUTデータが転送されるOUTトランザクション（第1の方向のデータ転送）の場合には、図12のようにしてデータフェーズの終了（全データ転送の終了。フェーズの切り替わり）を判断している。

【0128】

具体的には、まず、図12のG1に示すようにUSB（第1のバス）におけるデータ転送（データフェーズ）がスタートする。そして、転送されたデータがバッファ60のデータ記憶領域62に書き込まれると、G2に示すようにデータ記憶領域62が空（エンプティ）か否かを示す信号Emptyが非アクティブになり、データ記憶領域62が空でないことが示される。

10

【0129】

次に、G3、G4に示すように、CPUがDMA\_Startをアクティブに設定することで、DMA\_Runningがアクティブになり、EBUS（第2のバス）におけるデータ転送（EBUSのデータフェーズ）がスタートする。

【0130】

そして、G5に示すようにUSBにおけるデータ転送が終了し、G6に示すようにEBUSにおけるデータ転送（DMA転送）が終了すると、データフェーズが終了して、データフェーズからステータスフェーズに切り替わったと判断される。

20

【0131】

このG6のタイミングは、USBでのデータ転送が終了するG5のタイミングよりも遅いタイミングとなる。また、このG6のタイミングは、バッファ60のデータ記憶領域62が空になり、Empty信号がアクティブになるG7のタイミングと一致する。そして本実施形態では、このG6のタイミングを、EBUS側に設けられた図12のDMAカウンタ82を用いて、EBUSで転送されるデータサイズ（転送数。データ長）をカウントすることで判断している（図10のE9参照）。

【0132】

このように本実施形態では、OUTトランザクション（第1の方向のデータ転送）においては、EBUS（第2のバス）を介したデータ送信が終了したことを条件に、データフェーズが終了したと判断している。

30

【0133】

一方、デバイスからホストにINデータが転送されるINトランザクション（第2の方向のデータ転送）の場合には、図13のようにしてデータフェーズの終了（全データ転送の終了。フェーズの切り替わり）を判断している。

【0134】

具体的には、まず、H1、H2に示すように、CPUがDMA\_Startをアクティブに設定することで、DMA\_Runningがアクティブになり、EBUSにおけるDMA転送がスタートする。そして、EBUSを介して転送されたデータがバッファ60のデータ記憶領域62に書き込まれると、H3に示すように、Emptyが非アクティブになり、データ記憶領域62が空でないことが示される。

40

【0135】

次に、H4に示すようにUSBにおけるデータ転送がスタートする。そして、USBを介して転送されるデータがデータ記憶領域62から読み出され、もし、データ記憶領域62が空になると、Emptyがアクティブになる。

【0136】

そして本実施形態では、H6に示すように、EBUS（第2のバス）を介したデータ転送が終了したと判断された後、H7に示すように、データ記憶領域62が空になり、Empty信号がアクティブになったタイミングで、データフェーズ（全データの転送）が終了したと判断する。この場合、H6のタイミングは、EBUS側に設けられたDMAカウン

50



タ 8 2 を用いて、E B U S で転送されるデータサイズをカウントすることで判断できる。また、H 7 のタイミングは、U S B ( 第 1 のバス ) でのデータ転送 ( データフェーズ ) が終了する H 8 のタイミングと一致する。

【 0 1 3 7 】

このように本実施形態では、I N トランザクション ( 第 2 の方向のデータ転送 ) においては、E B U S ( 第 2 のバス ) を介したデータ受信が終了し、且つ、バッファ 6 0 のデータ記憶領域が空になったことを条件に、データフェーズが終了したと判断している。

【 0 1 3 8 】

例えば、図 1 2、図 1 3 の本実施形態の手法とは異なる手法として、E B U S 側と U S B 側の両方に、データサイズをカウントするカウンタを設ける手法を考えることができる。この手法では、例えば、O U T トランザクションにおいては、E B U S 側のカウンタのカウント値が 0 になったことを条件に、データフェーズが終了したと判断し、I N トランザクションにおいては、U S B 側のカウンタのカウント値が 0 になったことを条件に、データフェーズが終了したと判断する。

【 0 1 3 9 】

しかしながら、この手法では、2 つのカウンタを設ける必要が生じ、これらの 2 つのカウンタを制御する処理が煩雑化してしまう。この結果、回路制御や回路構成が複雑化してしまい、データ転送制御装置の大規模化、高コスト化を招く。また、データ転送制御装置を制御するファームウェア ( C P U ) の処理も複雑化し、その処理のオーバーヘッドのために、実効的なデータ転送速度が低下するおそれもある。

【 0 1 4 0 】

これに対して、図 1 2、図 1 3 の本実施形態の手法によれば、図 7 に示すように E B U S ( 第 2 のバス ) 側にだけカウンタを設ければよく、U S B ( 第 1 のバス ) 側にカウンタを設ける必要がなくなる。従って、回路制御を容易化できると共に回路構成を簡素化でき、データ転送制御装置のコンパクト化、低コスト化を図れる。また、データ転送制御装置を制御するファームウェアの処理も簡素化でき、実効的なデータ転送速度を向上できる。

【 0 1 4 1 】

なお、本実施形態では、図 1 3 の H 9 に示すように、E B U S ( 第 2 のバス ) を介したデータ受信が終了すると、このデータ受信のステータスをハードディスクドライブ H D D 側から取得し、この取得されたステータスに基づいて、図 4 に示す C S W ( ステータス ) を作成する。この C S W の作成処理は、例えばファームウェア等により行われる。

【 0 1 4 2 】

そして、図 1 3 の H 1 0 に示すように、この作成された C S W を、データフェーズの終了後のステータスフェーズにおいて、U S B ( 第 1 のバス ) を介して、ホスト側に送信する。

【 0 1 4 3 】

このようにすれば、図 1 3 の H 9 に示すように、U S B においてデータ転送を行っている期間を有効利用して、H D D 側からステータスを取得し、C S W を作成できるようになる。そして、H 1 0 に示すように、データフェーズからステータスフェーズに移行した後に、直ぐに、ホストからの I N トークンに対して C S W を返すことが可能になる。従って、ステータスフェーズにおいて、ホストからの I N トークンに対して N A K を返す頻度を少なくでき、実効的なデータ転送速度を向上できる。

【 0 1 4 4 】

## 2 . 5 ショートパケットの自動送信

さて、U S B では、ペイロードサイズが最大パケットサイズ未満であるショートパケットは特別な意味を有する。

【 0 1 4 5 】

即ち、ホストは、デバイスからショートパケットが転送されて来ると、データフェーズは終了したと判断し、ステータスフェーズに移行してしまう。従って、デバイスは、データフェーズを終了させる場合以外では、ショートパケットをホストに転送しないようデータ

