

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3636160号
(P3636160)

(45) 発行日 平成17年4月6日(2005.4.6)

(24) 登録日 平成17年1月14日(2005.1.14)

(51) Int. Cl.⁷

G06F 13/38

F I

G06F 13/38 310B

G06F 13/38 350

請求項の数 12 (全 37 頁)

<p>(21) 出願番号 特願2002-126708 (P2002-126708)</p> <p>(22) 出願日 平成14年4月26日(2002.4.26)</p> <p>(65) 公開番号 特開2003-316728 (P2003-316728A)</p> <p>(43) 公開日 平成15年11月7日(2003.11.7)</p> <p>審査請求日 平成15年2月18日(2003.2.18)</p> <p>特許法第30条第1項適用 Design Wave 2002年4月号に文書で発表</p>	<p>(73) 特許権者 000002369 セイコーエプソン株式会社 東京都新宿区西新宿2丁目4番1号</p> <p>(74) 代理人 100090479 弁理士 井上 一</p> <p>(74) 代理人 100090387 弁理士 布施 行夫</p> <p>(74) 代理人 100090398 弁理士 大淵 美千栄</p> <p>(72) 発明者 ▲齋▼藤 伸之 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</p> <p>(72) 発明者 久保田 慎介 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</p> <p style="text-align: right;">最終頁に続く</p>
--	--

(54) 【発明の名称】 データ転送制御装置、電子機器及びデータ転送制御方法

(57) 【特許請求の範囲】

【請求項1】

U S B (Universal Serial Bus) を介してデータを送受信するデータ転送制御装置であって、

転送条件情報が設定される転送条件レジスタと、

転送条件レジスタに設定される転送条件情報に基づいて、コントロール転送を含むデータ転送を行う転送コントローラと、

転送データを記憶するパケットバッファのアクセス制御を行うバッファコントローラとを含み、

前記転送コントローラは、

自動コントロール転送の開始が指示された場合に、セットアップステージのトランザクションを自動発生してセットアップステージのパケットを自動転送し、次に、転送すべきデータが存在する場合にはデータステージのトランザクションを自動発生してデータステージのパケットを自動転送し、次に、ステータスステージのトランザクションを自動発生してステータスステージのパケットを自動転送し、

前記転送条件レジスタは、

データステージの有無の情報と、データステージでの転送方向が、転送条件情報として設定され、

前記転送コントローラは、

セットアップステージのトランザクションが完了した場合に、前記転送条件情報として

10

20

設定されたデータステージでの転送方向とデータステージの有無の情報に基づいて、データステージのINトランザクションの実行、データステージのOUTトランザクションの実行、或いはデータステージ無しのいずれかを選択する回路を含むことを特徴とするデータ転送制御装置。

【請求項 2】

請求項 1 において、
前記転送条件レジスタに、
デバイスリクエストのデータと、データステージでの転送データのトータルサイズが、
転送条件情報として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 3】

請求項 1 又は 2 において、
前記転送条件レジスタに、
マックスパケットサイズが、転送条件情報として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 4】

請求項 1 乃至 3 のいずれかにおいて、
前記転送コントローラは、
ステータスステージのトランザクションが完了した場合に、処理部に対して割り込みを発生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、
前記転送コントローラは、
エンドポイントからストールを受信した場合に、自動コントロール転送を中止し、処理部に対して割り込みを発生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 1 乃至 5 のいずれかにおいて、
前記転送コントローラは、
自動コントロール転送の開始が指示された場合に、セットアップステージのトランザクションの開始を指示する回路と、
データステージのINトランザクションの実行が選択された場合には、データステージのINトランザクションの開始を指示し、データステージのOUTトランザクションの実行が選択された場合には、データステージのOUTトランザクションの開始を指示する回路と、
開始したデータステージのINトランザクションが完了した場合には、ステータスステージのOUTトランザクションの開始を指示し、開始したデータステージのOUTトランザクションが完了した場合、或いはデータステージ無しが選択された場合には、ステータスステージのINトランザクションの開始を指示する回路とを含むことを特徴とするデータ転送制御装置。

【請求項 7】

請求項 1 乃至 6 のいずれかにおいて、
前記パケットバッファに、
各エンドポイントとの間で転送されるデータが各パイプ領域に記憶される複数のパイプ領域が確保され、
各パイプ領域と各エンドポイントとの間でのデータ転送の転送条件情報が、複数の転送条件レジスタの各転送条件レジスタに設定され、
前記転送コントローラは、
コントロール転送のパイプ領域に対応する転送条件レジスタに設定された転送条件情報に基づいて、コントロール転送のトランザクションを自動発生し、コントロール転送のパイプ領域とコントロール転送のエンドポイントとの間で、自動コントロール転送を行うことを特徴とするデータ転送制御装置。

10

20

30

40

50

【請求項 8】

請求項 7 において、
前記パイプ領域は、
コントロール転送のエンドポイントに専用のパイプ領域と、任意のエンドポイントに割り当て可能な汎用のパイプ領域を含み、
前記転送コントローラは、
前記専用のパイプ領域を用いて、自動コントロール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 9】

請求項 1 乃至 8 のいずれかにおいて、
ホストの役割として動作するホスト動作のステートと、ペリフェラルの役割として動作するペリフェラル動作のステートを含む複数のステートの制御を行うステートコントローラを含み、
前記転送コントローラは、
ホスト動作時において、ホストとしてのデータ転送を行うホストコントローラと、
ペリフェラル動作時において、ペリフェラルとしてのデータ転送を行うペリフェラルコントローラとを含み、
前記ホストコントローラは、
ホスト動作時に自動コントロール転送を行うことを特徴とするデータ転送制御装置。

10

【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 O T G (O n - T h e - G o) 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。

20

【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、
前記データ転送制御装置及び U S B 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、
前記データ転送制御装置のデータ転送を制御する処理部と、
を含むことを特徴とする電子機器。

【請求項 12】

U S B (Universal Serial Bus) を介してデータを送受信するデータ転送制御装置のためのデータ転送制御方法であって、
転送条件レジスタに転送条件情報を設定し、
前記転送条件レジスタに設定される転送条件情報に基づいて、コントロール転送を含むデータ転送を行い、
転送データを記憶するパケットバッファのアクセス制御を行うと共に、
自動コントロール転送の開始が指示された場合に、セットアップステージのトランザクションを自動発生してセットアップステージのパケットを自動転送し、次に、転送すべきデータが存在する場合にはデータステージのトランザクションを自動発生してデータステージのパケットを自動転送し、次に、ステータスステージのトランザクションを自動発生してステータスステージのパケットを自動転送し、
前記転送条件レジスタに、データステージの有無の情報と、データステージでの転送方向を、転送条件情報として設定し、
セットアップステージのトランザクションが完了した場合に、前記転送条件情報として設定されたデータステージでの転送方向とデータステージの有無の情報に基づいて、データステージの I N トランザクションの実行、データステージの O U T トランザクションの実行、或いはデータステージ無しのいずれかを選択することを特徴とするデータ転送制御方法。

30

40

【発明の詳細な説明】

【0001】

50

【発明の属する技術分野】

本発明は、データ転送制御装置、電子機器及びデータ転送制御方法に関する。

【0002】**【背景技術及び発明が解決しようとする課題】**

H S (High Speed) モードをサポートする U S B (Universal Serial Bus) 2 . 0 の市場が順調に拡大しているなか、U S B Implementers Forum (U S B - I F) によって、U S B O n - T h e - G o (O T G) と呼ばれるインターフェース規格が策定された。U S B 2 . 0 を拡張する形で策定された O T G 規格 (O T G 1 . 0) は、U S B インターフェースの新たな付加価値を生む可能性を秘めており、その特質を生かしたアプリケーションの登場が待たれている。

10

【0003】

この O T G によれば、これまで U S B を介してホスト (パーソナルコンピュータ等) に接続されていたペリフェラル (周辺機器) に、ホスト機能を持たせることができる。これにより、ペリフェラル同士を U S B で接続してデータを転送することが可能になり、例えばデジタルカメラとプリンタとをダイレクトに接続して、デジタルカメラの画像を印刷することが可能になる。また、デジタルカメラやデジタルビデオカメラをストレージ装置に接続して、データを保存することが可能になる。

【0004】

しかしながら、O T G によりホスト機能を持たせるペリフェラルには、低性能の C P U (処理部) が組み込まれているのが一般的である。従って、ホスト機能の追加により、ペリフェラルが有する C P U (ファームウェア) の処理負荷が重くなったり、処理が複雑化すると、他の処理に支障が生じたり、機器の設計期間が長期化するなどの問題が生じる。

20

【0005】

また、O T G によりホスト機能を持たせることで、データ転送制御装置が大規模化してしまうと、データ転送制御装置の高コスト化や、データ転送制御装置が組み込まれる機器 (電子機器) の高コスト化等を招く。

【0006】

また、U S B では 4 種類の転送方式が定義されており、その中にはコントロール転送と呼ばれる転送方式がある。このコントロール転送は、ペリフェラルの制御に必要な情報をやり取りするための転送方式であり、U S B 機器はこのコントロール転送を必ずサポートしなければならない。

30

【0007】

そして、このコントロール転送は、セットアップステージ、データステージ、ステータスステージで構成されるが、これを各ステージ毎に制御、管理すると、C P U (処理部、ファームウェア) の処理負荷が重くなってしまふという問題が生じる。

【0008】

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、コントロール転送時の処理部の処理負荷の軽減等を図れるデータ転送制御装置、電子機器及びデータ転送制御方法を提供することにある。

【0009】**【課題を解決するための手段】**

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、転送条件情報が設定される転送条件レジスタと、転送条件レジスタに設定される転送条件情報に基づいて、コントロール転送を含むデータ転送を行う転送コントローラと、転送データを記憶するパケットバッファのアクセス制御を行うバッファコントローラとを含み、前記転送コントローラが、自動コントロール転送の開始が指示された場合に、セットアップステージのトランザクションを自動発生してセットアップステージのパケットを自動転送し、次に、転送すべきデータが存在する場合にはデータステージのトランザクションを自動発生してデータステージのパケットを自動転送し、次に、ステータスステージのトランザクションを自動発生してステータスステージのパケットを自動転送するデータ転送制御装置に関係

40

50

する。

【 0 0 1 0 】

本発明では、処理部（ファームウェア）等により転送条件情報が設定され、自動コントロール転送の開始が指示されると、セットアップステージのトランザクション、データステージのトランザクション（データステージ有りの場合）、ステータスステージのトランザクションがシーケンシャルに自動発生する。そして、各ステージのトランザクションに必要なパケットが自動転送される。このように本発明では、転送条件情報を設定し、自動コントロール転送の開始を指示するだけで、セットアップステージの開始からステータスステージの完了までの一連の処理が、転送コントローラにより自動的に行われる。従って、ステージ単位でデータ転送の制御、管理を行う手法に比べて、コントロール転送時の処理部の処理負荷を軽減できる。

10

【 0 0 1 1 】

また本発明では、前記転送条件レジスタに、デバイスリクエストのデータと、データステージでの転送データのトータルサイズが、転送条件情報として設定されるようにしてもよい。

【 0 0 1 2 】

このように、デバイスリクエストのデータを転送条件レジスタに設定することで、セットアップステージにおいて、デバイスリクエストのデータを自動転送できるようになる。また、転送データのトータルサイズを転送条件レジスタに設定することで、データステージにおいて、そのトータルサイズのデータを、データパケットを用いて自動転送できるようになる。

20

【 0 0 1 3 】

また本発明では、前記転送条件レジスタに、データステージの有無の情報と、データステージでの転送方向と、マックスパケットサイズが、転送条件情報として設定されるようにしてもよい。

【 0 0 1 4 】

なお、データステージの有無の情報やデータステージでの転送方向などは、デバイスリクエストのデータ（コマンド）をデコードすることで得るようにしてもよい。

【 0 0 1 5 】

また本発明では、前記転送コントローラが、ステータスステージのトランザクションが完了した場合に、処理部に対して割り込みを発生するようにしてもよい。

30

【 0 0 1 6 】

このようにすれば、自動コントロール転送の開始を指示した後、割り込みが発生するまでの期間を、処理部が他の処理に使用できるようになり、処理の効率化を図れる。

【 0 0 1 7 】

また本発明では、前記転送コントローラが、エンドポイントからストールを受信した場合に、自動コントロール転送を中止し、処理部に対して割り込みを発生するようにしてもよい。

【 0 0 1 8 】

このようにすれば、エンドポイントからのストールの受信時に、自動コントロール転送を無効にして、適正な処理を行うことが可能になる。

40

【 0 0 1 9 】

また本発明では、前記転送コントローラが、自動コントロール転送の開始が指示された場合に、セットアップステージのトランザクションの開始を指示する回路と、開始したセットアップステージのトランザクションが完了した場合に、データステージでの転送方向とデータステージの有無の情報に基づいて、データステージのINトランザクションの実行、データステージのOUTトランザクションの実行、或いはデータステージ無しのいずれかを選択する回路と、データステージのINトランザクションの実行が選択された場合には、データステージのINトランザクションの開始を指示し、データステージのOUTトランザクションの実行が選択された場合には、データステージのOUTトランザクション

50

の開始を指示する回路と、開始したデータステージのINトランザクションが完了した場合には、ステータスステージのOUTトランザクションの開始を指示し、開始したデータステージのOUTトランザクションが完了した場合、或いはデータステージ無しが選択された場合には、ステータスステージのINトランザクションの開始を指示する回路とを含むようにしてもよい。

【0020】

このようにすれば、セットアップステージ、データステージ、ステータスステージの一連のトランザクションを、処理部の介在無しに、自動的に発生できるようになる。

【0021】

また本発明では、前記パケットバッファに、各エンドポイントとの間で転送されるデータが各パイプ領域に記憶される複数のパイプ領域が確保され、各パイプ領域と各エンドポイントとの間でのデータ転送の転送条件情報が、複数の転送条件レジスタの各転送条件レジスタに設定され、前記転送コントローラが、コントロール転送のパイプ領域に対応する転送条件レジスタに設定された転送条件情報に基づいて、コントロール転送のトランザクションを自動発生し、コントロール転送のパイプ領域とコントロール転送のエンドポイントとの間で、自動転送コントロール転送を行うようにしてもよい。

10

【0022】

本発明では、パケットバッファに対して複数のパイプ領域（バッファ領域）が確保（allocate）される。この場合に、各パイプ領域は、バスの各エンドポイントに対応して確保される。また、各パイプ領域には、それに対応する各エンドポイントとの間で転送（送信、受信）されるデータが記憶（バッファリング）される。バッファコントローラは、このようなパイプ領域が確保されたパケットバッファ（バッファ）のアクセス制御（領域管理）を行う。

20

【0023】

そして本発明では、各パイプ領域と各エンドポイントとの間でのデータ転送の転送条件情報（エンドポイント情報、パイプ情報）が各転送条件レジスタ（パイプレジスタ）に設定される。この場合にコントロール転送の転送条件情報は、コントロール転送のパイプ領域に割り当てられる転送条件レジスタに設定される。そして、転送コントローラは、この転送条件レジスタに設定された転送条件情報に基づいて、コントロール転送のエンドポイントに対するトランザクションを自動発生し、パイプ領域とエンドポイントとの間で自動コントロール転送を行う。これにより、データ転送制御装置の制御等を行う処理部の処理負荷を軽減できる。

