

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4793798号
(P4793798)

(45) 発行日 平成23年10月12日 (2011.10.12)

(24) 登録日 平成23年8月5日 (2011.8.5)

(51) Int. Cl.

F I

G O 6 F 15/78 (2006.01)

G O 6 F 15/78 5 1 O C

G O 6 F 11/00 (2006.01)

G O 6 F 9/06 6 3 O A

G O 6 F 9/445 (2006.01)

G O 6 F 9/06 6 4 O A

請求項の数 4 (全 11 頁)

(21) 出願番号 特願2000-342849 (P2000-342849)
 (22) 出願日 平成12年11月10日 (2000.11.10)
 (65) 公開番号 特開2002-149626 (P2002-149626A)
 (43) 公開日 平成14年5月24日 (2002.5.24)
 審査請求日 平成19年10月31日 (2007.10.31)
 審判番号 不服2010-6462 (P2010-6462/J1)
 審判請求日 平成22年3月25日 (2010.3.25)

(73) 特許権者 311003743
 オンセミコンダクター・トレーディング・
 リミテッド
 英国領バミューダ・エイチエム 11 ハ
 ミルトン・チャーチストリート2・クラレ
 ンドンハウス・コーダン サービスーズ
 リミテッド 気付
 (74) 代理人 100107906
 弁理士 須藤 克彦
 (72) 発明者 近藤 英雄
 大阪府守口市京阪本通2丁目5番5号 三
 洋電機株式会社内

最終頁に続く

(54) 【発明の名称】 マイクロコンピュータ

(57) 【特許請求の範囲】

【請求項 1】

ホストとマイクロコンピュータ間のデータ送受信のインターフェースを行うU S B インターフェース回路と、書き込み制御プログラムが格納された第1のプログラム領域と書き換え及び読み出し可能なデータを格納する第2のプログラム領域とを有する不揮発性メモリと、前記U S B インターフェース回路によってパラレル変換されたホストからのプログラムデータを一時記憶するテンポラリレジスタと、前記テンポラリレジスタより容量が大きいR A M と、前記書き込み制御プログラムに従って、プログラム命令を実行するC P U と、を備えたマイクロコンピュータであって、

前記R A M と前記テンポラリレジスタとは前記テンポラリレジスタのビット数と等しい信号線によって接続され、前記R A M は前記信号線を介して前記テンポラリレジスタに保持されたデータを一括で保持し、前記R A M に蓄積されたデータが所定量に達すると、前記C P U のバスを経由し、前記R A M に蓄積されたデータを前記第2のプログラム領域に書き込むことを特徴とするマイクロコンピュータ。

【請求項 2】

前記プログラム格納用の不揮発性メモリのアドレスを制御するプログラムカウンタを備え、マイクロコンピュータのリセットに応じて、前記プログラムカウンタの値を前記第1のプログラム領域の先頭アドレスにジャンプさせることを特徴とする請求項1に記載のマイクロコンピュータ。

【請求項 3】

10

20

前記ＵＳＢインターフェース回路から出力される第１のアドレス信号と前記ＣＰＵから出力され

る第２のアドレス信号を選択して前記ＲＡＭのアドレスデコーダに入力するアドレス選択回路を備え、前記ＲＡＭのデータ領域をＵＳＢインターフェース回路及び前記ＣＰＵからアクセス可能としたことを特徴とする請求項2に記載のマイクロコンピュータ。

【請求項４】

前記アドレス選択回路は、ホストからのデータ受信中は前記ＵＳＢインターフェース回路から出力される第１のアドレス信号を選択することを特徴とする請求項3に記載のマイクロコンピュータ。

【発明の詳細な説明】

10

【０００１】

【発明の属する技術分野】

本発明は、ＵＳＢインターフェース回路及びプログラム格納用メモリとしてＥＥＰＲＯＭを内蔵したマイクロコンピュータに関するものであり、特にホスト（例えば、パーソナルコンピュータ）からＵＳＢインターフェース回路を介してＥＥＰＲＯＭへプログラムデータを書き込み、ベリファイ等を行うことを可能にしたマイクロコンピュータに関する。

【０００２】

【従来の技術】

近年、パーソナルコンピュータ等において、周辺デバイスの拡張性の自由度を高めるために、ＵＳＢ（Universal Serial Bus）のサポートが始められている。ＵＳＢはユーザの利便性を考慮して考案されたシリアルインターフェース規格であって、キーボード、マウス、カメラ、プリンタ、スキャナー、スピーカ等の様々な周辺デバイスとパーソナルコンピュータ等との通信に共通に使用できる。