10

20

30

40

50

転送を制御する。

【0146】

具体的には、デバイスは、バッファ60のデータ記憶領域62の残りデータサイズを常にチェックする。そして、残りデータサイズが最大パケットサイズ以上の場合には、残りデータの中から最大パケットサイズのデータパケットを作成して、ホストに転送する。そして、残りデータサイズが最大パケットサイズ未満と判断されると、ショートパケットの転送を許可して、その残りデータサイズのパケットをショートパケットしてホストに転送する。これにより、データフェーズが終了することになる。

【0147】

しかしながら、このように、データ記憶領域62の残りデータサイズを常にチェックし、  
10 チェック結果に応じた処理を行おうとすると、このチェック処理を行うファームウェア（CPU）の処理負荷が非常に重くなってしまふ。そして、このファームウェアの処理のオーバーヘッドに起因して、実効的なデータ転送が低下してしまふ事態を招く。

【0148】

そこで本実施形態では、E B U S（第2のバス）を介したデータ受信（DMA転送）が終了し、且つ、データ記憶領域62の残りデータサイズ（有効データ）が最大パケットサイズ未満である場合には、データ記憶領域62のショートパケットを、U S B（第1のバス）を介して自動的に（ファームウェアの処理を介さずにハードウェアの処理により）送信する手法を採用している。

【0149】

例えば図14に、I N T ランザクション（第2の方向のデータ転送）において、データ記憶領域62の残りデータサイズが最大パケットサイズ未満の場合のタイミング波形図を示す。  
20

【0150】

図14では、I 1に示すように、ショートパケットの自動送信（自動転送）の有効、無効を設定する信号であるA u t o E n S h o r t（図7参照）がアクティブ（Hレベル）に設定されている。このA u t o E n S h o r tのレベルは、例えばエニミュレーション時にファームウェア（CPU）が、A u t o E n S h o r tのレジスタに所与の値を書き込むことで設定される。この場合に、例えば、マストレージデバイスのようにショートパケットの自動送信を常に有効にすべきデバイスでは、A u t o E n S h o r tを固定的に  
30 アクティブに設定してもよい。

【0151】

そして、図14のI 1のようにA u t o E n S h o r tがアクティブに設定された状態で、I 2に示すようにE B U Sを介したデータ転送が終了し、I 3に示すように、最後の最大パケットサイズ（例えば512バイト）のパケットDn-1がU S Bを介して転送されたとする。

【0152】

この場合に、データ記憶領域62の残りデータサイズが最大パケットサイズ未満であると、I 4に示すように、ショートパケットの自動送信をS I E 3 0に指示するE n S h o r t P k tがアクティブになる。即ち、図7において、A u t o E n S h o r tがアクティブであり、残りデータサイズが最大パケットサイズか否かを示すH a v e M P Sが非アクティブである場合には、E n S h o r t P k tがアクティブになる。  
40

【0153】

そして、E n S h o r t P k tがアクティブになると、図14のI 5に示すように、データ記憶領域62に存在するショートパケットが、ファームウェアの処理が介在することなくハードウェア（S I E 3 0）によりホストに自動的に送信される。そして、ショートパケットがU S Bを介して転送されると、I 6に示すようにデータフェーズが終了して、ステータスフェーズに移行する。

【0154】

このように図14の本実施形態の手法では、A u t o E n S h o r tをアクティブに設定  
50

することで、データ記憶領域 62 のショートパケットをハードウェアにより自動送信できる。従って、データ記憶領域 62 の残りデータサイズをチェックする処理を、ファームウェアが行う必要がなくなる。この結果、ファームウェアの処理負荷を軽減でき、実効的なデータ転送速度を向上できる。

【0155】

特に、デバイス側では、低コスト化の要請から、例えば 20 ~ 50 MHz 程度のクロック周波数で動作する安価な CPU が使用される場合が多い。従って、残りデータサイズのチェック処理を CPU 上で動作するファームウェアが行うと、チェック処理に非常に時間を費やしてしまい、I 6 に示すステータスフェーズへの移行タイミングが非常に遅れてしまう可能性がある。

10

【0156】

これに対して、図 14 の本実施形態の手法では、残りデータのチェック処理をファームウェアが行わなくて済むため、早い段階でステータスフェーズに移行できるようになる。この結果、残りデータサイズのチェック処理をファームウェアで行う場合に比べて、実効的なデータ転送速度を格段に向上できる。

【0157】

なお、本実施形態では、AutoEnShort の設定を変更することで、ショートパケットの自動送信の有効、無効を任意に設定できるようになっている。

【0158】

例えば、ハードディスクドライブや CD ドライブなどのマスストレージデバイスでは、ショートパケットの自動送信が常に有効になるように、AutoEnShort を設定する。このようにすれば、ファームウェアの処理を介在させることなく、ショートパケットを自動送信して、ステータスフェーズに移行できるようになる。

20

【0159】

一方、スキャナなどのデバイスのように、E B U S (第 2 のバス) を介して受信されたデータに、データサイズなどの情報が付加されて、U S B (第 1 のバス) を介して送信されるようなデバイスでは、AutoEnShort を非アクティブ (L レベル) に設定する。

【0160】

即ち、スキャナでは、データの取り込みが完了した後でなければ、最終的なデータサイズが確定しない。従って、データ転送制御装置は、スキャナからの受信データの最後に、データサイズを含む制御情報 (ヘッダ、フッタ) を付加して、ホストに転送することになる。このため、E B U S を介して転送されるデータのサイズと、U S B を介して転送されるデータのサイズが一致せず、ショートパケットをホストに無条件に転送することは許されなくなり、AutoEnShort を非アクティブに設定する必要が生じる。

30

【0161】

本実施形態のように、ショートパケットの自動送信の有効、無効を任意に設定可能にすることで、ハードディスクドライブなどのマスストレージデバイスのみならず、スキャナなどのデバイスにおけるデータ転送にも柔軟に対応できるようになる。

【0162】

## 2.6 ショートパケットの存在の通知

さて、図 14 では、AutoEnShort 信号を用いて、データ記憶領域 62 のショートパケットを自動送信している。

40

【0163】

これに対して、データ記憶領域 62 でのショートパケットの存在を CPU (ファームウェア) に知らせるようにしてもよい。具体的には、E B U S (第 2 のバス) を介したデータ受信が終了し、且つ、データ記憶領域 62 の残りデータサイズが最大パケットサイズ未満である場合に、データ記憶領域 62 にショートパケットが存在することを、割り込み等を用いて CPU (処理手段) に通知するようにする。