30

【0024】

また本発明では、前記パイプ領域が、コントロール転送のエンドポイントに専用のパイプ領域と、任意のエンドポイントに割り当て可能な汎用のパイプ領域を含み、前記転送コントローラが、前記専用のパイプ領域を用いて、自動コントロール転送を行うようにしてもよい。

【0025】

このように、コントロール転送のエンドポイントに専用のパイプ領域を設けることで、ユーザの混乱等を防止できる。また、任意のエンドポイントに割り当て可能な汎用のパイプ領域を確保することで、パイプ領域とエンドポイントとの対応づけを動的に変化させることが可能になり、データ転送の効率的なスケジューリングが可能になる。

40

【0026】

また本発明では、ホストの役割として動作するホスト動作のステートと、ペリフェラルの役割として動作するペリフェラル動作のステートを含む複数のステートの制御を行うステートコントローラを含み、前記転送コントローラが、ホスト動作時において、ホストとしてのデータ転送を行うホストコントローラと、ペリフェラル動作時において、ペリフェラルとしてのデータ転送を行うペリフェラルコントローラとを含み、前記ホストコントローラが、ホスト動作時に自動コントロール転送を行うようにしてもよい。

【0027】

50

本発明によれば、例えば、ステートコントローラにより制御されるステートが、ホスト動作のステートになると、ホストコントローラにより、ホストの役割としてのデータ転送が行われる。また、ステートコントローラにより制御されるステートが、ペリフェラル動作のステートになると、ペリフェラルコントローラにより、ペリフェラルの役割としてのデータ転送が行われる。これにより、いわゆるデュアルロール・デバイス機能を実現できるようになる。

【0028】

そして本発明では、ホストコントローラが、ホスト動作時に、コントロール転送の一連のステージのトランザクションを自動発生する自動コントロール転送を行う。これにより、デュアルロール・デバイス機能を実現しながら、ホスト動作時における処理部の処理負荷を軽減できる。

【0029】

また本発明では、USB (Universal Serial Bus)のOTG (On - The - Go)規格に準拠したデータ転送を行うようにしてもよい。

【0030】

また本発明は、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、前記データ転送制御装置のデータ転送を制御する処理部とを含む電子機器に関係する。

【0031】

また本発明は、バスを介したデータ転送のためのデータ転送制御方法であって、転送条件レジスタに転送条件情報を設定し、転送条件レジスタに設定される転送条件情報に基づいて、コントロール転送を含むデータ転送を行い、転送データを記憶するケットバッファのアクセス制御を行うと共に、自動コントロール転送の開始が指示された場合に、セットアップステージのトランザクションを自動発生してセットアップステージのケットを自動転送し、次に、転送すべきデータが存在する場合にはデータステージのトランザクションを自動発生してデータステージのケットを自動転送し、次に、ステータスステージのトランザクションを自動発生してステータスステージのケットを自動転送するデータ転送制御方法に関係する。

【0032】

【発明の実施の形態】

以下、本実施形態について説明する。

【0033】

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不当に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。

【0034】

1. OTG

まず、OTG (USB On - The - Go) について簡単に説明する。

【0035】

1.1 Aデバイス、Bデバイス

OTGでは、コネクタの規格として、図1(A)に示すようなMini-Aプラグ、Mini-Bプラグが定義されている。また、これらのMini-Aプラグ、Mini-Bプラグ(広義にはケーブルの第1、第2のプラグ)の両方を接続できるコネクタとして、Mini-ABレセプタクル(receptacle)が定義されている。

【0036】

そして例えば図1(B)のように、USBケーブルのMini-Aプラグに電子機器Pが接続され、Mini-Bプラグに電子機器Qが接続されると、電子機器PはAデバイスに設定され、電子機器QはBデバイスに設定される。一方、図1(C)に示すように、Mini-Bプラグ、Mini-Aプラグが電子機器P、Qに接続されると、電子機器P、Qは、各々、Bデバイス、Aデバイスに設定される。

10

20

30

40

50

【 0 0 3 7 】

Mini-A プラグ内では I D ピンが G N D に接続されており、Mini-B プラグ内では I D ピンはフローティング状態になっている。電子機器は、内蔵するプルアップ抵抗回路を用いて、この I D ピンの電圧レベルを検出することで、自身が Mini-A プラグに接続されたのか、或いは Mini-B プラグに接続されたのかを判断する。

【 0 0 3 8 】

O T G では、A デバイス (マスター) が、電源 (V B U S) を供給する側 (供給元) になり、B デバイス (スレイブ) が、電源の供給を受ける側 (供給先) になる。また、A デバイスは、デフォルトのステートがホストになり、B デバイスは、デフォルトのステートがペリフェラル (周辺機器) になる。

10

【 0 0 3 9 】

1 . 2 デュアルロール・デバイス

O T G では、ホスト (簡易ホスト) としての役割とペリフェラルとしての役割の両方を持つことができるデュアルロール・デバイス (Dual-Role Device) が定義されている。

【 0 0 4 0 】

デュアルロール・デバイスは、ホストにもペリフェラルにもなれる。そして、デュアルロール・デバイスに接続された相手が、従来の U S B 規格におけるホストやペリフェラルである場合には、デュアルロール・デバイスの役割は一意に定まる。つまり、接続相手がホストであれば、デュアルロール・デバイスはペリフェラルになり、接続相手がペリフェラルであれば、デュアルロール・デバイスはホストになる。

20

【 0 0 4 1 】

一方、接続相手がデュアルロール・デバイスである場合には、両方のデュアルロール・デバイスは、お互いにホストとペリフェラルの役割を交換できる。

【 0 0 4 2 】

1 . 3 S R P、H N P

デュアルロール・デバイスは、図 2 (A)、(B) に示すようなセッション開始要求手順 S R P (Session Request Protocol) やホスト交換手順 H N P (Host Negotiation Protocol) の機能を持つ。

【 0 0 4 3 】

ここでセッション開始要求手順 S R P は、B デバイスが A デバイスに対して、V B U S (電源) の供給を要求するプロトコルである。

30

【 0 0 4 4 】

バスを使用しない場合に O T G では、A デバイスは V B U S の供給を停止できる。これにより、A デバイスが例えば小型の携帯機器であった場合に、無駄な電力消費を防止できる。そして、A デバイスが V B U S の供給を停止した後に、B デバイスが V B U S を供給してもらいたい場合には、この S R P を使用して、A デバイスに対して V B U S の供給の再開を要求する。

【 0 0 4 5 】

図 2 (A) に S R P の流れを示す。図 2 (A) に示すように、B デバイスは、データ・ライン・パルシングと V B U S パルシングを行うことで、A デバイスに対して V B U S の供給を要求する。そして、A デバイスによる V B U S の供給の開始後に、B デバイスのペリフェラル動作 (peripheral operation) と、A デバイスのホスト動作 (host operation) が開始する。

40

【 0 0 4 6 】

図 1 (A) ~ 図 1 (C) で説明したように、デュアルロール・デバイス同士の接続では、Mini-A プラグが接続された側である A デバイスがデフォルトのホストとなり、Mini-B プラグが接続された側である B デバイスがデフォルトのペリフェラルになる。そして、O T G では、プラグの抜き差しを行わなくても、ホストとペリフェラルの役割を交換できる。H N P は、このホストとペリフェラルの役割を交換するためのプロトコルである。

【 0 0 4 7 】

50

HNPの流れを図2(B)に示す。デフォルトのホストとして動作するAデバイスが、バスの使用を終了すると、バスがアイドル状態になる。その後、Bデバイスがデータ信号線DP(D+)のプルアップ抵抗を無効にすると、AデバイスがDPのプルアップ抵抗を有効にする。これにより、Aデバイスの役割はホストからペリフェラルに変わり、ペリフェラルとしての動作を開始する。また、Bデバイスの役割はペリフェラルからホストに変わり、ホストとしての動作を開始する。

【0048】

その後、Bデバイスが、バスの使用を終了し、AデバイスがDPのプルアップ抵抗を無効にすると、Bデバイスが、DPのプルアップ抵抗を有効にする。これにより、Bデバイスの役割はホストからペリフェラルに戻り、ペリフェラルとしての動作を再開する。また、Aデバイスの役割はペリフェラルからホストに戻り、ホストとしての動作を再開する。

10

【0049】

以上に説明したOTGによれば、携帯電話やデジタルカメラなどの携帯機器をUSBのホストとして動作させ、携帯機器同士をピア・ツー・ピアで接続してデータ転送を行うことが可能になる。これにより、USBインターフェースに新たな付加価値を生むことができ、これまでには存在しなかったアプリケーションを創出できる。

【0050】

2. OHCI

さて、従来のUSBでは、ホストであるパーソナルコンピュータが有するデータ転送制御装置(ホストコントローラ)は、マイクロソフト社が提唱したOHCI(Open Host Controller Interface)や、UHCI(Universal Host Controller Interface)といった規格に準拠していた。また、使用されるOS(Operating System)も、マイクロソフト社のWindowsやアップル社のマッキントッシュのOSなどに限定されていた。

20

【0051】

しかしながら、OTGのターゲット・アプリケーションである小型携帯機器では、組み込まれるCPUのアーキテクチャや、使用されるOSは千差万別である。更に、パーソナルコンピュータのホストコントローラ向けに規格化されたOHCIやUHCIは、USBホストとしての機能をフルに実装することを前提としており、小型携帯機器への実装に最適であるとは言い難い。

【0052】

例えば、図3(A)に、OHCIで使用されるリスト構造のディスクリプタの一例を示す。

30

【0053】

図3(A)において、エンドポイントディスクリプタED1、ED2、ED3は、リンクポイントによりリンクされており、エンドポイント1、2、3とのコミュニケーションに必要な情報が含まれている。そして、これらのED1、ED2、ED3には、転送ディスクリプタTD11~TD13、TD21、TD31~TD32がリンクポイントにより更にリンクされている。そして、これらの転送ディスクリプタには、エンドポイント1、2、3との間で転送するパケットデータに必要な情報が含まれている。

【0054】

図3(A)のリスト構造のディスクリプタは、図3(B)のCPU610(広義には処理部)上で動作するファームウェア(ホストコントローラ・ドライバ)が作成し、システムメモリ620に書き込む。即ち、ファームウェアは、システム中のエンドポイントに対してエンドポイントディスクリプタを割り当て、エンドポイント情報等に基づいてエンドポイントディスクリプタ、転送ディスクリプタを作成する。そして、これらのディスクリプタをリンクポイントでリンクさせて、システムメモリ620に書き込む。

40

【0055】

データ転送制御装置600(ホストコントローラ)は、システムメモリ620に書き込まれたリスト構造のディスクリプタを読み出し、エンドポイントディスクリプタや転送ディスクリプタに記述される情報に基づいて、データ転送を実行する。

50

【 0 0 5 6 】

具体的には、データ転送制御装置 6 0 0 (ホストコントローラ)は、E D 1に基づいてエンドポイント 1 の情報を設定し、E D 1にリンクされる T D 1 1に基づいて、エンドポイント 1 との間でデータ転送を行う。次に、E D 2に基づいてエンドポイント 2 の情報を設定し、E D 2にリンクされる T D 2 1に基づいて、エンドポイント 2 との間でデータ転送を行う。同様にして、データ転送制御装置 6 0 0 は、T D 3 1、T D 1 2、T D 3 2、T D 1 3に基づいてデータ転送を実行する。

【 0 0 5 7 】

以上のように、O H C I 準拠のデータ転送制御装置(ホストコントローラ)では、C P U 上で動作するファームウェア(ホストコントローラ・ドライバ)が、図 3 (A)に示すよ
10
うな複雑な構造のディスクリプタを作成しなければならない。従って、C P U の処理負荷が非常に重い。

【 0 0 5 8 】

この場合、従来の U S B では、ホストの役割が割り当てられるのはパーソナルコンピュータだけであり、このパーソナルコンピュータは高性能の C P U を有している。従って、図 3 (A)に示すような複雑な構造のディスクリプタの作成も、余裕を持って行うことが可能であった。

【 0 0 5 9 】

ところが、O T G のターゲットアプリケーションである小型携帯機器(デジタルカメラ、携帯電話等)に組み込まれる C P U (embedded CPU)は、パーソナルコンピュータの C P U に比べて、性能が格段に低いのが一般的である。従って、携帯機器に O T G のホスト動作を行わせると、携帯機器に組み込まれる C P U に過大な負荷がかかり、他の処理に支障が生じたり、データ転送のパフォーマンスが低下するなどの問題が生じる。
20

【 0 0 6 0 】

3 . 構成例

図 4 に、以上のような問題を解決できる本実施形態のデータ転送制御装置の構成例を示す。なお、本実施形態のデータ転送制御装置は、図 4 の全ての回路ブロックを含む必要はなく、その一部の回路ブロックを省略してもよい。

【 0 0 6 1 】

トランシーバ 1 0 (以下、適宜 X c v r と呼ぶ)は、差動データ信号 D P、D M を用いて U S B (広義にはバス)のデータを送受信する回路であり、U S B の物理層(P H Y)回路 1 2 を含む。より具体的にはトランシーバ 1 0 は、D P、D M のラインステート(J、K、S E 0 等)の生成、シリアル/パラレル変換、パラレル/シリアル変換、ビットスタ
30
ッピング、ビットアンスタッピング、N R Z I デコード、N R Z I エンコードなどを行う。なお、トランシーバ 1 0 をデータ転送制御装置の外部に設けるようにしてもよい。

【 0 0 6 2 】

O T G コントローラ 2 0 (広義にはステートコントローラ。以下、適宜 O T G C と呼ぶ)は、O T G の S R P 機能や H N P 機能(図 2 (A)、(B)参照)を実現するための種々の処理を行う。即ち、O T G コントローラ 2 0 は、ホストの役割として動作するホスト動作のステートや、ペリフェラルの役割として動作するペリフェラル動作のステートなど
40
を含む複数のステートの制御を行う。

【 0 0 6 3 】

より具体的には、O T G 規格には、デュアルロール・デバイスの A デバイス時(図 1 (B)、(C)参照)のステート遷移や B デバイス時のステート遷移が定義されている。O T G コントローラ 2 0 は、これらのステート遷移を実現するためのステートマシ
50
ンを含む。また、O T G コントローラ 2 0 は、U S B のデータラインステートや、V B U S レベルや、I D ピンのステートを検出(監視)する回路を含む。そして、O T G コントローラ 2 0 が含むステートマシ
ンは、これらの検出情報に基づいて、そのステート(例えば、ホスト、ペリフェラル、サスペンド又はアイドルなどのステート)を変化させる。この場合のステートの遷移は、ハードウェア回路により実現してもよいし、ファームウェアがステ

ートコマンドをレジスタに設定することで実現してもよい。そして、ステートが遷移すると、OTGコントローラ20は、遷移後のステートに基づいて、VBUSを制御したり、DP、DMのプルアップ抵抗/プルダウン抵抗の接続/非接続を制御する。また、ホストコントローラ50(以下、適宜HCと呼ぶ)、ペリフェラルコントローラ60(以下、適宜PCと呼ぶ)のイネーブル/ディスエーブルを制御する。