20

【０００３】

図４はＵＳＢを利用したパーソナルコンピュータと周辺デバイスとの接続構成例を示す図である。上位のパーソナルコンピュータ１００とハブ１０１との間はＵＳＢケーブルで接続され、さらにハブ１０１の下位には周辺デバイス１０２～１０５が接続され得る。そして、パーソナルコンピュータ１００によって周辺デバイス１０２～１０５の管理が行われる仕組みになっている。このように、ＵＳＢは多重スター型のネットワーク構造の双方向通信可能なシリアルバスといえる。

30

【０００４】

ここで、ＵＳＢケーブルには４本の信号線が含まれる。その内訳は電源用２本と、データ信号用２本である。データ信号は基本的には差動信号（ D^+ 、 D^- ）として扱われる。また、ＵＳＢを利用したデータ転送は、転送単位がフレームという概念で時間分割され、そのフレームを積み重ねていくことにより行う。１つのフレームはＳＯＦ（Start Of Frame）パケットにより開始する。そして、ホストのパーソナルコンピュータは予めそのフレームの中にスケジューリングされたデータ転送要求トークン（キーボードやカメラからのデータ入力要求や、音声データの出力要求）を順次送出することにより、複数の周辺デバイスとのデータ転送を並行して行う。

【０００５】

40

なお、ＵＳＢに関する技術文献として、例えば「Interface」（１９９７年１月号）、特開平１１－２０５４１２号公報等がある。

【０００６】

ところで、上述した周辺デバイスには、通常、デバイスの動作を制御するためのマイクロコンピュータが搭載される。ここで、マイクロコンピュータにはプログラム格納用メモリとして、電氣的に書き換え、読み出し及び消去可能な不揮発性メモリであるＥＥＰＲＯＭが内蔵されているものとする。このＥＥＰＲＯＭには、上記の機能に加えてプログラムデータを一括消去する機能を備えたフラッシュＲＯＭも含まれる。

【０００７】

従来、上記のＥＥＰＲＯＭにプログラムデータを書き込む場合、２つの方法が行われてい

50

た。以下では、プログラム格納用メモリとしてフラッシュROMを備えたマイクロコンピュータを例として説明する。

【0008】

1つの方法は、図5に示すようにROMライター110を用いてマイクロコンピュータ111に内蔵されたフラッシュROM112に平行にデータを書き込む場合である。例えば、8ビットのマイクロコンピュータ111においては、データ信号線が8本、アドレス信号線が16本、コントロール信号線が3本(チップイネーブル信号、ライトイネーブル信号、リードイネーブル信号)が必要であった。

【0009】

また、フラッシュROM112に書き込むべきプログラムデータはパーソナルコンピュータ100内にヘキサファイル等の所定のファイル形式で存在する場合が多い。そこで、第2の方法としてパーソナルコンピュータ100からフラッシュROM112にプログラムデータを書き込む場合、図6に示すように、パーソナルコンピュータ100に併設されたシリアル通信ユニット115(RS232C等)を用い、シリアル信号線113を介してマイクロコンピュータ111と接続していた。

10

【0010】

また、マイクロコンピュータ111にはSIO(Serial Input/Output)回路114が内蔵されると共に、フラッシュROM112の所定領域にはSIO回路114を動作させるためのSIO制御プログラムが予め書き込まれている。パーソナルコンピュータ100からシリアル信号線113を介してプログラムデータが転送されて来ると、SIO回路114はSIO制御プログラムに従って、フラッシュROM112に書き込み動作を行う。しかしながら、上述したシステム構成ではSIO回路114、シリアル通信ユニット115(RS232C等)という特別な外部回路と通信ソフトを必要としていた。

20

【0011】

【発明が解決しようとする課題】

上述したように、従来マイクロコンピュータに内蔵されたプログラム格納用メモリであるフラッシュROMにプログラムデータを書き込む場合、当該プログラムデータを平行に書き込むと信号線の本数が多くなり、シリアルに書き込む場合には特別の外部回路や通信ソフトを必要としていた。