【0164】

50

このような手法を採用する場合での、S I E 3 0、エンドポイント管理回路 4 0、バッファ管理回路 5 0、バッファ 6 0、バルク転送管理回路 7 0、D M A C 8 0の詳細な接続例を図 1 5 に示す。

【 0 1 6 5 】

図 1 5 が、前述の図 7 と異なるのは、図 1 5 では、バルク転送管理回路 7 0 が C P U ( C P U がアクセス可能なレジスタ) に対して割り込み信号 S h o r t I N T を出力している点と、E n S h o r t P k t が C P U により設定されて、S I E 3 0 に入力される点である。

【 0 1 6 6 】

ここで、S h o r t I N T は、データ記憶領域 6 2 にショートパケットが存在することを C P U に伝えるための割り込み信号であり、E n S h o r t P k t は、ショートパケットの転送を C P U が S I E 3 0 に指示するための信号である。

【 0 1 6 7 】

そして、例えば、D M A \_ R u n n i n g が非アクティブ(データ受信終了)になった後に、H a v e M P S が非アクティブ(最大パケットサイズ未満)になると、S h o r t I N T がアクティブになり、転送可能なショートパケットがデータ記憶領域 6 2 に存在することが、C P U に対して通知される。

【 0 1 6 8 】

すると、C P U は、ショートパケットの転送のための判断処理を行い、ショートパケットを転送してよいと判断した場合には、E n S h o r t P k t をアクティブに設定する。これにより、S I E 3 0 は、データ記憶領域 6 2 に存在するショートパケットを読み出して、U S B を介してホストに転送する処理を行うことになる。

【 0 1 6 9 】

図 1 6 に、I N トランザクション(第 2 の方向のデータ転送)において、データ記憶領域 6 2 の残りデータサイズが最大パケットサイズ未満の時のタイミング波形図を示す。

【 0 1 7 0 】

図 1 6 では、J 1 に示すように、ショートパケットの自動送信の有効、無効を設定する A u t o E n S h o r t は非アクティブ(L レベル)になっている。この状態で、J 2 に示すように E B U S を介したデータ転送が終了し、J 3 に示すように、最後の最大パケットサイズのパケット D n - 1 が U S B を介して転送されたとする。

【 0 1 7 1 】

この場合に、データ記憶領域 6 2 の残りデータサイズが最大パケットサイズ未満であると、J 4 に示すように、ショートパケットの存在を C P U に通知する割り込み信号 S h o r t I N T がアクティブになる。即ち、図 1 5 において、D M A \_ R u n n i n g が非アクティブであり、残りデータサイズが最大パケットサイズか否かを示す H a v e M P S が非アクティブである場合には、S h o r t I N T がアクティブになる。

【 0 1 7 2 】

すると、この S h o r t I N T を受けた C P U (ファームウェア) は、例えば、データサイズなどの情報を受信データに付加する処理(付加情報をデータ記憶領域 6 2 に書き込む処理)などを行う。そして、その後、J 5 に示すように E n S h o r t P k t をアクティブにする。これにより、J 6 に示すように、データ記憶領域 6 2 のデータが S I E 3 0 によりホストに転送される。そして、J 7 に示すようにデータフェーズが終了して、ステータスフェーズに移行する。

【 0 1 7 3 】

このように図 1 6 の本実施形態の手法では、S h o r t I N T を用いて、データ記憶領域 6 2 でのショートパケットの存在を、C P U 上で動作するファームウェアに知らせている。これにより、例えば、データサイズなどの情報が受信データの最後に付加されるスキャナなどのデバイスにおいても、適正なデータ転送を実現できる。

【 0 1 7 4 】

2 . 7 ファームウェアの処理

10

20

30

40

50

図17(A)に、INTランザクションにおける比較例のファームウェアの処理のフローチャートを示す。また、図17(B)に、ショートパケットを自動送信する本実施形態の手法におけるファームウェアの処理のフローチャートを示す。

【0175】

図17(A)の比較例では、ファームウェアは、まず、DMA\_Startを1(Hレベル)にセットして、EBUSでのDMA転送をスタートさせる(ステップS1)。そして、DMA\_Runningをチェックして、EBUSでのDMA転送が終了したか否かを判断する(ステップS2)。

【0176】

DMA転送が終了したと判断された場合には、IntranACKを用いて、USB側からのACKを検出する(ステップS3)。そして、ACKが検出された場合には、バッファのデータ記憶領域の残りデータサイズRDSをチェックする(ステップS4)。そして、RDS 最大パケットサイズMaxPacketSizeの場合は、ステップS3に戻り、次のACKが来るのを待つ。

【0177】

一方、RDS < MaxPacketSizeの場合には、EnShortPktを1にセットする(ステップS5)。これにより、データ記憶領域のショートパケットがホストに転送される。そして、その後ステップS3に戻り、次のACKが来るのを待つ。

【0178】

また、RDS = 0の場合には、データ記憶領域のデータが全て転送されたと考えられるので、EBUS側のステータスを取得する(ステップS6)。そして、この取得されたステータスに基づきCSWを作成して、USB側に送るべきステータスをセットする(ステップS7)。

【0179】

図17(B)の本実施形態では、ファームウェアは、まず、DMA\_Startを1にセットして、EBUSでのDMA転送をスタートさせる(ステップS11)。なお、AutoEnShortは、DMA転送の前に1にセットされている。そして、DMA\_Runningをチェックして、EBUSでのDMA転送が終了したか否かを判断する(ステップS12)。

【0180】

DMA転送が終了したと判断された場合には(図14のI2)、EBUS側のステータスを取得する(ステップS13)。そして、データフェーズ(全データの転送)が終了したか否かを判断する(ステップS14)。即ち、バッファのデータ記憶領域が空になり、最後のパケットに対するACKがホストから返ってきたか否かを判断する。そして、データフェーズが終了したと判断した場合には、ステップS13で取得されたステータスに基づきCSWを作成して、USB側に送るべきステータスをセットする(ステップS15)。

【0181】

このように、図17(B)の本実施形態では、図17(A)の比較例で必要な残りデータサイズのチェック処理(ステップS4)が不要になる。また、EnShortPktをアクティブにセットする処理も、ハードウェアにより行われ、ファームウェアが行わなくても済むようになる。従って、ファームウェアの処理負荷を軽減でき、実効的なデータ転送速度を向上できる。