【0064】

HC/PC切り替え回路30(HC/PC・コモン回路)は、トランシーバ10と、ホストコントローラ50又はペリフェラルコントローラ60との間の接続の切り替え制御を行う。また、USBのデータ(DP、DM)のラインステートの生成の指示をトランシーバ10に対して行う。なお、接続の切り替え制御は、HC/PCセクタ32により実現され、ラインステートの生成指示は、ラインステートコントローラ34により実現される。

10

【0065】

例えばOTGコントローラ20が、ホスト動作時(ホストステート時)にHCイネーブル信号をアクティブにすると、HC/PC切り替え回路30(HC/PCセクタ32)は、トランシーバ10とホストコントローラ50を接続する。一方、OTGコントローラ20が、ペリフェラル動作時(ペリフェラルステート時)にPCイネーブル信号をアクティブにすると、HC/PC切り替え回路30は、トランシーバ10とペリフェラルコントローラ60を接続する。このようにすることで、ホストコントローラ50とペリフェラルコントローラ60とを排他的に動作させることが可能になる。

【0066】

転送コントローラ40は、USB(広義にはバス)を介したデータ転送を制御する回路であり、ホストコントローラ50(HC)とペリフェラルコントローラ60(PC)を含む。

20

【0067】

ここでホストコントローラ50は、ホスト動作時(OTGコントローラ20からのHCイネーブル信号のアクティブ時)に、ホストの役割としてのデータ転送制御を行う回路である。

【0068】

即ち、ホストコントローラ50は、ホスト動作時に、HC/PC切り替え回路30によりトランシーバ10に接続される。そしてホストコントローラ50は、レジスタ部70の転送条件レジスタ部72に設定された転送条件情報に基づいて、エンドポイントに対するトランザクションを自動発生する。そして、パケットバッファ100に確保(allocate)されたパイプ領域(PIPE0~PIPEe。以下、適宜PIPEと呼ぶ)と、そのパイプ領域に対応するエンドポイントとの間で、データ(パケット)の自動転送(処理部が介在しないハードウェア回路によるデータ転送)を行う。

30

【0069】

より具体的にはホストコントローラ50は、複数のパイプ転送間の調停、フレームにおける時間管理、転送のスケジューリング、再送の管理などを行う。また、パイプ転送の転送条件情報(オペレーション情報)をレジスタ部70を介して管理する。また、トランザクションの管理を行ったり、パケットを生成/分解したり、サスペンド/レジューム/リセット状態生成の指示を行う。

40

【0070】

そして本実施形態ではホストコントローラ50が自動コントロール転送を行う。即ち、コントロール転送時に、セットアップステージ、データステージ(データステージ有りの場合)、ステータスステージの各ステージのトランザクションの自動発生(処理部が介在しないハードウェア回路によるトランザクションの自動発生)を行い、これらのトランザクションに必要なパケットを自動転送する。

【0071】

一方、ペリフェラルコントローラ60は、ペリフェラル動作時(OTGコントローラ20からのPCイネーブル信号のアクティブ時)に、ペリフェラルの役割としてのデータ転送

50

制御を行う回路である。

【 0 0 7 2 】

即ち、ペリフェラルコントローラ 6 0 は、ペリフェラル動作時に、H C / P C 切り替え回路 3 0 によりトランシーバ 1 0 に接続される。そして、レジスタ部 7 0 の転送条件レジスタ部 7 2 に設定された転送条件情報に基づいて、パケットバッファ 1 0 0 に確保されたエンドポイント領域 (E P 0 ~ E P e 。以下適宜 E P と呼ぶ) とホストとの間でデータを転送する。

【 0 0 7 3 】

より具体的には、ペリフェラルコントローラ 6 0 は、エンドポイント転送の転送条件情報 (オペレーション情報) をレジスタ部 7 0 を介して管理する。また、トランザクションの管理を行ったり、パケットを生成 / 分解したり、リモート・ウェイクアップ信号生成の指示を行う。

10

【 0 0 7 4 】

なお、エンドポイントは、ユニークなアドレスを割り当てることができる、ペリフェラル (デバイス) 上のポイント (部分) である。ホストとペリフェラル (デバイス) との間でのデータ転送は、全て、このエンドポイントを経由して行われる。また、トランザクションは、トークンパケットと、オプションなデータパケットと、オプションなハンドシェイクパケットにより構成される。

【 0 0 7 5 】

レジスタ部 7 0 は、データ転送 (パイプ転送、エンドポイント転送) 制御、バッファアクセス制御、バッファ管理、割り込み制御、ブロック制御、或いは D M A 制御などを行うための種々のレジスタを含む。なお、レジスタ部 7 0 が含むレジスタは、R A M などのメモリにより実現してもよいし、D フリップフロップなどにより実現してもよい。また、レジスタ部 7 0 のレジスタは、1 つにまとめずに、各ブロック (H C 、 P C 、 O T G C 、 X c v r 等) に分散して配置してもよい。

20

【 0 0 7 6 】

レジスタ部 7 0 は、転送条件レジスタ部 7 2 を含む。そして、この転送条件レジスタ部 7 2 は、ホスト動作時にパケットバッファ 1 0 0 に確保されるパイプ領域 (P I P E 0 ~ P I P E e) とエンドポイントとの間でのデータ転送の転送条件情報 (転送制御情報) を記憶するレジスタを含む。これらの各転送条件レジスタは、パケットバッファ 1 0 0 の各パイプ領域に対応して設けられる。

30

【 0 0 7 7 】

なお、ペリフェラル動作時には、パケットバッファ 1 0 0 にはエンドポイント領域 (E P 0 ~ E P e) が確保される。そして、転送条件レジスタ部 7 2 に設定された転送条件情報に基づいて、データ転送制御装置とホストとの間でのデータ転送が行われる。

【 0 0 7 8 】

バッファコントローラ 8 0 (F I F O マネージャ) は、パケットバッファ 1 0 0 に対するアクセス (リード / ライト) 制御や領域管理を行う。より具体的には、C P U (広義には処理部) 、D M A (Direct Memory Access) 、U S B によるパケットバッファ 1 0 0 へのアクセス・アドレスを生成・管理する。また、C P U 、D M A 、U S B によるパケットバッファ 1 0 0 へのアクセスの調停を行う。

40

【 0 0 7 9 】

例えば、ホスト動作時には、バッファコントローラ 8 0 は、インターフェース回路 1 1 0 (C P U 又は D M A) とパケットバッファ 1 0 0 の間のデータ転送経路と、パケットバッファ 1 0 0 とホストコントローラ 5 0 (U S B) の間のデータ転送経路を設定 (確立) する。

【 0 0 8 0 】

一方、ペリフェラル動作時には、バッファコントローラ 8 0 は、インターフェース回路 1 1 0 (C P U 又は D M A) とパケットバッファ 1 0 0 の間のデータ転送経路と、パケットバッファ 1 0 0 とペリフェラルコントローラ 6 0 (U S B) の間のデータ転送経路を設定

50

する。

【 0 0 8 1 】

パケットバッファ 1 0 0 (F I F O、パケットメモリ、バッファ)は、U S Bを介して転送されるデータ(送信データ又は受信データ)を一時的に記憶(バッファリング)するものである。このパケットバッファ 1 0 0は、例えばR A M (Random Access Memory)などにより構成できる。なお、パケットバッファ 1 0 0をデータ転送制御装置の外部に設けてもよい(外付けメモリにしてもよい)。

【 0 0 8 2 】

ホスト動作時には、パケットバッファ 1 0 0はパイプ転送用のF I F O (First-In First-Out)として使用される。即ち、パケットバッファ 1 0 0には、U S B (バス)の各エンドポイントに対応するように、パイプ領域P I P E 0 ~ P I P E e (広義にはバッファ領域)が確保される。また、各パイプ領域P I P E 0 ~ P I P E eには、それに対応する各エンドポイントとの間で転送されるデータ(送信データ又は受信データ)が記憶される。

10

【 0 0 8 3 】

一方、ペリフェラル動作時には、パケットバッファ 1 0 0はエンドポイント転送用のF I F Oとして使用される。即ち、パケットバッファ 1 0 0には、エンドポイント領域E P 0 ~ E P e (広義にはバッファ領域)が確保される。また、各エンドポイント領域E P 0 ~ E P eには、ホストとの間で転送されるデータ(送信データ又は受信データ)が記憶される。

【 0 0 8 4 】

なお、パケットバッファ 1 0 0に確保されるバッファ領域(ホスト動作時にパイプ領域に設定され、ペリフェラル動作時にエンドポイント領域に設定される領域)は、先に入力された情報が先に出力されるような記憶領域(F I F O領域)に設定されている。

20

【 0 0 8 5 】

また、P I P E 0は、コントロール転送用のエンドポイント0に専用のパイプ領域であり、P I P E a ~ P I P E eは、任意のエンドポイントに割り当て可能な汎用のパイプ領域である。

【 0 0 8 6 】

即ち、U S Bでは、エンドポイント0がコントロール転送に専用のエンドポイントに設定される。従って、本実施形態のようにP I P E 0をコントロール転送に専用のパイプ領域にすることで、ユーザが混乱するのを防止できる。また、P I P E a ~ P I P E eを、任意のエンドポイントに割り当て可能なパイプ領域にすることで、エンドポイントに対応するパイプ領域を動的に変化させることが可能になる。これにより、パイプ転送のスケジューリングの自由度を向上でき、データ転送の効率化を図れる。

30

【 0 0 8 7 】

なお本実施形態では、バッファ領域(パイプ領域又はエンドポイント領域)は、マックスパケットサイズMaxPktSize(広義にはページサイズ)とページ数BufferPageにより、その領域サイズRSizeが設定される($RSize=MaxPktSize \times BufferPage$)。このようにすることで、バッファ領域の領域サイズや面数(ページ数)を任意に設定できるようになり、パケットバッファ 1 0 0のリソースの有効利用を図れる。

40

【 0 0 8 8 】

インターフェース回路 1 1 0は、U S Bとは異なる他のバスであるD M A (システムメモリ)バスやC P Uバスと、パケットバッファ 1 0 0との間でのデータ転送を行うための回路である。このインターフェース回路 1 1 0は、パケットバッファ 1 0 0と外部のシステムメモリとの間で、D M A転送を行うためのD M Aハンドラ回路 1 1 2を含む。また、パケットバッファ 1 0 0と外部のC P Uとの間で、P I O (Parallel I/O)転送を行うためのC P Uインターフェース回路 1 1 4を含む。なお、C P U (処理部)をデータ転送制御装置に内蔵させてもよい。

【 0 0 8 9 】

クロックコントローラ 1 2 0は、内蔵P L L又は外部入力クロックに基づいて、データ転

50

送制御装置の内部で使用する各種のクロックを生成する。

【0090】

4. パイプ領域

本実施形態では図5(A)に示すように、ホスト動作時に、パケットバッファ100にパイプ領域PIPE0~PIPEeが確保(allocate)される。そして、この各パイプ領域PIPE0~PIPEeとペリフェラルの各エンドポイントの間で、データが転送される。

【0091】

ここで、本実施形態のパイプ領域の「パイプ」は、USBで定義される「パイプ」(デバイス上のエンドポイントとホスト上のソフトウェアとの間の関連を表す論理的な抽象化、論理的な経路)とは若干意味合いが異なる。

10

【0092】

本実施形態のパイプ領域は、図5(A)に示すように、USB(バス)に接続されるペリフェラルが有する各エンドポイントに対応して、パケットバッファ100上に確保される。例えば図5(A)において、パイプ領域PIPEaはペリフェラル1のエンドポイント1(バルクIN)に対応し、PIPEbはペリフェラル1のエンドポイント2(バルクOUT)に対応する。また、PIPEcはペリフェラル2のエンドポイント1(バルクIN)に対応し、PIPEDはペリフェラル2のエンドポイント2(バルクOUT)に対応する。また、PIPEeはペリフェラル3のエンドポイント1(インタラプトIN)に対応する。なお、PIPE0は、コントロール転送のエンドポイント0に専用のパイプ領域である。

20

【0093】

そして図5(A)の例では、パイプ領域PIPEaとペリフェラル1のエンドポイント1との間でUSBのバルクIN転送が行われ、PIPEbとペリフェラル1のエンドポイント2の間ではバルクOUT転送が行われる。また、PIPEcとペリフェラル2のエンドポイント1の間ではバルクIN転送が行われ、PIPEDとペリフェラル2のエンドポイント2の間ではバルクOUT転送が行われる。また、PIPEeとペリフェラル3のエンドポイント1の間ではインタラプトIN転送が行われる。

【0094】

このように本実施形態では、パイプ領域(汎用)とそれに対応するエンドポイントの間では、任意のデータ転送(アイソクロナス転送、バルク転送、インタラプト転送)を行うことができる。

30

【0095】

そして本実施形態では、パイプ領域とそれに対応するエンドポイントの間では、所与のデータ単位(トータルサイズで指定されるデータ単位)のデータが転送される。この場合のデータ単位としては、例えばIRP(I/Oリクエストパケット)により転送要求されるデータ単位又はそれを適当なサイズに分割したデータ単位などを考えることができる。エンドポイントに対するこのデータ単位のデータ転送(一連のトランザクション)を、本実施形態における「パイプ」と呼ぶことができる。そして、そのような「パイプ」のデータ(送信データ、受信データ)を記憶する領域がパイプ領域になる。

40

【0096】

パイプ領域を用いた所与のデータ単位の転送が終了すると、そのパイプ領域は解放することができる。そして、解放されたパイプ領域は、任意のエンドポイントに割り当てることができる。このように本実施形態では、パイプ領域とエンドポイントとの対応づけを、動的に変化させることができる。

【0097】

また本実施形態では図5(B)に示すように、ペリフェラル動作時には、パケットバッファ100にエンドポイント領域EP0~EPeが確保(設定)される。そして、この各エンドポイント領域EP0~EPeとホスト(ホストコントローラ、システムメモリ)との間で、データが転送される。

50

【 0 0 9 8 】

このように本実施形態では、パケットバッファ 1 0 0 のバッファ領域を、ホスト動作時にはパイプ領域に割り当て、ペリフェラル動作時にはエンドポイント領域に割り当てている。これにより、ホスト動作時とペリフェラル動作時とでパケットバッファ 1 0 0 のリソースを共用（兼用）することが可能になり、パケットバッファ 1 0 0 の使用記憶容量を節約できる。

【 0 0 9 9 】

なおパイプ領域、エンドポイント領域の個数は 6 個に限定されず任意である。

【 0 1 0 0 】

5 . 自動コントロール転送

さて、U S B では、アイソクロナス転送、バルク転送、インタラプト転送、コントロール転送という 4 種類の転送方式が定義されている。この中でコントロール転送は、クライアント・ソフトウェアとファンクションとの間でのコンフィギュレーション / コマンド / ステータスに関するコミュニケーションを、サポートする転送方式である。

【 0 1 0 1 】

このコントロール転送は、ホストがペリフェラル（デバイス）にデバイスリクエストを送るセットアップステージと、デバイスリクエストで指定される転送方向でデータを転送するデータステージと、データ転送が成功したか否かを伝えるステータスステージとにより構成される。また、各ステージは少なくとも 1 以上のトランザクションにより構成される。ここでトランザクションは、トークンパケット、オプションなデータパケット、オプションなハンドシェイクパケットにより構成される、エンドポイントに対するサービスである。このコントロール転送により、ペリフェラル（デバイス、エンドポイント）のエnumレーション（enumeration）、コンフィギュレーション（configuration）などが行われる。