【0012】

30

そこで本発明は、ホスト(パーソナルコンピュータ)と周辺デバイスとを接続して双方向通信可能な環境において備えられているUSBケーブルをそのまま利用して、ホスト(パーソナルコンピュータ)からマイクロコンピュータに内蔵されたフラッシュROMへプログラムデータの書き込み等を行うと共に、プログラムデータ書き込み後はUSBケーブルを本来の目的である双方向通信に用いるようにし、特別のシリアルラインや外部回路、通信ソフト等を不要とすることを目的としている。

【0013】

【課題を解決するための手段】

本発明のマイクロコンピュータは、上述した課題を解決するために、ホストとマイクロコンピュータ間のデータの送受信のインターフェースを行うUSBインターフェース回路と、電氣的に書き換え及び読み出し可能なプログラム格納用の不揮発性メモリと、前記USBインターフェース回路によって平行変換されたホストからのプログラムデータを一時記憶するデータメモリと、前記不揮発性メモリから読み出されるプログラム命令を実行するCPUと、を備えたマイクロコンピュータであって、前記プログラム格納用の不揮発性メモリは、書き込み制御プログラムが格納された第1のプログラム領域と前記プログラムデータを書き込むべき第2のプログラム領域とを有し、前記第1のプログラム領域に格納された書き込み制御プログラムに従って、前記データメモリに一時記憶されたプログラムデータを第2のプログラム領域に書き込むことを特徴とする。

40

【0014】

これにより、USBケーブルを利用して、ホストからマイクロコンピュータへプログラム

50

データを高速に書き込むことが可能となる。

【 0 0 1 5 】

また、前記プログラム格納用の不揮発性メモリのアドレスを制御するプログラムカウンタを備え、マイクロコンピュータのリセットに応じて、前記プログラムカウンタの値を前記第 1 のプログラム領域の先頭アドレスにジャンプさせることを特徴する。これにより、マイクロコンピュータのリセットに応じて、確実に書き込み制御プログラムをスタートさせることができる。

【 0 0 1 6 】

また、前記データメモリは R A Mであることを特徴とする。汎用のマイクロコンピュータが備えた R A Mを有効利用するためである。

10

【 0 0 1 7 】

また、R A Mは、U S B インターフェース回路及び前記 C P U からアクセス可能であることを特徴とする。R A Mを有効利用するためである。

【 0 0 1 8 】

また、前記 U S B インターフェース回路から出力される第 1 のアドレス信号と前記 C P U から出力される第 2 のアドレス信号を選択して前記 R A Mのアドレスデコードに inputs アドレス選択回路を備え、前記 R A Mのデータ領域を U S B インターフェース回路及び前記 C P U からアクセス可能としたことを特徴とする。

【 0 0 1 9 】

また、前記アドレス選択回路は、ホストからのデータ受信中は前記 U S B インターフェース回路から出力される第 1 のアドレス信号を選択することを特徴とする。これにより、ホストからのデータ受信を優先させ、確実にデータ受信をすることが可能となる。

20

【 0 0 2 0 】

【 発明の実施の形態 】

次に、本発明の実施形態について図面を参照しながら説明する。図 1 は、本発明の実施形態に係るマイクロコンピュータの構成を示すブロック図である。

【 0 0 2 1 】

以下で、マイクロコンピュータ 1 0 は 8 ビット構成として説明する。マイクロコンピュータ 1 0 とパーソナルコンピュータ 1 0 0 とは 1 対の差動信号線によって接続される。そして、U S B データすなわち、U S B 差動信号 (D^+ , D^-) はマイクロコンピュータ 1 0 の端子 P 1 , P 2 を介して入出力される。ここで、U S B 差動信号 (D^+ , D^-) は、U S B 通信プロトコルに従ったシリアルデータ信号である。

30

【 0 0 2 2 】

2 0 は、端子 P 1 , P 2 に接続された入出力回路であって、差動入力バッファ 2 1、入力バッファ 2 2 , 2 3 及び出力バッファ 2 4 , 2 5 から構成されている。ここで、入力バッファ 2 2 , 2 3 は U S B 差動信号 (D^+ , D^-) の状態が (L , L) となる場合を考慮して設けられている。