【0182】

図18に、割り込みを用いてショートパケットの存在をファームウェアに通知する手法(図15、図16)におけるファームウェアの処理のフローチャートを示す。

【0183】

図18では、ファームウェアは、まず、EBUSでのDMA転送をスタートさせ、その後、DMA転送が終了したか否かを判断する(ステップS21、S22)。

【0184】

DMA転送が終了したと判断された場合には、割り込み信号ShortINTが検出(図

10

20

30

40

50

16のJ4)されたか否かを判断する(ステップS23)。そして、ShortINTが検出された場合には、DMA転送の受信データに付加すべき情報(例えばデータサイズ)を、データ記憶領域(FIFO)に書き込む(ステップS24)。

【0185】

そして、全ての情報が付加されたと判断されると(ステップS25)、EnShortPktを1にセットする(ステップS26)。

【0186】

次に、USB側からのACKを検出する(ステップS27)。そして、ACKが検出された場合には、データ記憶領域の残りデータサイズRDSが0か否かをチェックする(ステップS28)。そして、RDSが0でない場合には、ステップS27に戻り、次のACKが検出されるのを待つ。

10

【0187】

一方、RDSが0の場合には、データ記憶領域の全てのデータが転送されたと考えられるので、EBUS側のステータスを取得する(ステップS29)。そして、この取得されたステータスに基づきCSWを作成して、USB側に送るべきステータスをセットする(ステップS30)。

【0188】

### 3. 電子機器

次に、本実施形態のデータ転送制御装置を含む電子機器の例について説明する。

【0189】

例えば図19(A)に電子機器の1つであるプリンタの内部ブロック図を示し、図20(A)にその外観図を示す。CPU(マイクロコンピュータ)510はシステム全体の制御などを行う。操作部511はプリンタをユーザが操作するためのものである。ROM516には、制御プログラム、フォントなどが格納され、RAM517はCPU510のワーク領域として機能する。DMAC518は、CPU510を介さずにデータ転送を行うためのDMAコントローラである。表示パネル519はプリンタの動作状態をユーザに知らせるためのものである。

20

【0190】

USBを介してパーソナルコンピュータなどの他のデバイスから送られてきたシリアル印字データは、データ転送制御装置500によりパラレル印字データに変換される。そして、変換後のパラレル印字データは、CPU510又はDMAC518により、印字処理部(プリンタエンジン)512に送られる。そして、印字処理部512においてパラレル印字データに対して所与の処理が施され、プリントヘッドなどからなる印字部(データの出力処理を行う装置)514により紙に印字されて出力される。

30

【0191】

図19(B)に電子機器の1つであるスキャナの内部ブロック図を示し、図20(B)にその外観図を示す。CPU520はシステム全体の制御などを行う。操作部521はスキャナをユーザが操作するためのものである。ROM526には制御プログラムなどが格納され、RAM527はCPU520のワーク領域として機能する。DMAC528はDMAコントローラである。

40

【0192】

光源、光電変換器などからなる画像読み取り部(データの取り込み処理を行う装置)522により原稿の画像が読み取られ、読み取られた画像のデータは画像処理部(スキャナエンジン)524により処理される。そして、処理後の画像データは、CPU520又はDMAC528によりデータ転送制御装置500に送られる。データ転送制御装置500は、このパラレルの画像データをシリアルデータに変換し、USBを介してパーソナルコンピュータなどの他のデバイスに送信する。

【0193】

図19(C)に電子機器の1つであるCD-RWドライブの内部ブロック図を示し、図20(C)にその外観図を示す。CPU530はシステム全体の制御などを行う。操作部5

50

31はCD-RWをユーザが操作するためのものである。ROM536には制御プログラムなどが格納され、RAM537はCPU530のワーク領域として機能する。DMAC538はDMAコントローラである。

【0194】

レーザ、モータ、光学系などからなる読み取り&書き込み部(データの取り込み処理を行う装置又はデータの記憶処理を行うための装置)533によりCD-RW532から読み取られたデータは、信号処理部534に入力され、エラー訂正処理などの所与の信号処理が施される。そして、信号処理が施されたデータが、CPU530又はDMAC538によりデータ転送制御装置500に送られる。データ転送制御装置500は、このパラレルのデータをシリアルデータに変換し、USBを介してパーソナルコンピュータなどの他の

10

【0195】

一方、USBを介して他のデバイスから送られてきたシリアルデータは、データ転送制御装置500によりパラレルデータに変換される。そして、このパラレルデータは、CPU530又はDMAC538により信号処理部534に送られる。そして、信号処理部534においてこのパラレルデータに対して所与の信号処理が施され、読み取り&書き込み部533によりCD-RW532に記憶される。

【0196】

なお、図19(A)、(B)、(C)において、CPU510、520、530の他に、データ転送制御装置500でのデータ転送制御のためのCPUを別に設けるようにしても

20

【0197】

本実施形態のデータ転送制御装置を電子機器に用いれば、CPU上で動作するファームウェアの処理負荷が軽減され、安価なCPUを用いることが可能になる。更に、データ転送制御装置の低コスト化、小規模化を図れるため、電子機器の低コスト化、小規模化も図れるようになる。

【0198】

また、本実施形態のデータ転送制御装置を電子機器に用いれば、USB2.0におけるHSモードでのデータ転送を適正に行うことが可能になる。従って、ユーザがパーソナルコンピュータなどによりプリントアウトの指示を行った場合に、少ないタイムラグで印字が完了するようになる。また、スキャナへの画像取り込みの指示の後に、少ないタイムラグで読み取り画像をユーザは見るようになる。また、CD-RWからのデータの読み取りや、CD-RWへのデータの書き込みを高速に行うことが可能になる。

30

【0199】

なお本実施形態のデータ転送制御装置を適用できる電子機器としては、上記以外にも例えば、種々の光ディスクドライブ(CD-ROM、DVD)、光磁気ディスクドライブ(MO)、ハードディスクドライブ、TV、VTR、ビデオカメラ、オーディオ機器、電話機、プロジェクタ、パーソナルコンピュータ、電子手帳、ワードプロセッサなど種々のものを考えることができる。