【 0 1 0 2 】

さて、以上のようなコントロール転送を実現する 1 つの手法として、C P U 上で動作するファームウェアがコントロール転送の全てを制御、管理する手法が考えられる。即ち、ファームウェアが、コントロール転送を構成するセットアップステージ、データステージ、ステータスステージの各ステージ（トランザクション）単位で、データ転送を制御、管理する。

【 0 1 0 3 】

しかしながら、コントロール転送は、少なくともホストによるペリフェラル（デバイス）のエnumレーション時には必ず発生し、複数回のコントロール転送（デバイスリクエスト）が行われる場合が多い。しかも、コントロール転送はセットアップステージ、データステージ、ステータスステージという複数のシーケンシャルなステージにより構成される。従って、各ステージが行われる毎にファームウェアがデータ転送を制御、管理する上記手法では、ファームウェア（処理部）の処理負荷が重くなってしまうという課題がある。

【 0 1 0 4 】

このような課題を解決するために、本実施形態では、コントロール転送をハードウェア回路により自動的に行う手法を採用している。この自動コントロール転送手法について、以下、説明する。

【 0 1 0 5 】

本実施形態では、まず、ファームウェアが図 4 の転送条件レジスタ部 7 2 の転送条件レジスタ（P I P E 0 用レジスタ）に、コントロール転送のための転送条件情報を設定する。具体的にはファームウェアは、デバイスリクエストのデータや、データステージでの転送データのトータルサイズを設定する。更に望ましくは、データステージの有無の情報や、データステージでの転送方向や、マックスパケットサイズなどを設定する。

【 0 1 0 6 】

即ち図 6 に示すように、U S B では、デバイスリクエストの種別により、データステージの有無、データステージでの転送方向、ステータスステージでの転送方向は一意的に決ま

10

20

30

40

50

っており、ファームウェア（ホストのソフトウェア）もそれを認識できる。また、コントロール転送のステージはシーケンシャルな構成となっている。本実施形態では、この2つの点に着目し、ファームウェアによる1回の設定（指示）だけで、ホストコントローラ50が1つのコントロール転送を完結できるようにした。即ち、ファームウェアは、デバイスリクエストのデータ（コマンド）、データステージの有無、データステージがある場合にはその転送方向、マックスパケットサイズ（データペイロードサイズ）、データステージでの転送データのトータルサイズを、転送条件レジスタ（ホストコントローラ50内に設けてもよい）に設定し、自動コントロール転送の開始（モード）を指示する。すると、この設定により、ホストコントローラ50（広義には転送コントローラ、データ転送制御装置又はハードウェア回路。以下の説明でも同様）が、図7に示すようなシーケンス動作を行う。

10

【0107】

即ち、自動コントロール転送の開始が指示されると、ホストコントローラ50が、セットアップステージのトランザクション（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）を発生する。このセットアップステージでは、まず、ホストコントローラ50が、SETUPトークンパケットを生成して、ペリフェラルに送信（発行）する。次に、図6に例示されるデバイスリクエストのデータパケットをペリフェラルに送信する。そして、ペリフェラルからACK（ACKnowledgement）のハンドシェイクパケットが返ってくると、ホストコントローラ50は、セットアップステージを終了する。

【0108】

なお、図7の「H P」は「ホストからペリフェラルにパケットが転送されること」を表し、「P H」は「ペリフェラルからホストにパケットが転送されること」を表す。

20

【0109】

セットアップステージが終了すると、ホストコントローラ50は、データステージのトランザクションを自動発生する。なお、「データステージ無し」が設定されていた場合には、ステータスステージのトランザクションを自動発生する。

【0110】

データステージにおいて、「転送方向IN」が設定された場合には、ホストコントローラ50は、INトランザクションを発生する。このINトランザクションでは、ホストコントローラ50がINTトークンパケットをペリフェラルに送信する。そして、データパケットをペリフェラルから受信すると、ACKのハンドシェイクパケットをペリフェラルに送信する。そして、このINトランザクションを繰り返して、ファームウェアにより設定されたトータルサイズ分のデータの受信が完了すると、データステージを終了する。

30

【0111】

一方、データステージにおいて、「転送方向OUT」が設定された場合には、ホストコントローラ50は、OUTトランザクションを発生する。このOUTトランザクションでは、ホストコントローラ50が、OUTトークンパケットをペリフェラルに送信し、データパケットをペリフェラルに送信する。そして、ACKのハンドシェイクパケットをペリフェラルから受信する。そして、このOUTトランザクションを繰り返して、トータルサイズ分のデータをペリフェラルに送信すると、データステージを終了する。

40

【0112】

データステージが終了した場合又はファームウェアにより「データステージ無し」が設定された場合、ホストコントローラ50が、ステータスステージを発生する。そして、データステージがINトランザクションであった場合には、ホストコントローラ50は、OUTトークンパケットを送信した後、零長のデータパケットをペリフェラルに送信する。そして、ACKのハンドシェイクパケットをペリフェラルから受信すると、ホストコントローラ50は、ステータスステージを終了する。

【0113】

一方、データステージがOUTトランザクションであった場合には、ホストコントローラ50は、INTトークンパケットをペリフェラルに送信する。そして、零長のデータパケッ

50

トをペリフェラルから受信すると、ACKのハンドシェイクパケットをペリフェラルに送信し、ステータスステージを終了する。

【0114】

図8(A)、(B)、(C)に、自動コントロール転送時の信号波形例を示す。

【0115】

図8(A)は、ホスト(ホストコントローラ)からペリフェラルにデータを送信する時(書き込み転送時)の信号波形例である。

【0116】

ファームウェアがコントロール転送用のPIPE0の自動転送の開始指示を行うと、図8(A)のC1に示すように、Pipe0TranGo(Pipe0についてのファームウェアからの転送要求信号)がアクティブになる。これにより、ホストコントローラ50による自動コントロール転送が開始する。そして、C2、C3、C4に示すように、セットアップステージのSETUPトランザクション、データステージのOUTトランザクション、ステータスステージのINTトランザクションが発生し、各トランザクションのパケット転送が行われる。そして、ステータスステージが完了すると、C5に示すようにCTLTranComp(ファームウェアへのコントロール転送終了通知信号)がアクティブになり、このCTLTranCompによる割り込みにより、コントロール転送の全ステージの完了がファームウェアに伝えられる。

10

【0117】

このように本実施形態では、ステータスステージのトランザクションが完了すると、ファームウェア(処理部)に対して割り込みが発生する。そして、ファームウェアは、割り込みの要因(ステータス)を調べることで、コントロール転送が完了したことを知ることができる。

20

【0118】

従って本実施形態によれば、ファームウェアは、転送条件情報(デバイスリクエストデータ、トータルサイズ、マックスパケットサイズ等)を設定した後、割り込み(CTLTranComp)が発生するまでは、自動コントロール転送に関わらなくて済むようになる。これにより、ファームウェアの処理負荷を格段に軽減できる。

【0119】

なお、本実施形態では、不正リクエストの送信などの理由で、ペリフェラル(エンドポイント)からストール(STALL)を受信した場合に、自動コントロール転送をその時点で中断し、ファームウェアに対して割り込みを発生するようにしている。

30

【0120】

即ち、ストールは、ペリフェラル(デバイス)がデータを送受信できなかった場合や、コントロール転送のリクエストをペリフェラルがサポートしていない場合等に、ペリフェラルからホストに転送される。このような場合に本実施形態によれば、ファームウェアは、ストール返信を割り込みにより知ることができ、コントロール転送を無効にすることが可能になる。

【0121】

図8(B)は、ホストがペリフェラルからデータを受信する時(読み込み転送時)の信号波形例である。この場合には、C6に示すように、データステージがINTトランザクションになる。また、C7に示すように、ステータスステージがOUTトランザクションになる。

40

【0122】

図8(C)は、「データステージ無し」が設定された時の信号波形例である。この場合には、C8に示すように、データステージが省略される。また、C9に示すように、ステータスステージがINTトランザクションになる。

【0123】

図9(A)は、セットアップステージの詳細な信号波形例である。D1に示すように、PipeTranGo(ホストコントローラ50内のHCシーケンス管理回路からの転送要求信号)が

50

アクティブになると、トークンパケット、デバイスリクエスト（リクエストコマンド）のデータパケットをホストがペリフェラルに送信する。このデバイスリクエストのデータは、ファームウェアにより転送条件レジスタに設定される。そして、ペリフェラルからハンドシェイクパケット（ACK）を受信すると、D 2 に示すようにTranCmpACKがアクティブになる。これによりセットアップステージが終了する。

【 0 1 2 4 】

図 9（B）は、データ送信時におけるデータステージの詳細な信号波形例である。D 3 に示すようにPipeTranGoがアクティブになると、トークンパケット、データパケットをホストがペリフェラルに送信する。そして、ペリフェラルからハンドシェイクパケット（ACK）を受信すると、D 4 に示すようにTranCmpACKがアクティブになる。

10

【 0 1 2 5 】

そしてD 5 では、NextTotalSize（残りデータサイズ = 転送データのトータルサイズ - 転送済みサイズ）が零になっていない。従って、D 6 に示すように、PipeTranGoがアクティブになり、トークンパケット、データパケットをホストがペリフェラルに送信する。そしてペリフェラルからのハンドシェイクパケット（ACK）の受信後、D 7 に示すようにNextTotalSizeが零になっていると、データステージが完了する。

【 0 1 2 6 】

図 9（C）は、データ受信時におけるデータステージの詳細な信号波形例である。この場合には図 9（B）と異なり、D 8、D 9 に示すようにデータパケットをホストがペリフェラルから受信し、D 10、D 11 に示すようにホストがペリフェラルにハンドシェイクパケット（ACK）を送信する。

20

【 0 1 2 7 】

図 10（A）は、データ送信時又はデータステージ無しの場合におけるステータスステージの詳細な信号波形例である。E 1 に示すようにPipeTranGoがアクティブになると、トークンパケットをホストがペリフェラルに送信し、ペリフェラルから零長のデータパケット（空パケット）を受信する。そして、ホストがペリフェラルにハンドシェイクパケット（ACK）を送信し、E 2 に示すようにTranCmpACKがアクティブになる。これによりステータスステージが完了する。

【 0 1 2 8 】

図 10（B）は、データ受信時におけるステータスステージの詳細な信号波形例である。この場合には図 10（A）と異なり、E 3、E 4 に示すようにホストがペリフェラルにトークンパケット、零長のデータパケットを送信し、E 5 に示すようにハンドシェイクパケット（ACK）をホストがペリフェラルから受信する。

30

【 0 1 2 9 】

6 . ファームウェアの処理

次に、ファームウェア（処理部）の詳細な処理例について説明する。

【 0 1 3 0 】

図 11 は、自動コントロール転送モード（EnAutoMode）をオンにした場合のファームウェアの処理例を示すフローチャートである。

【 0 1 3 1 】

まず、ファームウェアは、エンドポイント（転送条件）情報を転送条件レジスタにセットする（ステップ S 5 1）。即ち、エンドポイント番号 [EPNumber]、ファンクションアドレス [FuncAddr]、マックスパケットサイズ [MaxPktSize]、デバイスリクエストデータ [DeviceReqData]などをセットする。そして、HC のスケジューリング [連続実行回数 : Continuity] を指定する（ステップ S 5 2）。

40

【 0 1 3 2 】

次に、データステージがあるか否かを判断する（ステップ S 5 3）。そして、データステージが無い場合には、データステージ無し [NoDataStage=TRUE] を設定し（ステップ S 5 4）、ステップ S 6 4 に移行する。

【 0 1 3 3 】

50

一方、データステージがある場合には、パイプ（バッファ）領域を既に確保しているか否かを確認し（ステップ S 5 5）、確保している場合にはパイプ領域のデータクリア指示 [FIFOClr] を行う（ステップ S 5 6）。

【 0 1 3 4 】

次に、データステージ有り [NoDataStage=FALSE) を設定する（ステップ S 5 7）。そして、データステージの転送方向 [DataStageDir] を設定し、転送データのトータルサイズ [TotalSize] をセットする（ステップ S 5 8、S 5 9）。

【 0 1 3 5 】

なお、ステップ S 5 1 ~ S 5 9 の設定順序は任意である。

【 0 1 3 6 】

次に、パイプ（バッファ）領域のページ数 [BufferPage] をセットし、パイプ領域確保の指示 [SetAllocation] を行う（ステップ S 6 0、S 6 1）。そして、DMA を用いるか否かを判断し、用いる場合には DMA バスの接続指示 [JoinDMA] を行う（ステップ S 6 2、S 6 3）。

【 0 1 3 7 】

次に、自動トランザクションの開始指示 [TranGo] を行う（ステップ S 6 4）。そして、割り込みが発生するのを待ち（ステップ S 6 5）、正常完了、STALL 返答、ハンドシェイク待ちのタイムアウト処理などを行う（ステップ S 6 6）。

【 0 1 3 8 】

本実施形態では、ファームウェアが図 1 1 のステップ S 5 1 ~ S 6 4 の設定を行うと、ホストコントローラ 5 0（転送コントローラ）が、自動的にコントロール転送のセットアップステージ、データステージ、ステータスステージを順次発生する。そして、最後のステータスステージが終了するとホストコントローラ 5 0 が割り込みを発生し、これによりファームウェアがステップ S 6 6 の処理を行う。

【 0 1 3 9 】

従って、DMA 転送を利用した場合には、ファームウェアは、図 1 1 のステップ S 5 1 ~ S 6 4 の設定を行った後、割り込みが発生するまでの期間を、他の処理に使用できるようになる。これにより、ファームウェアの処理の効率化を図れる。

【 0 1 4 0 】

例えば図 1 2 に、自動コントロール転送モード（EnAutoMode）をオフにした場合のファームウェアの処理のフローチャートを示す。自動コントロール転送モードをオフにした場合には、ファームウェアは、セットアップステージ、データステージ、ステータスステージの全てのステージで、図 1 2 のステップ S 7 1 ~ S 8 3 の処理を行う必要がある。従って、1 つのコントロール転送において、図 1 2 のステップ S 7 1 ~ S 8 3 の処理を 3 回分（データステージが無い場合は 2 回分）行わなければならない。このため、ファームウェアの処理負荷が非常に重くなるという問題が生じる。

【 0 1 4 1 】

本実施形態によれば、自動コントロール転送モード（EnAutoMode）をオンにすることで、このような問題を解消できる。

【 0 1 4 2 】

なお、本実施形態では、データステージの有無の情報、データステージでの転送方向等を、ファームウェアが転送条件レジスタに設定しているが、デバイスリクエストのコマンドを、デコーダ回路によりデコードすることで、これらの情報を得るようにしてもよい。この場合には、図 1 1 のステップ S 5 4、S 5 7、S 5 8 の設定（データステージ有無、転送方向の設定）等は不要になる。