【 0 0 2 3 】

マイクロコンピュータ 1 0 に内蔵された U S B インターフェース回路 3 0 は、パーソナルコンピュータ 1 0 0 との間のデータ送受信のインターフェースを行うもので、特にデータ受信時は入出力回路 2 0 からの U S B 差動信号 (D^+ , D^-) を受けて各種のデータ処理を行う。U S B インターフェース回路 3 0 は、上記シリアルデータ信号から必要なデータを抽出する。この時、U S B インターフェース回路 3 0 は当該シリアルデータ信号が如何なる転送フォーマットであるかを判別すると共に、エラー信号処理等を行う。また U S B インターフェース回路 3 0 は、上記データ処理が施されたシリアル信号をマイクロコンピュータ 1 0 が処理可能な所定形式の平行信号 (例えば 8 ビット構成) に変換する。

40

【 0 0 2 4 】

さらに、U S B インターフェース回路 3 0 は、平行変換された 8 ビット $\times$ 4 = 3 2 ビットのプログラムデータを一時記憶するテンポラリレジスタ 3 1 及び制御レジスタ (不図示) を備えている。制御レジスタにはパーソナルコンピュータ 1 0 0 がホストとして管理

50

すべき各種データ（デバイスに割り付けられるアドレスデータ等）がセットされる。

【0025】

なお、USBインターフェース回路30はマイクロコンピュータ10からパーソナルコンピュータ100へのデータ送信時には上記と全く逆のデータ処理（パラレルデータからシリアルデータへの変換等）を行っている。

【0026】

RAM40はUSBインターフェース回路30のテンポラリレジスタ31のデータから逐次転送される32ビット単位のプログラムデータを一時記憶するために利用される。そして、USBインターフェース回路30とRAM40との間のデータ転送を行うために、専用の32本の信号線が設けられている。RAM40に蓄積されたプログラムデータが所定量（例えば128バイト）に達すると、128バイトのプログラムデータはマイクロコンピュータ10のバス45を経由してフラッシュROM50へ転送される。

10

【0027】

逆に、フラッシュROM50に書き込まれたプログラムデータをRAM40へ転送し、そのRAM40内に記憶されたプログラムデータをUSBインターフェース回路30のテンポラリレジスタ31へ転送することも可能である。

【0028】

一般に、USB通信によればパーソナルコンピュータ100から大量のデータがデバイス側に送出されるため、デバイス側には特別のデータバッファを設けることが行われる。

【0029】

20

これに対して、本発明ではマイクロコンピュータ10がデータメモリとして本来有しているRAM40をUSB通信によるデータを一時記憶するために利用するという構成を採ることでデータメモリの有効活用を図っている点も特徴である。

【0030】

図2は、RAM40及び周辺回路を示すブロック図である。USBインターフェース回路30からはアドレス信号ADR1、CPU70からはアドレス信号ADR2が出力され、アドレス選択回路80に入力される。アドレス選択回路80はアドレス信号ADR1、ADR2のいずれかを選択してアドレス指定回路81に入力する。

【0031】

そして、アドレス指定回路81の出力はアドレスデコーダ41に入力され、アドレス信号ADR1、ADR2のいずれかに応じて同一のデータ領域がアクセス可能に構成されている。

30

【0032】

上述した構成によれば、RAM40のデータ領域42はアドレス信号ADR2が選択された場合はCPU70がコントロールするデータメモリ領域として利用可能であると共に、アドレス信号ADR1が選択された場合には、USBインターフェース回路30からのプログラムデータ（32ビット単位）を一時記憶するためのデータメモリ領域としても利用可能である。すなわち、RAM40のデータ領域は、CPU70とUSBインターフェース回路30の両方からアクセス可能である。

【0033】

40

ただし、上記のアドレス選択は、パーソナルコンピュータ100とのデータ送受信中については、USBインターフェース回路30からのアドレス信号ADR1を選択するように構成されている。これはパーソナルコンピュータ100からのデータ転送が途中で中断できないというUSBの特性に基づくものである。具体的には、USBインターフェース回路30のテンポラリレジスタ31がフル状態になったことを検知する信号に基づいて、マイクロコンピュータ10はウェイト（待機）状態に自動的に設定される。

【0034】

また、図1において、50はフラッシュROMであり、USB制御プログラム（具体的には、書き込み制御プログラム）が予め書き込まれ、格納された第1のプログラム領域53と、パーソナルコンピュータ100からのプログラムデータがRAM40を経由して書き