【0200】

なお、本発明は本実施形態に限定されず、本発明の要旨の範囲内で種々の変形実施が可能である。

40

【0201】

例えば、本発明のデータ転送制御装置の構成は、図6、図7、図15に示す構成に限定されるものではなく、種々の変形実施が可能である。

【0202】

また本実施形態では、USBのBulk-Only仕様への適用例を説明したが、本発明が適用されるのは、USBのBulk-Only仕様に限定されるものではない。

【0203】

また、本発明は、USB2.0でのデータ転送に適用されることが特に望ましいが、これ

50

に限定されるものではない。例えばUSB 2.0と同様の思想に基づく規格やUSB 2.0を発展させた規格におけるデータ転送にも本発明は適用できる。

【図面の簡単な説明】

【図1】図1(A)、(B)、(C)、(D)は、USBのエンドポイントやトランザクション構成について説明するための図である。

【図2】図2(A)、(B)は、CBI仕様とBulk-Only仕様について説明するための図である。

【図3】CBWのフォーマットについて示す図である。

【図4】CSWのフォーマットについて示す図である。

【図5】図5(A)、(B)は、Bulk-Onlyにおけるデータの書き込み処理、読み出し処理について説明するための図である。 10

【図6】本実施形態のデータ転送制御装置の構成例を示す図である。

【図7】本実施形態のデータ転送制御装置の詳細な接続例を示す図である。

【図8】図8(A)、(B)、(C)、(D)は、OUTトランザクション、PINGトランザクションについて説明するためのタイミング波形図である。

【図9】図9(A)、(B)、(C)は、INTトランザクションについて説明するためのタイミング波形図である。

【図10】HDDへの送信時におけるDMA転送について説明するためのタイミング波形図である。

【図11】HDDからの受信時におけるDMA転送について説明するためのタイミング波形図である。 20

【図12】OUTトランザクション(第1の方向のデータ転送)における本実施形態の動作について説明するためのタイミング波形図である。

【図13】INTトランザクション(第2の方向のデータ転送)における本実施形態の動作について説明するためのタイミング波形図である。

【図14】ショートパケットを自動送信する手法について説明するためのタイミング波形図である。

【図15】本実施形態のデータ転送制御装置の詳細な他の接続例を示す図である。

【図16】ショートパケットの存在をファームウェアに通知する手法について説明するためのタイミング波形図である。 30

【図17】図17(A)、(B)は、比較例と本実施形態のファームウェアの処理を示すフローチャートである。

【図18】ショートパケットの存在をファームウェアに通知する手法におけるファームウェアの処理を示すフローチャートである。

【図19】図19(A)、(B)、(C)は、種々の電子機器の内部ブロック図の例である。

【図20】図20(A)、(B)、(C)は、種々の電子機器の外観図の例である。

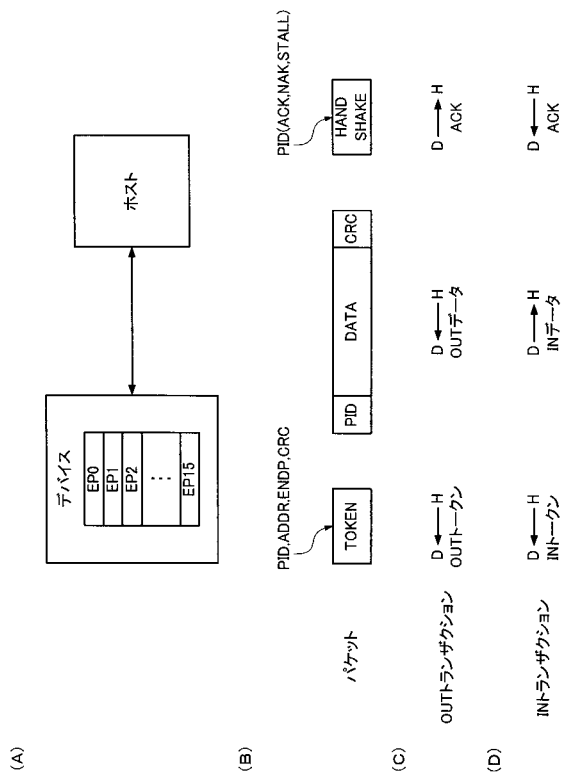
【符号の説明】

20	トランシーバマクロ	
22	トランシーバ回路	40
24	クロック生成回路	
30	SIE	
32	パケットハンドラ回路	
33	CRC処理回路	
34	サスペンド&レジューム回路	
36	トランザクション管理回路	
40	エンドポイント管理回路	
41	残りデータサイズカウンタ	
50	バッファ管理回路	
60	バッファ	50



- 6 2 データ記憶領域
- 7 0 バルク転送管理回路
- 8 0 D M A C
- 8 2 D M A カウンタ

【 図 1 】



【 図 2 】

- (A) CBI
- EP0 (コントロール) コントロール, コマンド
  - EP1 (バルクアウト) OUTデータ
  - EP2 (バルクイン) INデータ
  - EP3 (インタラプト) インタラプトIN
- (B) Bulk-Only
- EP0 (コントロール) コントロール
  - EP1 (バルクアウト) コマンド (CBW), OUTデータ (A1, A2)
  - EP2 (バルクイン) ステータス (CSW), INデータ (A3, A4)

【図3】

バイト	7	6	5	4	3	2	1	0
0~3	dCBWSignature							
4~7	dCBWTag							
8~11	dCBWDataTransferLength							
12	bmCBWFlags							
13	予約済み(0)				bCBWLUN			
14	予約済み(0)				bCBWCBLLength			
15~30	CBWCB							

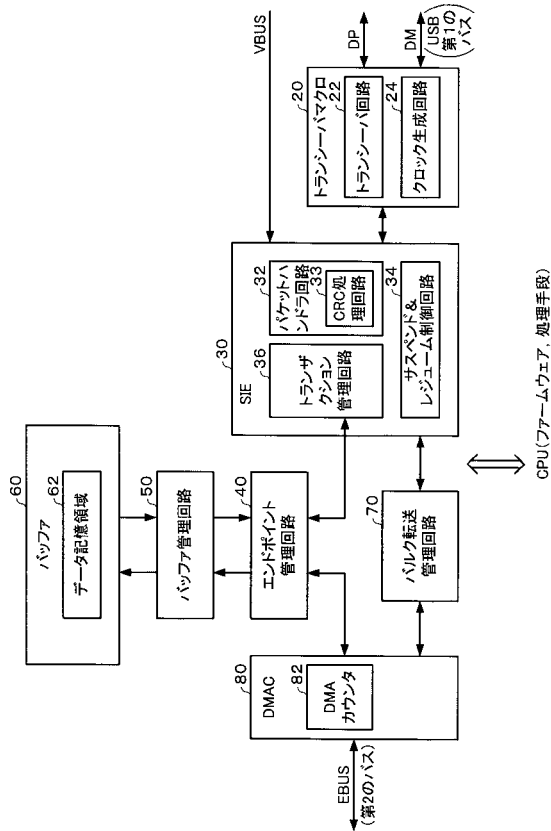
CBW

【図4】

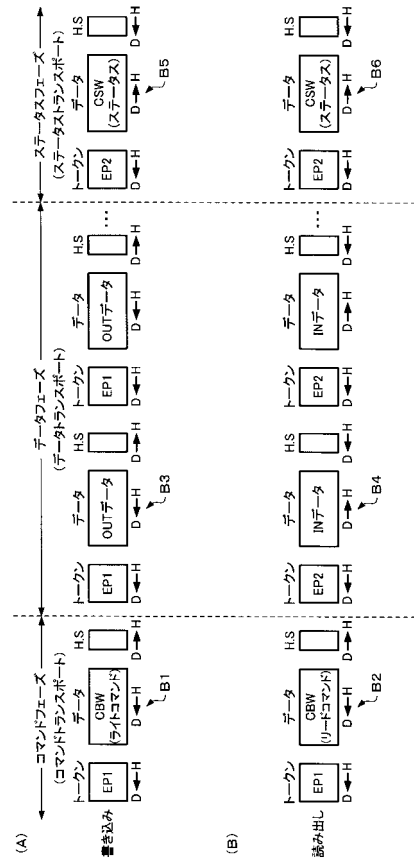
バイト	7	6	5	4	3	2	1	0
0~3	dCSWSignature							
4~7	dCSWTag							
8~11	dCSWDataResidue							
12	bCSWStatus							