【 0 1 4 3 】

7 . 回路例

図 1 3 に、自動コントロール転送を実現する回路の例を示す。この回路は、図 1 8（A）に示すホストコントローラ 5 0 のターゲットパイプ管理回路 5 4 内に設けられる。

【 0 1 4 4 】

10

20

30

40

50

パイプ情報管理回路240は、ターゲットとなるパイプの情報を管理する回路である。このパイプ情報管理回路240は、図18(A)のHCシーケンス管理回路52からターゲットとなるパイプが指示された場合に、レジスタ部70の転送条件レジスタから、対応するパイプの情報を選択して、トランザクション管理回路56やパケットハンドラ回路58に渡す処理などを行う。

【0145】

自動コントロール転送シーケンサ200は、図7で説明した自動コントロール転送のシーケンスを管理する回路である。この自動コントロール転送シーケンサ200は、パイプ情報をトランザクション管理回路56やパケットハンドラ回路58に渡す前に、このパイプ情報に、自動コントロール転送用の情報(自動コントロール転送の各ステージで必要になる情報)を付加する処理などを行う。

10

【0146】

自動コントロール転送シーケンサ200は、自動コントロール転送判断回路202と、セットアップステージモニタ回路204と、データステージ選択回路210と、データステージモニタ回路220と、ステータスステージモニタ回路230を含む。なお、自動コントロール転送シーケンサ200はこれらの全ての回路ブロックを含む必要はなく、その一部を省略してもよい。

【0147】

自動コントロール転送判断回路202は、自動コントロール転送モード(EnAutoMode)の信号とターゲットパイプ番号を受け、そのパイプについて自動コントロール転送を行うか否かを判断する。例えば、自動コントロール転送モード信号がアクティブ(オート)になり、ターゲットパイプがコントロール転送用のPIPE0であった場合には、自動コントロール転送を行うと判断し、自動コントロール転送イネーブル信号をアクティブにする。これにより自動コントロール転送の開始が指示される。

20

【0148】

セットアップステージモニタ回路204は、自動コントロール転送の開始が指示された場合に、セットアップステージのトランザクションの開始を指示する回路である。

【0149】

即ち、セットアップステージモニタ回路204は、自動コントロール転送イネーブル信号がアクティブになると、セットアップステージイネーブル信号をアクティブにして、そのパイプ(PIPE0)についてのセットアップステージのトランザクションの開始を指示する。そして、セットアップステージが完了すると、セットアップステージ完了信号をアクティブにする。

30

【0150】

データステージ選択回路210は、データステージでの転送方向(DataStageDir)とデータステージの有無の情報(NoDataStage)に基づいて、データステージのINトランザクションの実行、データステージのOUTトランザクションの実行、或いは「データステージ無し」のいずれかを選択する回路である。

【0151】

即ちデータステージ選択回路210は、転送方向が「IN」であり、「データステージ有り」の場合には、INデータ・スタートトリガをアクティブにして、INトランザクションの実行を選択する。また、転送方向が「OUT」であり、「データステージ有り」の場合には、OUTデータ・スタートトリガをアクティブにして、OUTトランザクションの実行を選択する。一方、「データステージ無し」の場合は、ステータス・スタートトリガをアクティブにして、データステージをバイパスする。

40

【0152】

データステージモニタ回路220は、データステージのINトランザクションが選択された場合には、INトランザクションの開始を指示し、データステージのOUTトランザクションが選択された場合には、OUTトランザクションの開始を指示する回路である。このデータステージモニタ回路220は、INデータステージモニタ回路222とOUTデ

50

ータステージモニタ回路 224 を含む。

【0153】

ここで、INデータステージモニタ回路 222 は、INデータ・スタートトリガがアクティブになると、INデータステージイネーブル信号をアクティブにして、INトランザクションの開始を指示する。そして、データステージが完了すると、OUTステータス・スタートトリガをアクティブにする。

【0154】

一方、OUTデータステージモニタ回路 224 は、OUTデータ・スタートトリガがアクティブになると、OUTデータステージイネーブル信号をアクティブにして、OUTトランザクションの開始を指示する。そして、データステージが完了すると、INステータス・スタートトリガをアクティブにする。

10

【0155】

ステータスステージモニタ回路 230 は、データステージのINトランザクションが完了した場合には、ステータスステージのOUTトランザクションの開始を指示し、データステージのOUTトランザクションが完了した場合、或いは「データステージ無し」が選択された場合には、ステータスステージのINトランザクションの開始を指示する回路である。このステータスステージモニタ回路 230 は、OUTステータスステージモニタ回路 232 とINステータスステージモニタ回路 234 を含む。

【0156】

ここで、OUTステータスステージモニタ回路 232 は、OUTステータス・スタートトリガがアクティブになると、OUTステータスステージイネーブル信号をアクティブにして、ステータスステージのOUTトランザクションの開始を指示する。

20

【0157】

一方、INステータスステージモニタ回路 234 は、INステータス・スタートトリガがアクティブになると、INステータスステージイネーブル信号をアクティブにして、ステータスステージのINトランザクションの開始を指示する。

【0158】

図 13 に示すような回路構成を採用することで、セットアップステージ、データステージ、ステータスステージの一連のトランザクションを、ファームウェアの介在無しに、自動的に発生できるようになる。

30

【0159】

8．転送条件レジスタ（共用レジスタ）

本実施形態では図 14 に示すように、ホスト動作時には、パイプ領域 PIPE0 ~ PIPEe とエンドポイントとの間で行われるデータ転送の転送条件情報（転送方向、転送種別、マックスパケットサイズ又はページ数等）が、転送条件レジスタ TREG0 ~ TREGe に設定される。即ち、PIPE0、PIPEa、PIPEb、PIPEc、PIPEd、PIPEe の転送条件情報は、各々、TREG0、TREGa、TREGb、TREGc、TREGd、TREGe に設定（記憶）される。この設定は、例えばファームウェア（CPU）により行われる。

【0160】

そしてホストコントローラ 50（広義には転送コントローラ）は、転送条件レジスタ TREG0 ~ TREGe に設定された転送条件情報に基づいて、エンドポイントに対するトランザクションを発生する。そして、パイプ領域とそれに対応するエンドポイントとの間で、データ（パケット）を自動転送する。

40

【0161】

このように本実施形態では、各パイプ領域（バッファ領域）に対応して各転送条件レジスタが設けられ、この各転送条件レジスタに設定された転送条件情報に基づいて、各パイプ領域のパイプ転送（所与のデータ単位の転送）がホストコントローラ 50 により自動的に行われる。従って、ファームウェア（ドライバ、ソフトウェア）は、転送条件レジスタに転送条件情報を設定した後は、データ転送が完了するまで、データ転送制御に関わらなく

50

て済むようになる。そして、所与のデータ単位のパイプ転送が完了すると割り込みが発生し、転送の完了がファームウェアに伝えられる。これにより、ファームウェア（CPU）の処理負荷を格段に低減できる。

【0162】

なお、本実施形態では、PIPE0は、エンドポイント0（コントロール転送のエンドポイント）に専用のパイプ領域に設定され、PIPEa～PIPEeは、任意のエンドポイントに割り当て可能なパイプ領域に設定される。この場合に本実施形態では、PIPE0（コントロール転送のパイプ領域）に対応する転送条件レジスタTREG0に設定された転送条件情報に基づいて、図7で説明したようにコントロール転送のトランザクションを自動発生する。そして、PIPE0とエンドポイント0との間で、データを自動転送する

10

【0163】

なお、本実施形態では図15に示すようにペリフェラル動作時には、エンドポイント領域EP0～EPeとホストとの間で行われるデータ転送の転送条件情報（転送方向、転送種別、マックスパケットサイズ又はページ数等）が、転送条件レジスタTREG0～TREGeに設定される。そしてペリフェラルコントローラ60（広義には転送コントローラ）は、転送条件レジスタTREG0～TREGeに設定された転送条件情報に基づいて、エンドポイント領域とホストとの間でのデータ転送を行う。

【0164】

このように本実施形態では、転送条件レジスタTREG0～TREGeが、ホスト動作時とペリフェラル動作時とで共用（兼用）される。これにより、レジスタ部70のリソースを節約でき、データ転送制御装置を小規模化できる。

20

【0165】

図16に、レジスタ部70のレジスタ構成例を示す。なお、レジスタ部70のレジスタの一部を、各ブロック（OTGC、HC、PC、Xcvr等）内に含ませてもよい。

【0166】

図16に示すように、レジスタ部70の転送条件レジスタ（TREG0～TREGeの各々）は、ホスト動作時（HC、PIPE）とペリフェラル動作時（PC、EP）で共用されるHC/PC共用レジスタ（共用転送条件レジスタ）を含む。また、ホスト動作時のみ使用されるHC（PIPE）用レジスタ（ホスト用転送条件レジスタ）を含む。また、ペリフェラル動作時のみ使用されるPC（EP）用レジスタ（ペリフェラル用転送条件レジスタ）を含む。また、パケットバッファ（FIFO）のアクセス制御などを行うためのレジスタであり、ホスト動作時とペリフェラル動作時で共用されるアクセス制御レジスタを含む。

30

【0167】

例えば、デュアルロール・デバイスのホスト動作時に、ホストコントローラ50（HC）は、HC/PC共用レジスタとHC用レジスタに設定される転送条件情報に基づいて、データ（パケット）を転送する。

【0168】

一方、ペリフェラル動作時には、ペリフェラルコントローラ60（PC）は、HC/PC共用レジスタとPC用レジスタに設定される転送条件情報に基づいて、データ（パケット）を転送する。

40

【0169】

また、ホスト動作時、ペリフェラル動作時の両方において、バッファコントローラ80は、共用アクセス制御レジスタに基づいて、パケットバッファ100へのアクセス制御（リード/ライト・アドレスの発生、データのリード/ライト、アクセスの調停等）を行うことになる。

【0170】

図16のHC/PC共用レジスタには、データの転送方向（IN、OUT又はSETUP等）、転送種別（アイソクロナス、バルク、インタラプト、コントロールなどのトランザ

50

クシヨンの種別)、エンドポイント番号(各USBデバイスのエンドポイントに関連づけられる番号)、マックスパケットサイズ(エンドポイントが送信又は受信可能なパケットの最大ペイロードサイズ。ページサイズ)が設定される。また、バッファ領域(パイプ領域、エンドポイント領域)のページ数(バッファ領域の面数)が設定される。また、DMA接続の有無(DMAハンドラ回路112によるDMA転送の使用の有無)を指示する情報が設定される。

【0171】

HC(PIPE)用レジスタには、インタラプト転送のトークン発行周期(インタラプト・トランザクションを起動する周期、インターバル)が設定される。また、トランザクションの連続実行回数(パイプ領域間の転送比率を設定する情報。各パイプ領域のトランザクションの連続実行回数)が設定される。また、ファンクションアドレス(エンドポイントを有するファンクションのUSBアドレス)、転送データのトータルサイズ(各パイプ領域を介して転送されるデータのトータルサイズ。IRPなどのデータ単位)が設定される。また、自動トランザクションの開始指示(ホストコントローラに対する自動トランザクション処理の開始指示)が設定される。また、自動コントロール転送モードの指示(コントロール転送のセットアップステージ、データステージ、ステータスステージのトランザクションを自動発生するモードの指示)が設定される。

10

【0172】

PC(EP)用レジスタには、エンドポイントイネーブル(エンドポイントのイネーブルやディスエーブルの指示)、ハンドシェーク指定(各トランザクションで行われるハンドシェークの指定)が設定される。

20

【0173】

パケットバッファ(FIFO)用の共用アクセス制御レジスタには、バッファ・I/Oポート(CPUによりPIO転送を行う場合のI/Oポート)が設定される。また、バッファ・フル/エンプティ(各バッファ領域のフル、エンプティの通知)、バッファ・残りデータサイズ(各バッファ領域の残りデータサイズ)が設定される。

【0174】

レジスタ部70は、インタラプト系レジスタ、ブロック系レジスタ、DMA制御レジスタなども含む。

【0175】

インタラプト系レジスタは、割り込みのステータス(要因)をCPUに対して示すためのインタラプト・ステータスレジスタ、割り込みのイネーブル、ディスエーブル(非マスク、マスク)を設定するインタラプト・イネーブルレジスタを含む。なお、割り込みには、OTGコントローラ20系、ホストコントローラ50系、ペリフェラルコントローラ60系の割り込みがある。

30

【0176】

ブロック系レジスタは、ブロック間で共用されるブロック間共用レジスタや、各ブロック(Xcvr、OTGC、HC、PC)内で使用されるブロック用レジスタを含む。

【0177】

ブロック間共用レジスタには、各ブロックのリセットを指示するレジスタなどがある。ブロック用レジスタには、トランシーバ10(Xcvr)を制御するためのレジスタや、OTGコントローラ20(OTGC)のステートコマンドレジスタや、ホストコントローラ50(HC)のステートコマンドレジスタや、フレーム番号を設定するレジスタなどがある。

40

【0178】

以上のように本実施形態では、ホスト動作時とペリフェラル動作時で共用されるレジスタ(HC/PC共用レジスタ、共用アクセス制御レジスタ)をレジスタ部70に設けている。これにより、ホスト動作時用のレジスタとペリフェラル動作時用のレジスタを全く別個に設ける場合に比べて、レジスタ部70を小規模化できる。また、CPU上で動作するファームウェア(ドライバ)から見た共用レジスタのアクセスアドレスを、ホスト動作時と

50

ペリフェラル動作時とで同一にできる。従って、ファームウェアは、これらの共用レジスタを同一アドレスで管理できるようになり、ファームウェア処理を簡素化できる。

【0179】

また、HC用レジスタや、PC用レジスタを設けることで、ホスト動作時(P I P E)の転送やペリフェラル動作時(E P)の転送に特有の転送条件を設定できる。例えば、トークン発行周期を設定することで、ホスト動作時にインタラプト転送のトークンを所望の周期で発行することが可能になる。また、連続実行回数を設定することで、ホスト動作時にパイプ領域間の転送比率を任意に設定できる。また、トータルサイズを設定することで、ホスト動作時にパイプ領域を介して自動転送されるデータのサイズを任意に設定できる。またファームウェアは、ホスト動作時に、自動トランザクションの開始を指示したり、自動コントロール転送モードのオン/オフを指示できるようになる。

10

【0180】

9. 各ブロックの詳細な構成例

次に各ブロックの詳細な構成例について説明する。

【0181】

9.1 OTGコントローラ

図17に、OTGコントローラ20の構成例を示す。

【0182】

OTGコントローラ20は、OTGレジスタ部22を含む。このOTGレジスタ部22は、OTGコントローラ20のモニタレジスタや制御レジスタを含む。またファームウェア(C P U)により書き込まれるOTGステートコマンドをデコードする回路を含む。

20

【0183】

またOTGコントローラ20はOTG制御回路23を含む。そして、このOTG制御回路23は、OTGステートの管理を行うOTG管理回路24、IDピンの電圧レベルを検出するID検出回路25、V B U Sの電圧レベルを検出するV B U S検出回路26、D P及びD Mのラインステートを検出するラインステート検出回路27を含む。

【0184】

またOTGコントローラ20は、OTGステートの遷移判断条件の1つである時間を計測するタイマ28を含む。

【0185】

OTGステートを遷移させるために検出すべき情報は、ID、V B U Sの電圧レベル、D P / D Mのラインステートである。本実施形態のOTGコントローラ20は、これらの情報を検出し、モニタレジスタを介してファームウェア(C P U)に伝える。