50

込まれる第2のプログラム領域52と、に分割されている。ここで、第1のプログラム領域53は書き換えが不能なようにライトプロテクトされている。

【0035】

60はプログラムカウンタであって、その出力はフラッシュROM50のアドレスデコーダ51に印加されている。プログラムカウンタ60の出力値は後に説明するようにUSB通信の状態に応じて、CPUからの命令により所定番地にジャンプする。すなわち、パーソナルコンピュータ100からのプログラムデータの書き込み時には、プログラムカウンタ60は第1のプログラム領域53（書き込み制御プログラム）の先頭アドレスである（FF00）番地にジャンプすると共に、プログラムデータの書き込み後は、第2のプログラム領域52の先頭アドレスである（0000）番地にジャンプする。そして、CPU70は、フラッシュROM50から読み出されるプログラム命令に従ってマイクロコンピュータ10の動作を実行する。

10

【0036】

次に、上述したマイクロコンピュータ10の動作例について図3のフローチャートを参照しながら説明する。まず、最初のステップ200では、マイクロコンピュータ10がUSBケーブルに接続される。このとき、USBケーブルの電源ラインによってマイクロコンピュータ10に電源が投入されることにより、マイクロコンピュータ10がパワーオンリセットによりリセットされる。

【0037】

次に、ステップ201において、プログラムカウンタ60の値は、第1のプログラム領域53（書き込み制御プログラム）の先頭アドレスである（FF00）番地へジャンプする。従って、その後マイクロコンピュータ10は当該書き込み制御プログラムに従って以下の処理を実行する。

20

【0038】

上記のようにステップ201においてUSBケーブルにマイクロコンピュータ10が接続されると、マイクロコンピュータ10側に設けられたプルアップ抵抗を介して、USB差動信号（D⁺、D⁻）が（L、L）から例えば（H、L）へと変化する。パーソナルコンピュータ100はこのUSB差動信号（D⁺、D⁻）の変化により、マイクロコンピュータ10がUSBネットワークに接続されたことを検知し、所定時間後にUSBバスリセット信号を発行する。ステップ202では、このUSBバスリセット信号待ち状態である。

30

【0039】

ステップ203は、USBバスリセット信号を受信したか否かを判定するステップであり、NOと判定された場合には待ち状態を維持する。YESと判定されると、次のステップ204に進む。

【0040】

ステップ204は、エニュメレーション（Enumeration）によるUSBの初期化を行う。ここで、エニュメレーションとは、一般にマイクロコンピュータ10とパーソナルコンピュータ100との間でUSBデータの送受信を行うことが可能な環境設定を行うための一連のソフトウェア処理である。

【0041】

エニュメレーションにより行われる主な処理は、パーソナルコンピュータ100の初期化と、パーソナルコンピュータ100が支配するデバイスにアドレスを割り付ける処理である。後者において、USBインターフェース回路30内の制御レジスタ（アドレスレジスタ）内に、パーソナルコンピュータ100が割り当てた特定のアドレスが記憶される。これにより、マイクロコンピュータ10は、パーソナルコンピュータ100が送信して来たUSBパケット内のアドレスと上記アドレスレジスタ内のアドレスとを照合し、それらが一致した場合にのみ送信されきたUSBデータの処理を行う。

40

【0042】

こうして、USBデータの送受信を行うことが可能な環境設定が終了すると、ステップ205ではパーソナルコンピュータ100からフラッシュROMに書き込むべきプログラム

50

データがUSB差動信号データ (D^+ , D^-) の形で入力されてくる。

【0043】

ステップ206ではこの入力されたUSB差動信号データ (D^+ , D^-) をUSBインターフェース回路30によってデータ処理する。このデータ処理内容は上述した通りであるが、シリアルデータ (8ビット×4) を所定の平行データ (32ビット) に変換するのがその主な処理である。

【0044】

ステップ207では、USBインターフェース回路30からRAM40へ平行変換されたプログラムデータが書き込まれる。そして、RAM40へ書き込まれたプログラムデータ量が所定量 (例えば128バイト) に達すると、この所定量を単位としてRAM40からバス45を介してフラッシュROM50の第2のプログラム領域52へ書き込みが開始される (ステップ208)。これはフラッシュROM50が複数ブロックに分割されており、128バイトをブロックとして構成されていることによる。したがって、RAM40のデータの蓄積量はフラッシュROM50のブロック構成に応じて適宜に選択可能である。