CSW

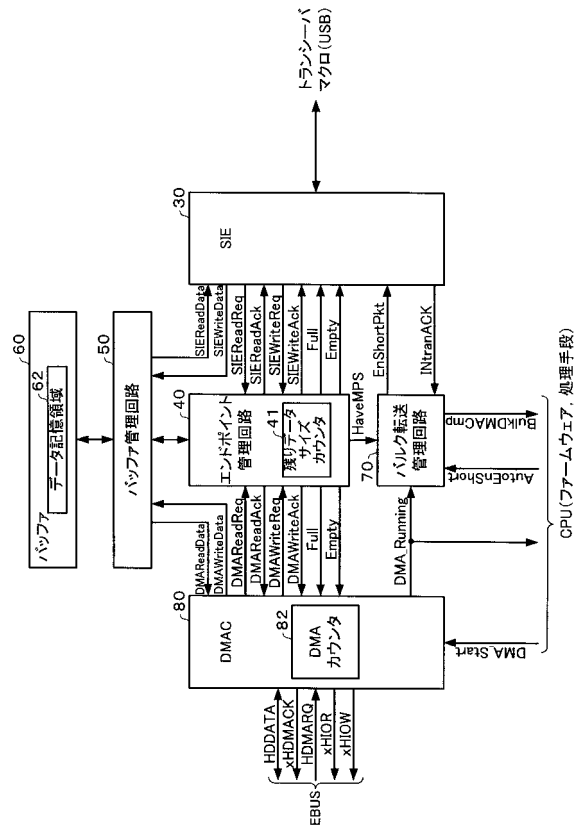
【図6】



【図5】

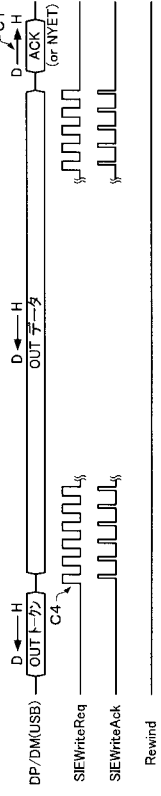


【図7】

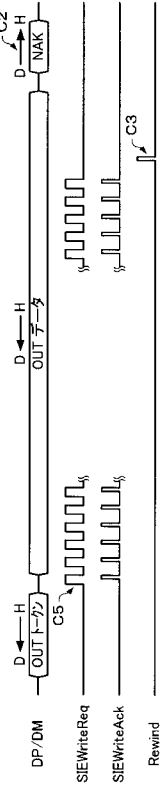


【 8 】

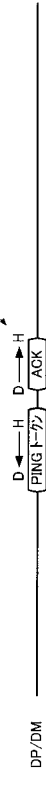
(A) OUTランザクション(残りエリアサイズ≧データサイズ(データ載))



(B) OUTランザクション(残りエリアサイズ<データサイズ)



(C) PINGランザクション(残りエリアサイズ≧データサイズ)

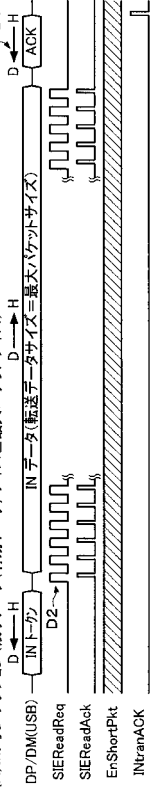


(D) PINGランザクション(残りエリアサイズ<データサイズ)

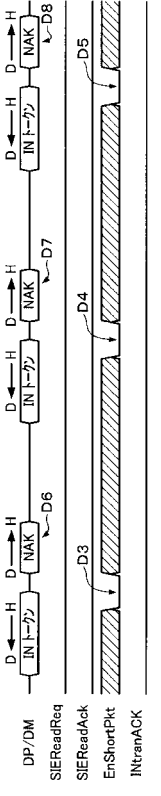


【 9 】

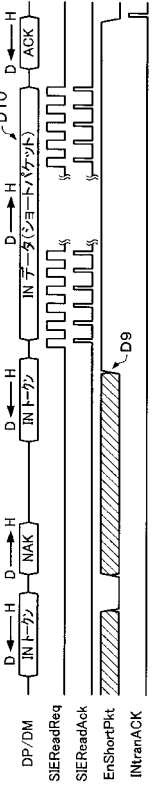
(A) INTランザクション(残りデータ(有効データ)サイズ≧最大パケットサイズ)



(B) INTランザクション(残りデータサイズ<最大パケットサイズ, EnShortPkt=L)

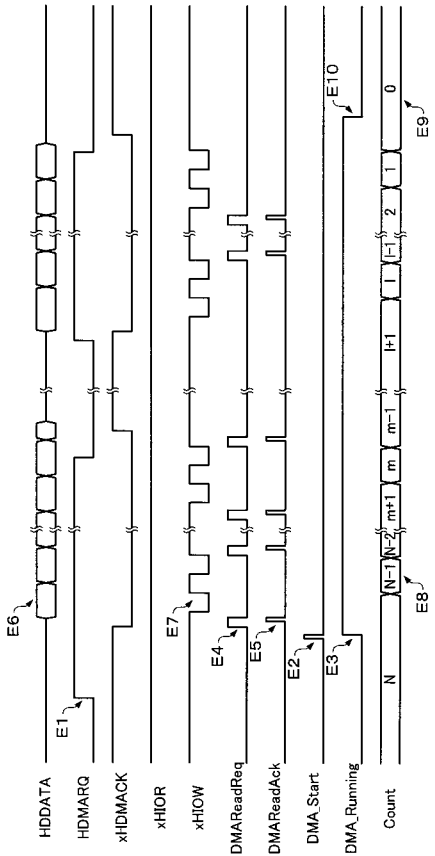


(C) INTランザクション(残りデータサイズ<最大パケットサイズ, EnShortPkt=H)