30

【0186】

ファームウェアは、これらの検出情報に基づいて自身のステートを遷移させると共に、次に遷移すべきステートを、OTGステートコマンドを用いてOTGコントローラ20に伝える。

【0187】

OTGコントローラ20は、OTGステートコマンドをデコードし、そのデコード結果に基づいて、V B U Sのドライブ制御、プルアップ/プルダウン抵抗の接続制御等を行い、図2(A)、(B)で説明したS R PやH N Pを実現する。

40

【0188】

このように本実施形態では、ステート毎のOTG制御はOTGコントローラ20が担当し、ファームウェアはステートの遷移管理に専念できる。この結果、全てのステート制御をファームウェアで実現する場合に比べて、ファームウェア(C P U)の処理負荷を軽減できると共に、効率的なファームウェア開発が可能になる。

【0189】

なお、OTGのステート遷移の判断を、ファームウェアが行わずに、ハードウェア回路が行うようにしてもよい。或いは、OTGコントローラ20のほとんど全ての処理(例えばV B U S制御、プルアップ/プルダウン抵抗制御、ID検出、V B U S検出、ラインステ

50

ート検出以外の処理)をファームウェア(ソフトウェア)により実現してもよい。

【0190】

9.2 ホストコントローラ、ペリフェラルコントローラ

図18(A)に、ホストコントローラ50の構成例を示す。

【0191】

ホストコントローラ50はHCシーケンス管理回路52を含む。このHCシーケンス管理回路52は、パイプ転送(パイプ領域を用いたデータ転送)の調停、時間管理、パイプ転送のスケジューリング、再送管理などを行う。

【0192】

より具体的にはHCシーケンス管理回路52は、フレーム番号のカウントや、SOF(Start-Of-Frame)パケットの送信指示を行う。また、アイソクロナス転送を各フレームの先頭で優先的に実行するための処理を行ったり、インタラプト転送をアイソクロナス転送の次に優先的に取り扱うための処理を行う。また、パイプ転送の順序に従って各パイプ転送を指示する処理を行う。また、トランザクションの連続実行回数を管理したり、残りフレーム時間の確認処理を行う。また、ペリフェラルから返ってきたハンドシェイクパケット(ACK、NAK)に対する処理を行う。また、トランザクション実行時のエラー処理を行う。

【0193】

ホストコントローラ50はターゲットパイプ管理回路54を含む。このターゲットパイプ管理回路54は、レジスタ部70の転送条件レジスタに設定された転送条件情報のハンドリング処理などを行う。

【0194】

より具体的にはターゲットパイプ管理回路54は、転送条件情報の選択処理や、割り込み信号の生成処理を行う。また自動トランザクションの開始が指示された場合に、そのパイプ領域の転送データのトータルサイズをロードする。そして、残り転送データサイズのカウント(デクリメント)処理を行う。また、バッファコントローラ80へのデータの送受信の際にバッファ(FIFO)領域の状態を確認する処理を行う。また、トランザクション管理回路56への転送指示を行う。また、予期しないショートパケットの受信の判断処理や、マックスパケットサイズ以上のパケットの受信の判断処理を行う。また、零長パケットを自動転送するモードが設定されている場合には、最後の零長パケットの送信をトランザクション管理回路56に指示する。また、自動コントロール転送モードでのシーケンス管理を行う。

【0195】

ホストコントローラ50はトランザクション管理回路56を含む。このトランザクション管理回路56は、転送パケットの種類や転送順序の管理(トランザクションのシーケンス管理)を行う。また、タイムアウトの監視処理を行う。また、トランザクション終了の通知処理を行う。

【0196】

ホストコントローラ50はパケットハンドラ回路58を含む。このパケットハンドラ回路58は、パケットの生成、分解処理を行う。また、PIDのチェックやCRCのデコード、エンコードを行う。また、バッファ領域のパケットのペイロードのリード、ライト処理や、SOFパケットの送信処理を行う。また、送受信データのカウント処理を行う。

【0197】

図18(B)にペリフェラルコントローラ60の構成例を示す。

【0198】

ペリフェラルコントローラ60は、トランザクション管理回路62、パケットハンドラ回路64を含む。これらのトランザクション管理回路62、パケットハンドラ回路64は、ホストコントローラ50のトランザクション管理回路56、パケットハンドラ回路58とほぼ同様の処理を行う。

【0199】

10

20

30

40

50

9.3 バッファコントローラ

図19にバッファコントローラ80の構成例を示す。

【0200】

バッファコントローラ80は領域確保(allocation)回路82を含む。この領域確保回路82は、パケットバッファ100に、バッファ領域(ホスト動作時にパイプ領域に設定され、ペリフェラル動作時にエンドポイント領域に設定される領域)を確保する回路である。

【0201】

領域確保回路82は領域計算回路83を含む。この領域計算回路83は、マックスパケットサイズ(広義にはページサイズ)やページ数に基づいて、バッファ領域の領域サイズ、

10

【0202】

例えば図20(A)に示すバッファ領域PIPE0/EP0、PIPEa/EPa、PIPEb/EPb、PIPEc/EPcでは、マックスパケットサイズ(MaxPktSize)が、各々、32、64、64、64バイトに設定され、ページ数(BufferPage)が、各々、1、1、3、2ページに設定されている。領域計算回路83は、これらのマックスパケットサイズ、ページ数などに基づいて、バッファ領域PIPE0/EP0~PIPEc/EPcの領域サイズ、スタートアドレス、エンドアドレスを計算する。例えば図20(A)において、PIPE0/EP0、PIPEa/EPa、PIPEb/EPb、PIPEc/EPcの領域サイズは、各々、32(=32×1)、64(=64×1)、192(=64×3)、128(=64×2)バイトと計算されることになる。

20

【0203】

ポインタ割り当て回路84は、各バッファ領域の書き込みポインタWP(WP0、WPa、WPb、WPc)、読み出しポインタRP(RP0、RPa、RPb、RPC)を、DMA用ポインタ、CPU用ポインタ、USB用ポインタに割り当てる回路である。

【0204】

例えば図20(B)に示すように、データ送信時(DMA又はCPUからパケットバッファ100を介してUSB側にデータが転送される時)であり、且つ、DMA転送使用時には、そのバッファ領域の書き込みポインタWPはDMA(DMAアクセス)用のポインタに割り当てられ、読み出しポインタRPはUSB(USBアクセス)用のポインタに割り

30

【0205】

一方、図20(C)に示すように、データ受信時(USBからパケットバッファ100を介してDMA又はCPU側にデータが転送される時)であり、且つ、DMA転送使用時には、そのバッファ領域の書き込みポインタWPはUSB用ポインタに割り当てられ、読み出しポインタRPはDMA用ポインタに割り当てられる。また、データ受信時であり且つCPU転送使用時には、そのバッファ領域の書き込みポインタWPはUSB用ポインタに割り当てられ、読み出しポインタRPはCPU用ポインタに割り当てられる。

40

【0206】

なお、各バッファ領域の書き込みポインタWP、読み出しポインタRPのポインタ情報(位置情報)は、レジスタ部70の各転送条件レジスタ(PIPE/EPレジスタ)に保持される。

【0207】

ポインタ管理回路86は、ポインタの更新を行いながら、パケットバッファ100にアクセスするための実アドレスを生成する回路である。

【0208】

ポインタ管理回路86は、CPU用アドレス生成回路87、DMA用アドレス生成回路88、USB用アドレス生成回路89を含む。これらの生成回路87、88、89は、各々

50

、ポインタ割り当て回路 84 により割り当てられた CPU 用ポインタ、DMA 用ポインタ、USB 用ポインタに基づいて、CPU 用アドレス、DMA 用アドレス、USB 用アドレスを生成する。また、CPU (CPU インターフェース回路)、DMA (DMA ハンドラ回路) からのアクセス毎に、或いは USB (HC 又は PC) のトランザクション終了 (ACK、NAK などのハンドシェイク送受信) 毎に、ポインタを更新する処理を行う。なお、更新後のポインタの情報は、領域確保回路 82 を介してレジスタ部 70 の各転送条件レジスタに書き戻される。

【0209】

バッファ管理回路 90 は、パケットバッファ 100 へのアクセスを管理する回路である。

【0210】

バッファ管理回路 90 はバッファインターフェース回路 92 を含む。このバッファインターフェース回路 92 は、ポインタ管理回路 86 からの CPU 用アドレス、DMA 用アドレス、USB 用アドレスなどを受け、パケットバッファ 100 へのデータの入出力や、アドレス、出力イネーブル、ライトイネーブル、リードイネーブルなどの出力を行う。

【0211】

バッファ管理回路 90 は調停回路 93 を含む。この調停回路 93 は、CPU (CPU インターフェース回路)、DMA (DMA ハンドラ回路)、USB (ホストコントローラ又はペリフェラルコントローラ) からのアクセスを調停する回路である。この調停結果に基づいて、CPU 用アドレス、DMA 用アドレス、USB 用アドレスのいずれかが、パケットバッファ 100 のアクセス・アドレスとして出力され、CPU、DMA 又は USB とパケットバッファ 100 との間のデータ転送経路が設定される。

【0212】

HC / PC セレクタ 94 は、バッファ管理回路 90 (バッファコントローラ 80) とホストコントローラ 50 (HC) 又はペリフェラルコントローラ 60 (PC) との間の接続の切り替え制御を行う。例えばホスト動作時には、ホストコントローラ 50 とバッファ管理回路 90 を接続し、ペリフェラル動作時には、ペリフェラルコントローラ 60 とバッファ管理回路 90 を接続する。なお、この接続の切り替え制御は、OTG コントローラ 20 (OTGC) からの HC / PC イネーブル信号に基づいて行う。

【0213】

10. 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。

【0214】

例えば図 21 (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 22 (A) にその外観図を示す。CPU 510 (処理部) 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。ROM 516 には、制御プログラム、フォントなどが格納され、RAM 517 (システムメモリ) は CPU 510 のワーク領域として機能する。DMAC 518 は、CPU 510 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。

【0215】

USB を介してパーソナルコンピュータ、デジタルカメラ、デジタルビデオカメラなどの他の機器から送られてきたシリアル印刷データ (印字データ、画像データ) は、データ転送制御装置 500 によりパラレル印刷データに変換される。そして、変換後のパラレル印刷データは、CPU 510 又は DMAC 518 により、印刷処理部 (プリンタエンジン) 512 に送られる。そして、印刷処理部 512 においてパラレル印刷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刷部 (データの出力処理を行う装置) 514 により紙に印刷されて出力される。

【0216】

図 21 (B) に電子機器の 1 つであるデジタルカメラの内部ブロック図を示し、図 22 (B) にその外観図を示す。CPU 520 はシステム全体の制御などを行う。操作部 521

10

20

30

40

50

(シャッターボタン、操作ボタン等)はデジタルカメラをユーザが操作するためのものである。ROM 526には制御プログラムなどが格納され、RAM 527はCPU 520のワーク領域として機能する。DMAC 528はDMAコントローラである。

【0217】

CCD、レンズなどからなる撮像部(データの取り込み処理を行う装置)522により画像が撮像され、撮像された画像のデータは画像処理部524により処理される。そして、処理後の画像データは、CPU 520又はDMAC 528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を介してプリンタ、ストレージ装置、パーソナルコンピュータなどの他の機器に送信する。

10

【0218】

図21(C)に電子機器の1つであるCD-RWドライブ(ストレージ装置)の内部ブロック図を示し、図22(C)にその外観図を示す。CPU 530はシステム全体の制御などを行う。操作部531はCD-RWをユーザが操作するためのものである。ROM 536には制御プログラムなどが格納され、RAM 537はCPU 530のワーク領域として機能する。DMAC 538はDMAコントローラである。

【0219】

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-RW 532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、CPU 530又はDMAC 538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を介して他の機器に送信する。

20

【0220】

一方、USBを介して他の機器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、CPU 530又はDMAC 538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-RW 532に記憶される。

【0221】

なお、図21(A)、(B)、(C)において、CPU 510、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よい。

30

【0222】

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、OTG機能を有する電子機器を実現できる。即ち、電子機器にホストとしての役割を持たせたり、デバイスとしての役割を持たせることが可能になり、これまでに存在しなかったアプリケーションを創出できる。

【0223】

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、電子機器に組み込まれるCPU(処理部)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。また、CPUが、データ転送制御処理以外の他の処理を余裕を持って行うことが可能になり、電子機器の性能向上や低コスト化を図れる。また、CPU上で動作するファームウェアのプログラムを簡素化でき、電子機器の開発期間の短縮化を図れる。

40

【0224】

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、デジタルビデオカメラ、携帯電話、スキャナ、TV、VTR、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、或いは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。

50

【0225】

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。

【0226】

例えば、本発明のデータ転送制御装置の構成は、図4等で説明した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。

【0227】

また、自動コントロール転送を実現する回路の構成も図13に示す構成に限定されない。

【0228】

また、データ転送制御装置の各ブロック（HC、PC、OTGC等）の構成も、本実施形態で説明したものに限定されず、種々の変形実施が可能である。 10

【0229】

また、転送条件レジスタに設定される転送条件情報も、本実施形態で説明した情報に限定されるものではない。

【0230】

また、明細書中の記載において広義な用語（ステートコントローラ、処理部、転送コントローラ、バス、転送比率情報、バッファ領域、ページサイズ等）として引用された用語（OTGコントローラ、CPU・ファームウェア、ホストコントローラ・ペリフェラルコントローラ、USB、連続実行回数、パイプ領域・エンドポイント領域、マックスパケットサイズ等）は、明細書中の他の記載においても広義な用語に置き換えることができる。 20

【0231】

また、本発明のうち従属請求項に係る発明においては、従属先の請求項の構成要件の一部を省略する構成とすることもできる。また、本発明の1の独立請求項に係る発明の要部を、他の独立請求項に従属させることもできる。

【0232】

また、本実施形態ではUSBのOTG規格への適用例を説明したが、本発明が適用されるのはOTG規格に限定されない。例えばOTG規格と同様の思想に基づく規格やOTG規格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)は、USBのOTG規格について説明するための図である。 30

【図2】図2(A)、(B)は、SRPやHNPの手順について説明するための図である。

【図3】図3(A)、(B)は、OHCIのリスト構造のディスクリプタなどについて説明するための図である。

【図4】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5】図5(A)、(B)は、パイプ領域、エンドポイント領域について説明するための図である。