10

【0045】

ここで、実際にはUSBインターフェース回路30からRAM40へ平行変換されたプログラムデータの書き込み動作と、RAM40からフラッシュROM50への書き込み動作は並行して行われるために、高速書き込みが実現される。

【0046】

ステップ208において、フラッシュROMへの書き込みが開始されるがこれには所定の時間を要する。そこで、ステップ209ではマイクロコンピュータ10はソフト的にNACK状態にセットされる。これはUSBパケットのハンドシェイク・パケットの一種であって、ホストであるパーソナルコンピュータ100からのデータを受け付けることができないことを知らせるためにパーソナルコンピュータ100へ返される。

20

【0047】

そして、ステップ210では書き込み終了か否かを判定する。その判定結果がNOであれば、NACK状態を維持する。その判定結果がYESであれば、ACK状態にセットされ、ACKはマイクロコンピュータ10側でデータを受け付け可能であることを知らせるためにパーソナルコンピュータ100へ返される。

30

【0048】

そして、次のステップ212ではフラッシュROM50へのプログラムデータの書き込みが全て終了したかを判定する。その判定結果がNOであれば、ステップ205へ戻り、残余のプログラムデータの書き込みを続行する。ここで、プログラムデータの書き込みはブロック (ページ) 単位 (例えば128バイト) で行われるため、全部のページが書き込まれるまでこの処理は繰り返される。

【0049】

判定結果がYESの場合には、プログラムカウンタ60の値は第2のプログラム領域52の先頭アドレスである (0000) 番地にジャンプする。そして、マイクロコンピュータ10はパーソナルコンピュータ100から供給されたプログラムデータを読み出し、CPU70は解読されたプログラム命令に基づいてマイクロコンピュータ10の動作を実行開始する。

40

【0050】

なお、上述した実施形態では、パーソナルコンピュータ100からマイクロコンピュータ10のフラッシュROM50に対してプログラムデータを書き込む場合について説明したが、フラッシュROM50に書き込まれたプログラムデータを読み出して、パーソナルコンピュータ100へ送り返し、ベリファイを行うことも可能である。その場合には、データ処理の順序は上述したものと逆の順序となる。

【0051】

すなわち、フラッシュROM50から読み出されたプログラムデータはRAM40に一時

50

記憶された後、ＵＳＢインターフェース回路３０へ逐次転送される。そして、ＵＳＢインターフェース回路３０では書き込みの際とは逆のデータ処理を施し、パラレルデータを所定のシリアルデータに変換後、ＵＳＢケーブルを介してパーソナルコンピュータ１００へ送出する。

【００５２】

また、上述した実施形態では初期状態においてプログラムデータを書き込むべき第２のプログラム領域が空状態であるが、これに限定されることなくプログラムのバージョンアップに伴うプログラムの書き換えに対しても同様に適用することができる。

【００５３】

【発明の効果】

本発明によれば、ホスト（例えばパーソナルコンピュータ）と周辺デバイスとの接続のためのＵＳＢケーブルを利用して、ホストからマイクロコンピュータへのプログラム書き込みを高速に行うことが可能となる。

【００５４】

また、プログラムの転送にＵＳＢを利用しているので、特別のシリアルラインや外部回路、通信ソフト等を不要とすることができる。

【００５５】

さらにまた、プログラムデータを一時記憶するためのデータメモリとして、マイクロコンピュータのＲＡＭのデータ領域を利用することにより、特別のデータバッファを必要としないという利点も有している。

【図面の簡単な説明】

【図１】本発明の実施形態に係るマイクロコンピュータを示すブロック図である。

【図２】本発明の実施形態に係るマイクロコンピュータのＲＡＭ及び周辺回路を示すブロック図である。

【図３】本発明の実施形態に係るマイクロコンピュータ１０の動作例を示すフローチャートである。

【図４】ＵＳＢを利用したパーソナルコンピュータと周辺デバイスとの接続構成例を示す図である。

【図５】従来のフラッシュＲＯＭへのプログラムデータ書き込み方法を示す図である。

【図６】従来のフラッシュＲＯＭへのプログラムデータ書き込み方法を示す図である。

【符号の説明】

- １０ マイクロコンピュータ
- ２０ 入出力回路
- ３０ ＵＳＢインターフェース回路
- ４０ ＲＡＭ
- ４５ バス
- ５０ フラッシュＲＯＭ
- ５１ アドレスデコーダ
- ５２ 第２のプログラム領域
- ５３ 第１のプログラム領域
- ６０ プログラムカウンタ
- ７０ ＣＰＵ