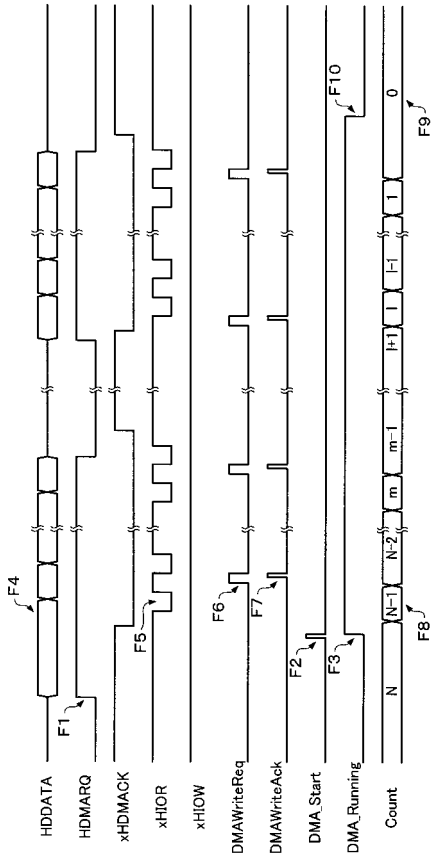
【 10 】

DMA転送(HDDへの送信)



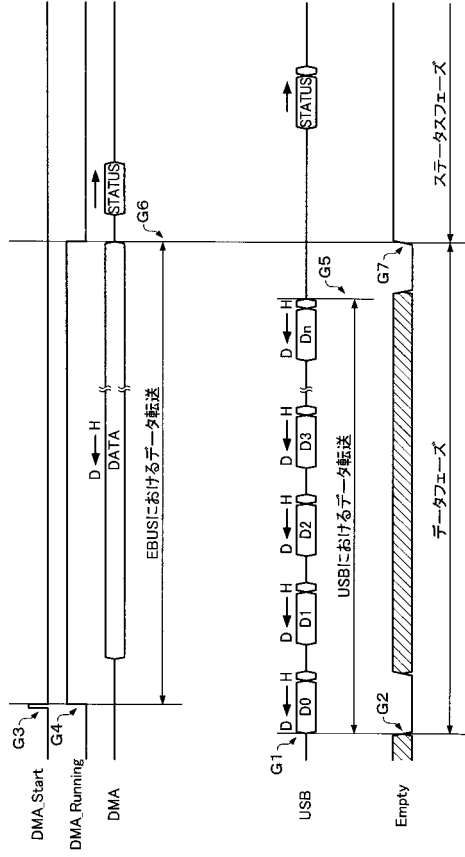
【 11 】

DMA転送(HDDからの受信)