【図6】デバイスリクエストについて説明するための図である。

【図7】本実施形態の自動コントロール転送手法について説明するための図である。 40

【図8】図8(A)、(B)、(C)は、自動コントロール転送時の信号波形例を示す図である。

【図9】図9(A)、(B)、(C)は、セットアップステージ、データステージでの信号波形例を示す図である。

【図10】図10(A)、(B)は、ステータスステージでの信号波形例を示す図である。

【図11】自動コントロール転送モードをオンにした時のファームウェアの処理例を示すフローチャートである。

【図12】自動コントロール転送モードをオフにした時のファームウェアの処理例を示すフローチャートである。 50

【図13】自動コントロール転送を実現する回路の例である。

【図14】データ転送制御装置のホスト時の動作について説明するための図である。

【図15】データ転送制御装置のペリフェラル時の動作について説明するための図である。

【図16】レジスタ部について説明するための図である。

【図17】OTGコントローラの詳細な構成例を示す図である。

【図18】図18(A)、(B)は、ホストコントローラ、ペリフェラルコントローラの詳細な構成例を示す図である。

【図19】バッファコントローラの詳細な構成例を示す図である。

【図20】図20(A)、(B)、(C)は、領域確保手法やポインタ割り当て手法について説明するための図である。 10

【図21】図21(A)、(B)、(C)は、種々の電子機器の内部ブロック図の例である。

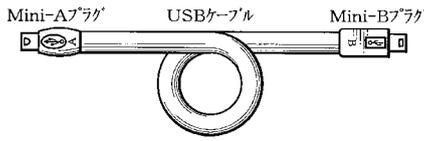
【図22】図22(A)、(B)、(C)は、種々の電子機器の外観図の例である。

【符号の説明】

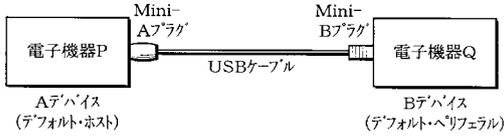
PIPE0 ~ PIPEe	パイプ(バッファ)領域	
EP0 ~ EPe	エンドポイント(バッファ)領域	
TREG0 ~ TREGe	転送条件レジスタ(共用レジスタ)	
10	トランシーバ	
12	物理層回路	20
20	OTGコントローラ(ステートコントローラ)	
30	HC/PC切り替え回路	
32	HC/PCセクタ	
34	ラインステートコントローラ	
40	転送コントローラ	
50	ホストコントローラ	
60	ペリフェラルコントローラ	
70	レジスタ部	
72	転送条件レジスタ部(共用レジスタ)	
80	バッファコントローラ	30
100	パケットバッファ(FIFO、RAM)	
110	インターフェース回路	
112	DMAハンドラ回路	
114	CPUインターフェース回路	
120	クロックコントローラ	

【 図 1 】

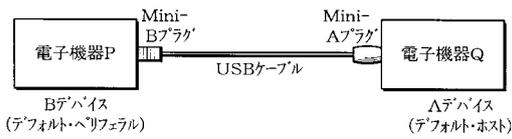
(A)



(B)

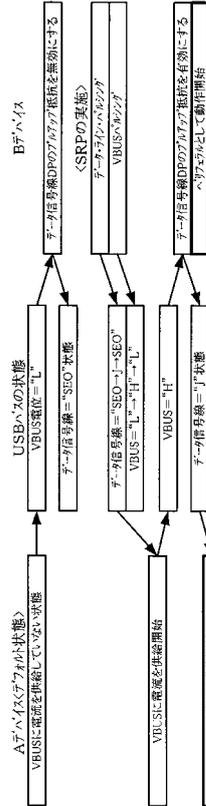


(C)

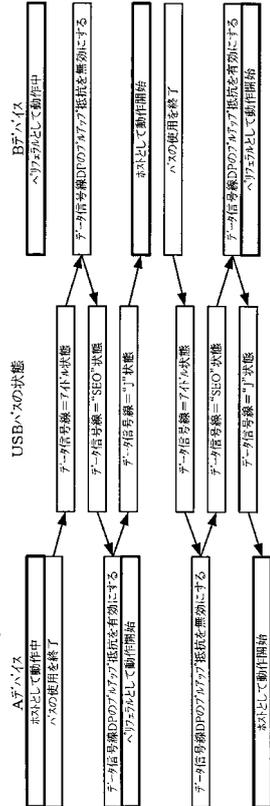


【 図 2 】

(A) SRP (Session Request Protocol)

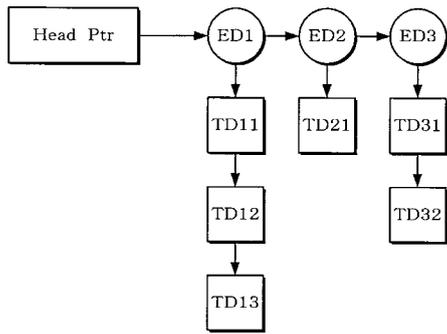


(B) HNP (Host Negotiation Protocol)



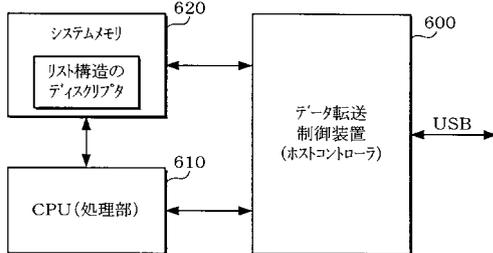
【 図 3 】

(A)

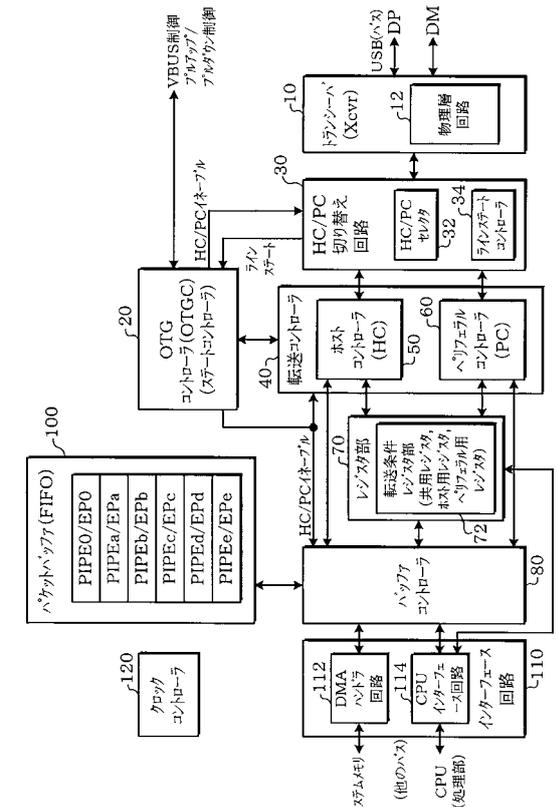


リスト構造のディスクリプタ

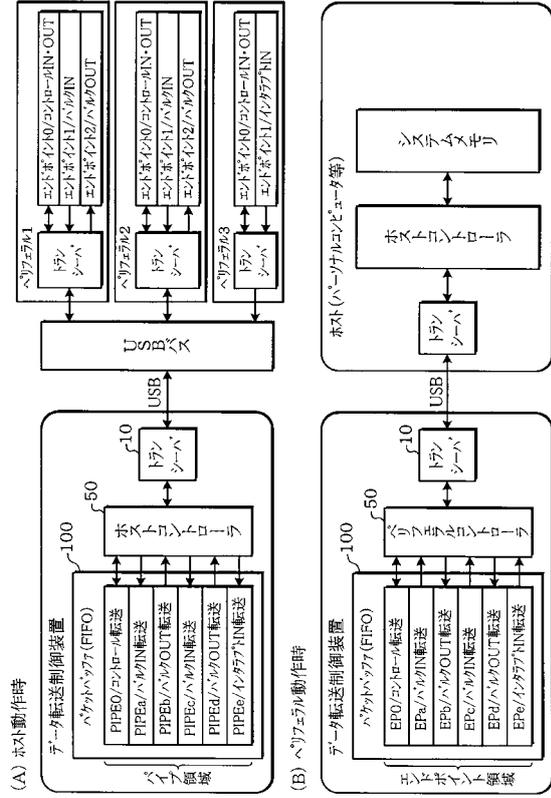
(B) ホスト(パーソナルコンピュータ等)



【 図 4 】



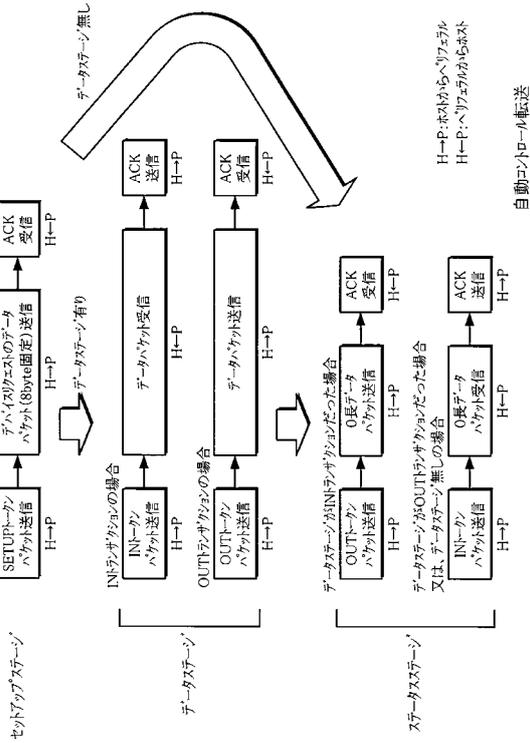
【 図 5 】



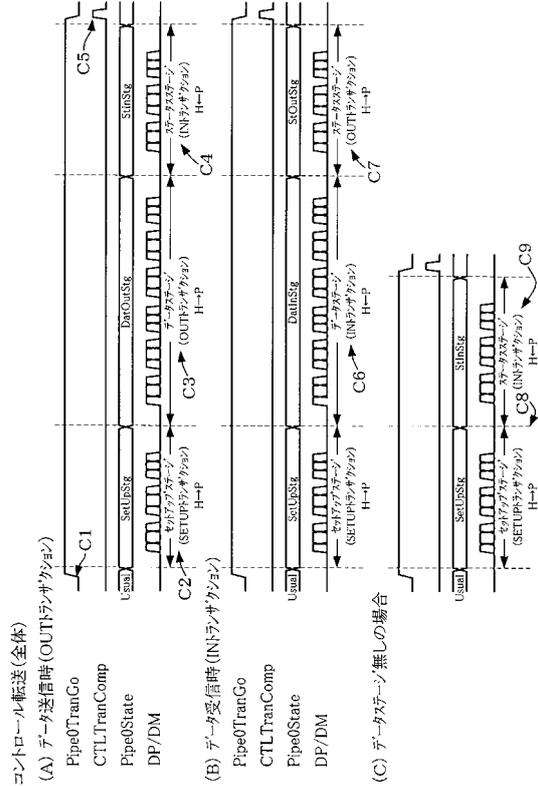
【 図 6 】

デバイスリクエスト	データステージの有無	データステージでの転送方向	ステータスステージでの転送方向
SET-ADDRESS	無	—	—
GET-DESCRIPTOR	有	IN	OUT
SET-DESCRIPTOR	有	OUT	IN
SET-FEATURE	無	—	—

【 図 7 】



【 図 8 】



コントロール転送(全体)

(A) データ送信時(OUTランダム)

Pipe0TranGo

CTLTranComp

Pipe0State

DP/DM

(B) データ受信時(INランダム)