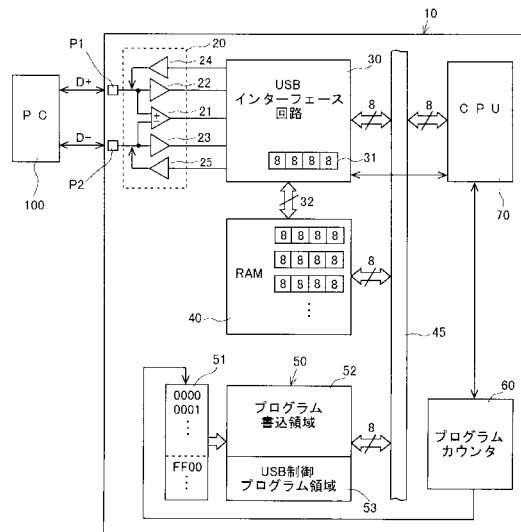
10

20

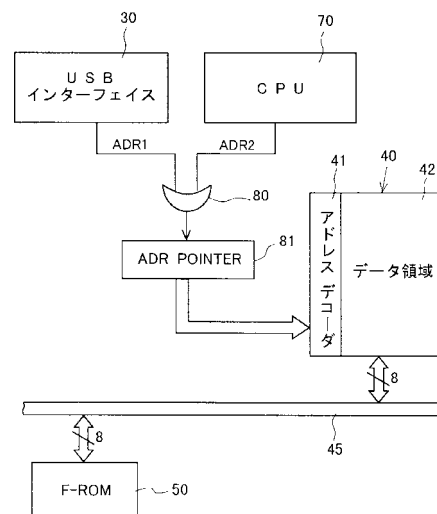
30

40

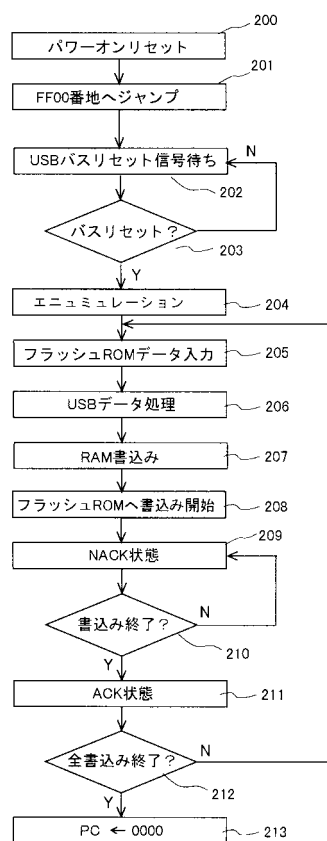
【図 1】



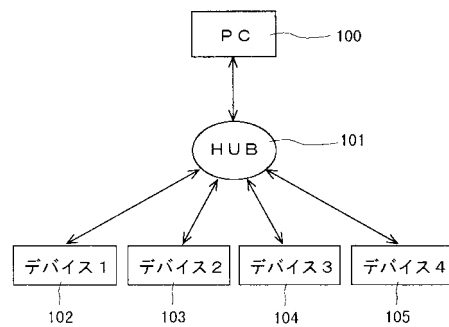
【図 2】



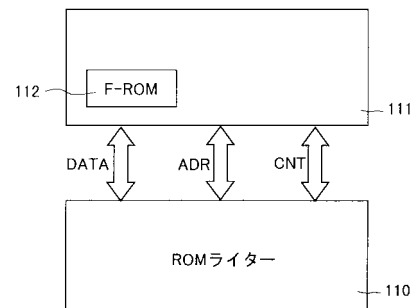
【図 3】



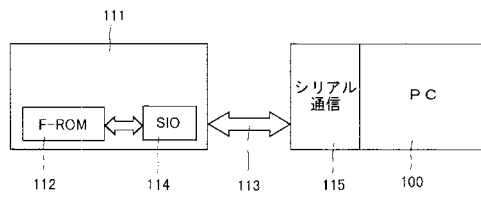
【図 4】



【図 5】



【図 6】



フロントページの続き

合議体

審判長 鈴木 匡明

審判官 ▲吉▼田 美彦

審判官 石井 茂和