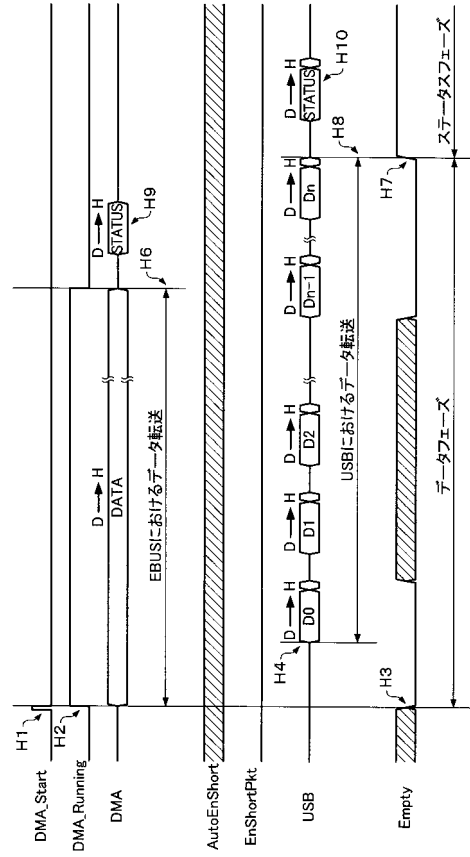
【図 1 2】

デバイス→ホスト(OUT)トランザクション、第1の方向のデータ転送



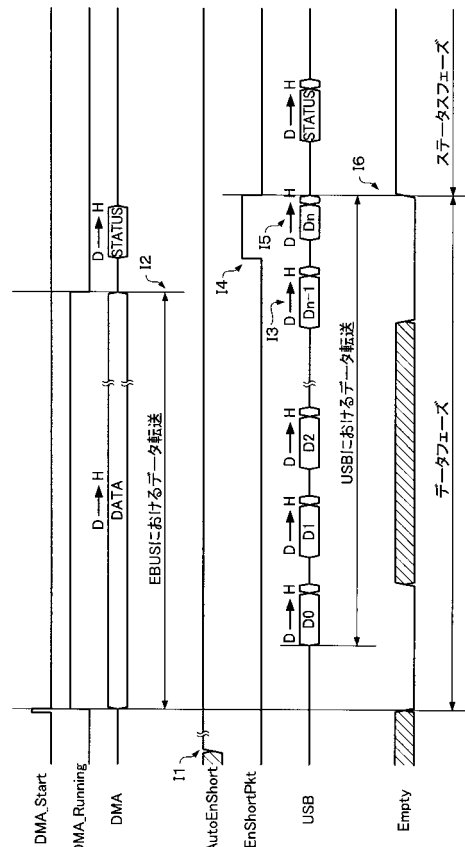
【図 1 3】

デバイス→ホスト(IN)トランザクション、第2の方向のデータ転送

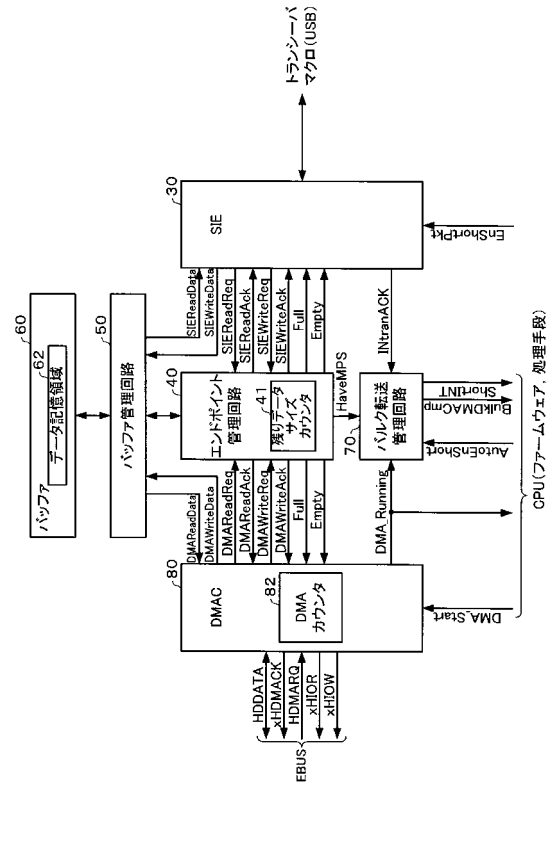


【図 1 4】

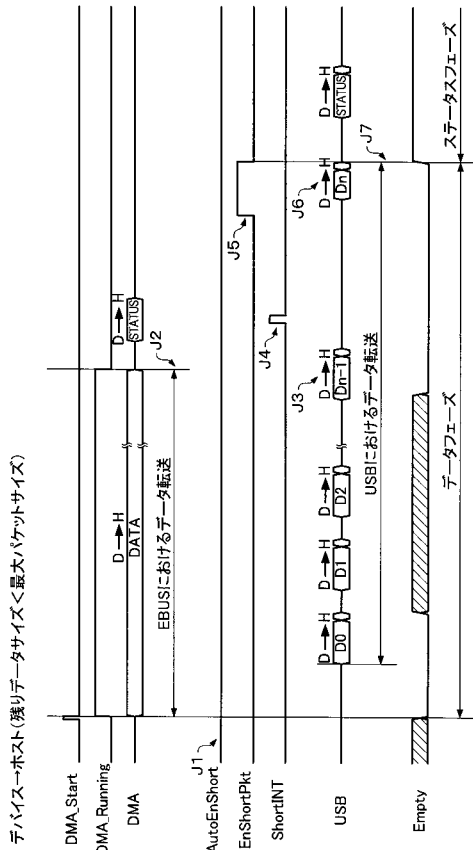
デバイス→ホスト(残りデータサイズ<最大パケットサイズ)



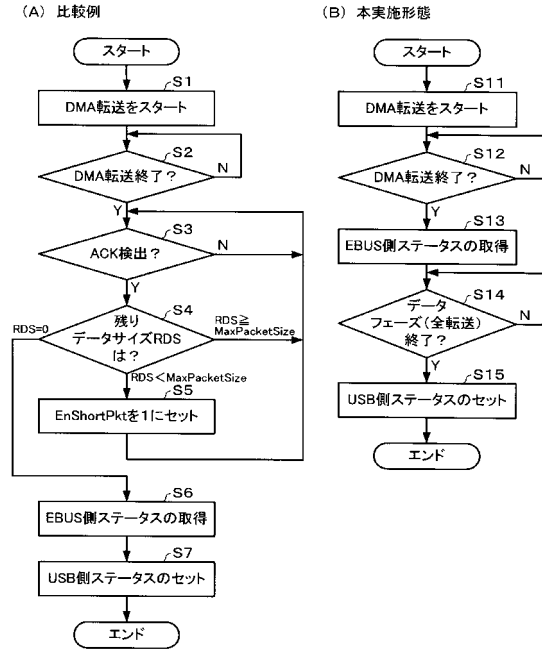
【図 1 5】



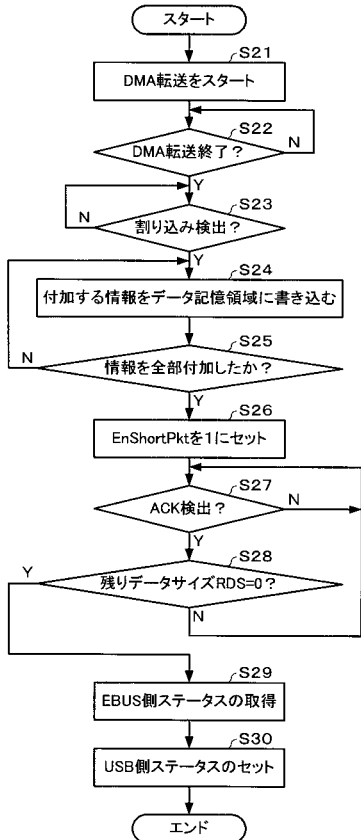
【図16】



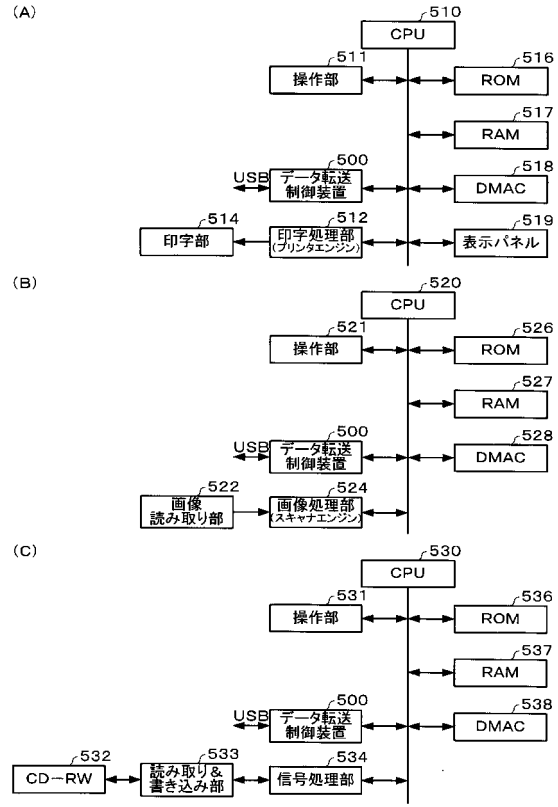
【図17】



【図18】

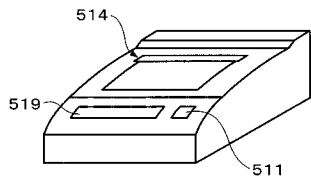


【図19】

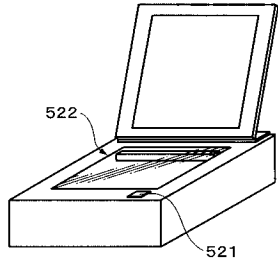


【 図 20 】

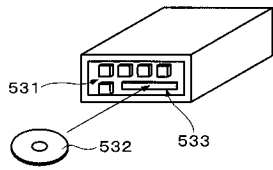
(A)



(B)



(C)



---

フロントページの続き

審査官 鈴木 匡明

(56)参考文献 特開2000-276435(JP,A)  
特開昭59-000721(JP,A)

(58)調査した分野(Int.Cl., DB名)  
G06F 13/28  
G06F 13/38