Pipe0TranGo

CTLTranComp

Pipe0State

DP/DM

(C) データステージ無しの場合

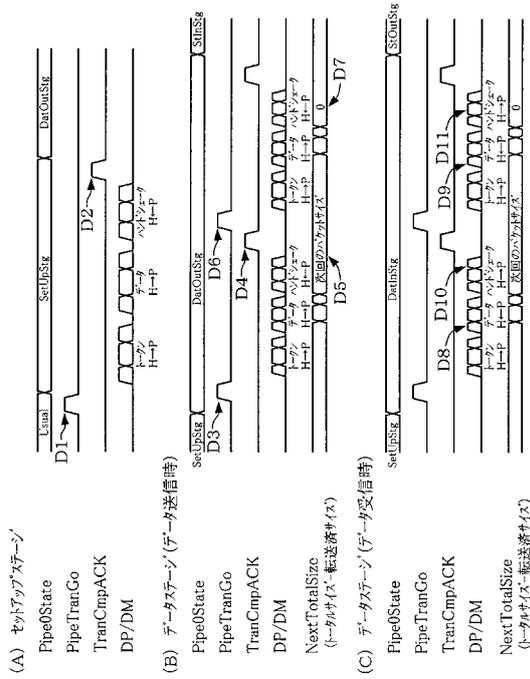
SetUSBSig

barOutSig

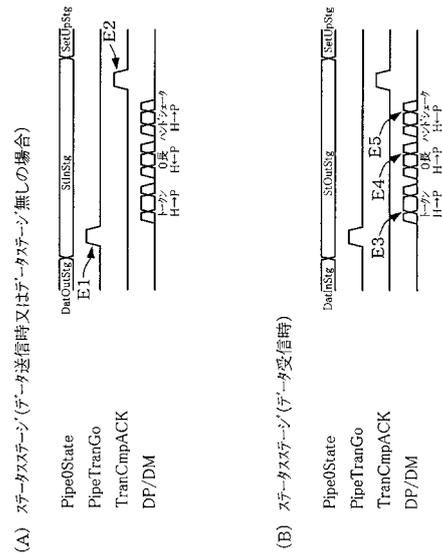
barInSig

SetUSBSig

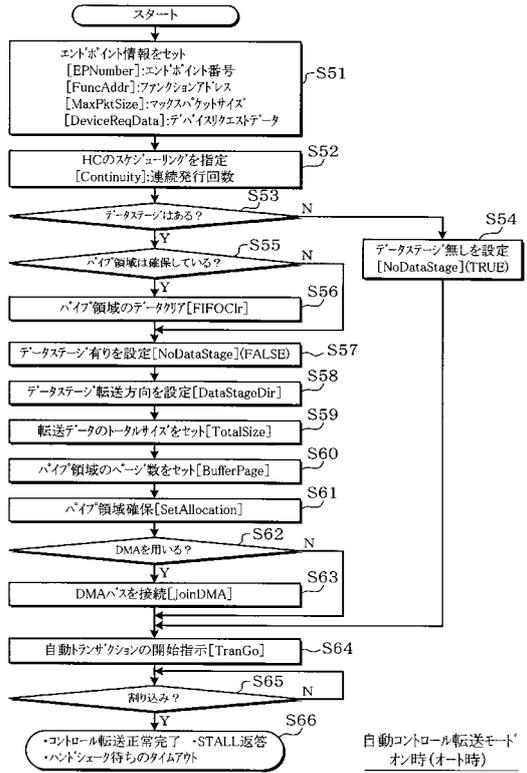
【 図 9 】



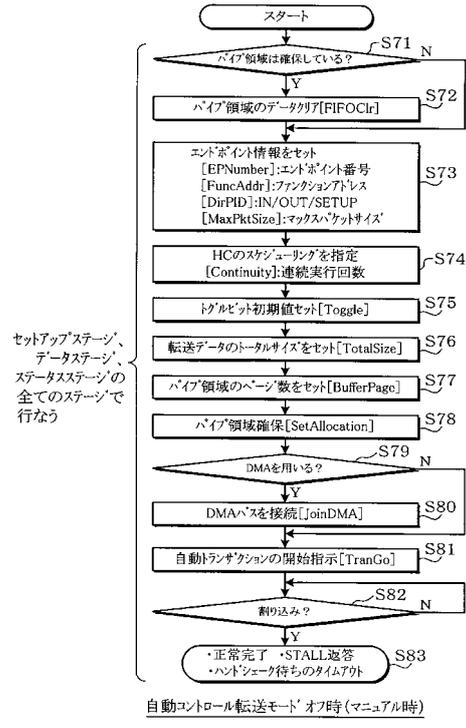
【 図 10 】



【 図 11 】

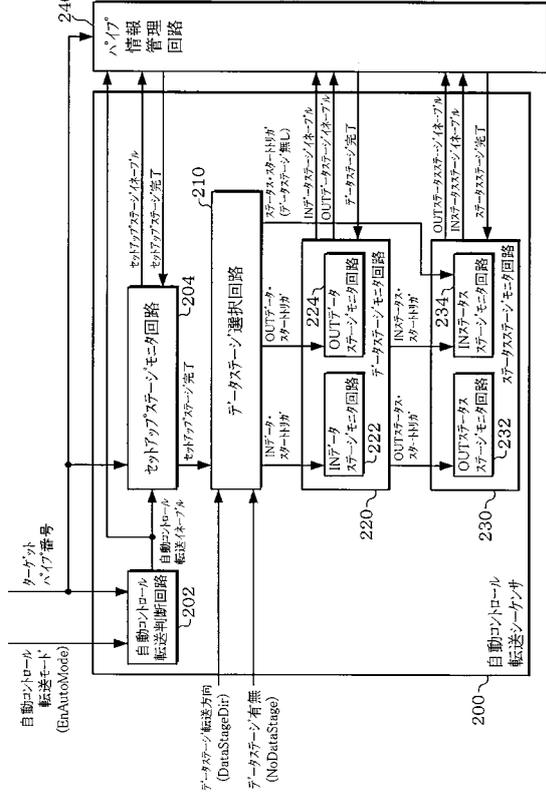


【 図 12 】

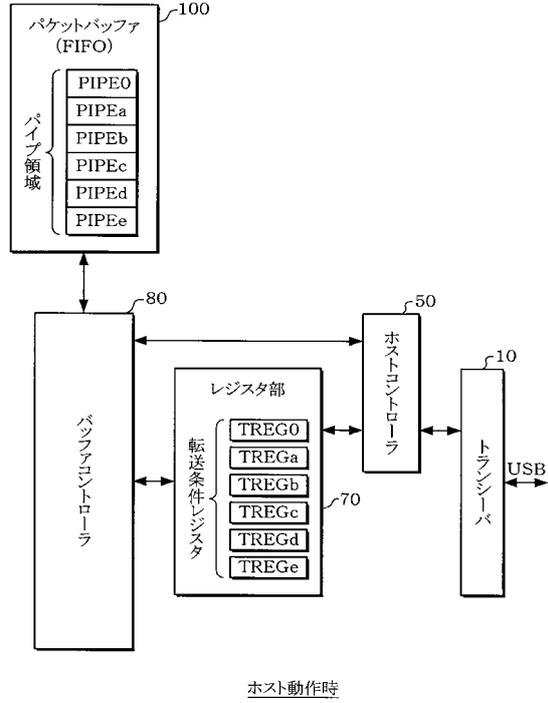


セットアップステージ、データステージ、ステータスステージの全てのステージで行なう

【図 13】

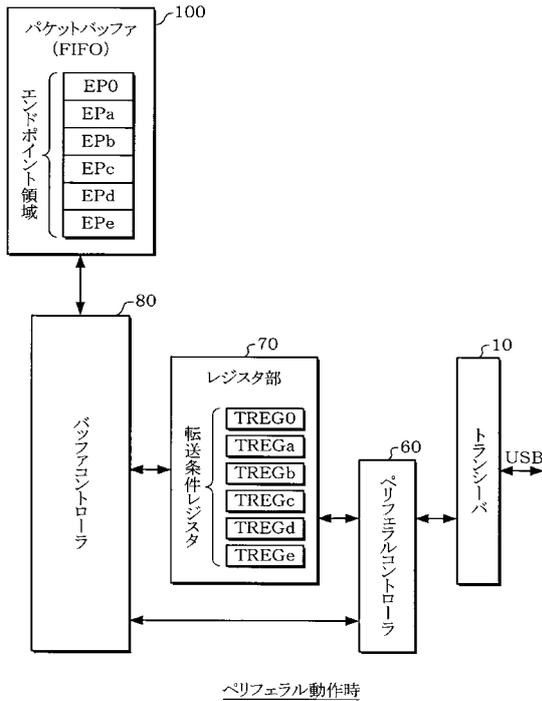


【図 14】



ホスト動作時

【図 15】

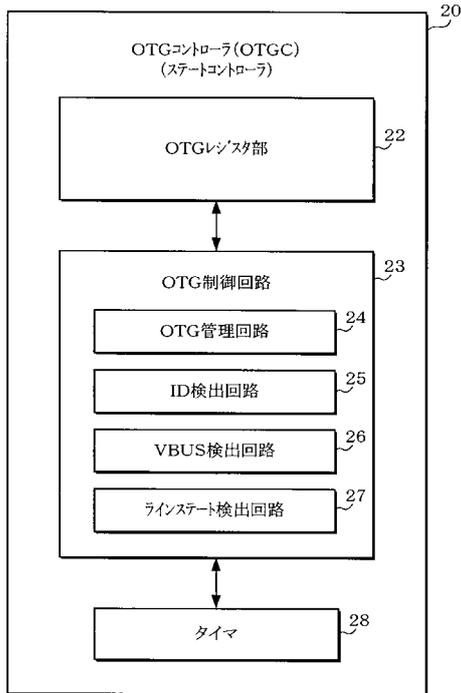


ペリフェラル動作時

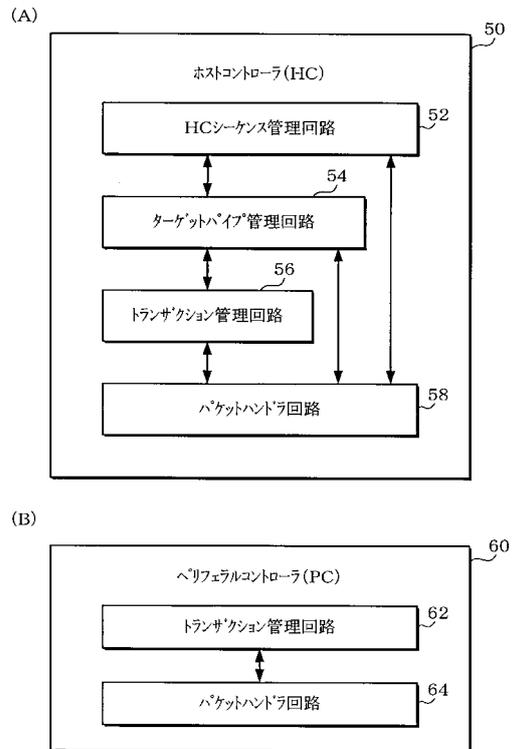
【図 16】

レジスタ部	レジスタ名	機能	動作時
転送条件レジスタ (PIPE/EP制御レジスタ) (O, a~e)	HC/PC(PIPE/EP) 共用レジスタ	データの転送方向、転送種別、エントリ番号、マックスパケットサイズ、ヘジ数、DMA接続の有無等	データの転送方向、転送種別、エントリ番号、マックスパケットサイズ、ヘジ数、DMA接続の有無等
	HC(PIPE)用レジスタ	インフラ転送のトーン発行制御、トランザクションの連続実行回数、アンクショアトレス、転送データのトータルサイズ、自動小断割込の開始指示、自動小断制御の開始指示	インフラ転送のトーン発行制御、トランザクションの連続実行回数、アンクショアトレス、転送データのトータルサイズ、自動小断割込の開始指示、自動小断制御の開始指示
インフラ系レジスタ	PC(EP)用レジスタ	ペリフェラル動作時(EP)にのみ使用されるレジスタ	ペリフェラル動作時(EP)にのみ使用されるレジスタ
	パケットバッファ(FIFO)用の制御レジスタ	パケットバッファ(FIFO)のアクセス制御を行なうためのレジスタ	パケットバッファ(FIFO)のアクセス制御を行なうためのレジスタ
フレグ系レジスタ	インフラ、スタートレジスタ	動作のスタート要因(要因)を示すレジスタ	動作のスタート要因(要因)を示すレジスタ
	インフラ、イベントレジスタ	動作のイベント、アドレスを指定するレジスタ	動作のイベント、アドレスを指定するレジスタ
DMA制御レジスタ	フレグ間共用レジスタ	フレグ間で共用されるレジスタ	フレグ間で共用されるレジスタ
	フレグ用レジスタ	Xerr, OTCC, HC, PCの各フレグ内で使用されるレジスタ	Xerr, OTCC, HC, PCの各フレグ内で使用されるレジスタ
DMA制御レジスタ	—	DMA転送に関する設定を行うレジスタ	DMA転送開始指示、DMA転送データのポート番号等
	—	—	DMA転送開始指示、DMA転送データのポート番号等

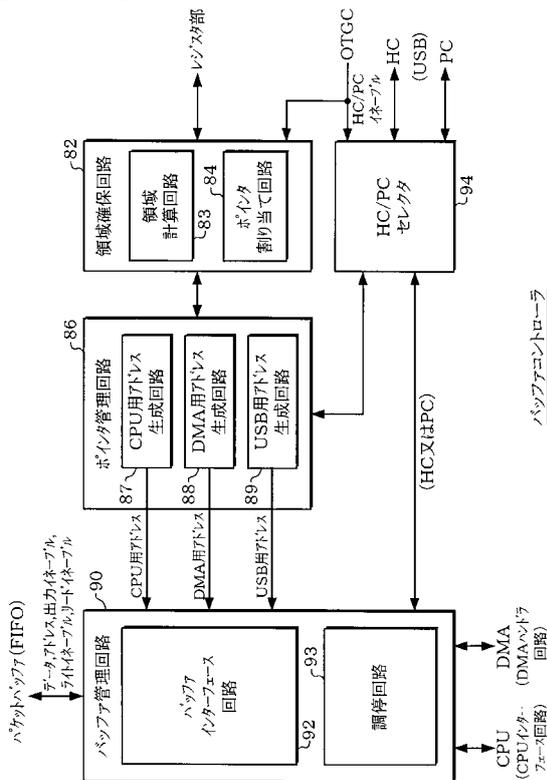
【図17】



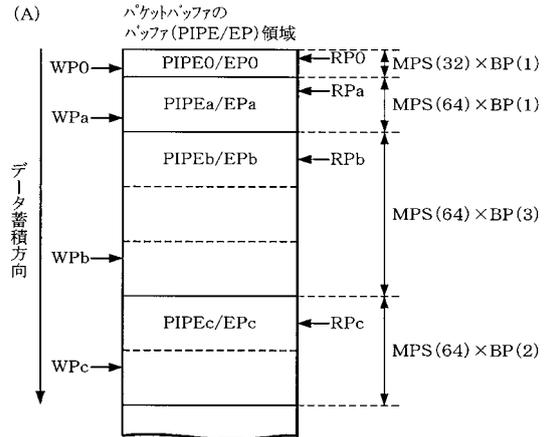
【図18】



【図19】



【図20】



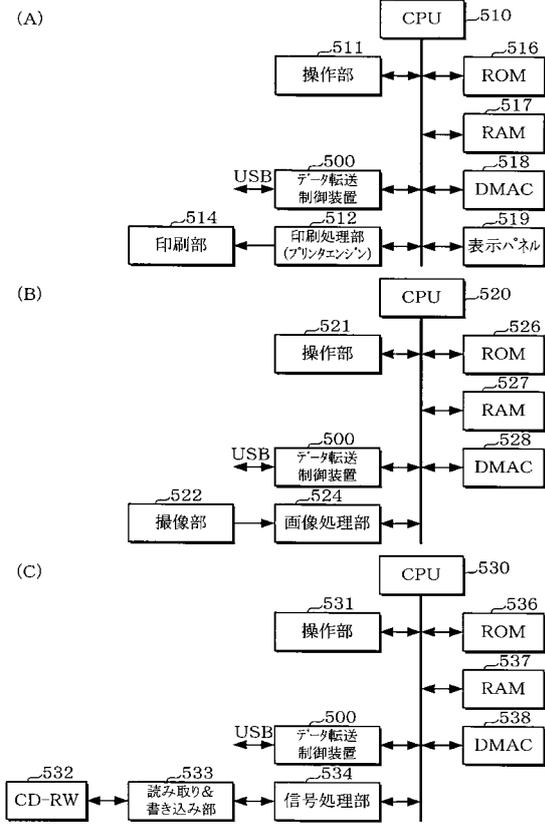
(B) 送信時(DMA又はCPU→パケットバッファ→USB)

	DMA転送使用	CPU転送使用
WP	DMA	CPU
RP	USB	USB

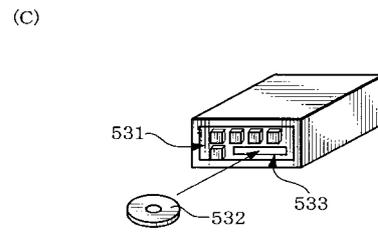
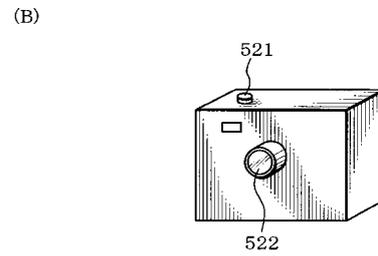
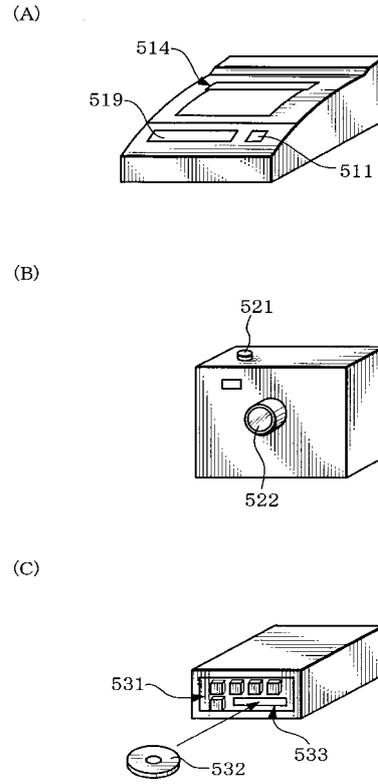
(C) 受信時(USB→パケットバッファ→DMA又はCPU)

	DMA転送使用	CPU転送使用
WP	USB	USB
RP	DMA	CPU

【 図 2 1 】



【 図 2 2 】



フロントページの続き

- (72)発明者 松田 邦昭
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 長尾 謙陽
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

審査官 石井 茂和

- (56)参考文献 特開平10-301898(JP,A)
特開2001-119415(JP,A)
特開2002-055936(JP,A)
特開2002-116998(JP,A)
特開2002-009849(JP,A)
特開2002-091717(JP,A)
特開2002-051100(JP,A)
Zong Liang WU, USB On-The-Go Implementation, 米国, TransDimension, 2001年11月9日, p1-25, URL, http://www.usb.org/data/developers/otg/presentations/london/OTG_implementation.pdf
- (58)調査した分野(Int.Cl.⁷, DB名)
G06F 13/38 310
G06F 13/38 350
WPI(DIALOG)