

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3955876号

(P3955876)

(45) 発行日 平成19年8月8日(2007.8.8)

(24) 登録日 平成19年5月11日(2007.5.11)

(51) Int. Cl.		F I			
<b>G06F 11/22</b>	<b>(2006.01)</b>	G06F 11/22	340A		
<b>G06F 11/28</b>	<b>(2006.01)</b>	G06F 11/28	L		

請求項の数 17 (全 23 頁)

(21) 出願番号	特願2005-513467 (P2005-513467)	(73) 特許権者	503121103
(86) (22) 出願日	平成16年8月27日(2004.8.27)		株式会社ルネサステクノロジ
(86) 国際出願番号	PCT/JP2004/012350		東京都千代田区大手町二丁目6番2号
(87) 国際公開番号	W02005/022390	(74) 代理人	100089071
(87) 国際公開日	平成17年3月10日(2005.3.10)		弁理士 玉村 静世
審査請求日	平成17年12月26日(2005.12.26)	(72) 発明者	薄井 一雄
(31) 優先権主張番号	特願2003-304276 (P2003-304276)		東京都千代田区丸の内二丁目4番1号 株
(32) 優先日	平成15年8月28日(2003.8.28)		式会社ルネサステクノロジ内
(33) 優先権主張国	日本国(JP)		

審査官 久保 正典

最終頁に続く

(54) 【発明の名称】 マイクロコンピュータ及びシステムプログラムの開発方法

(57) 【特許請求の範囲】

【請求項1】

中央処理装置、デバッグ用インタフェースに利用可能な高速シリアル通信インタフェース回路、及び外部メモリに接続可能な外部バスインタフェース回路を有し、

前記高速シリアル通信インタフェース回路はその内部に複数の入力バッファを有し、相互に一の入力バッファに対する入力動作に並行して他の入力バッファからデータを出力可能とされ、

前記デバッグモードにおいて、前記高速シリアル通信インタフェース回路はシステムプログラムを受信し、受信されたシステムプログラムをメモリアクセス制御信号と共に前記外部バスインタフェース回路から出力可能であることを特徴とするマイクロコンピュータ

10

【請求項2】

前記高速シリアル通信インタフェース回路は、ユニバーサルシリアルバスインタフェース回路であることを特徴とする請求項1記載のマイクロコンピュータ。

【請求項3】

受信されたシステムプログラムを外部バスインタフェースに接続されたメモリに転送制御が可能なダイレクトメモリアクセスコントローラを有することを特徴とする請求項2記載のマイクロコンピュータ。

【請求項4】

前記ダイレクトメモリアクセスコントローラによるシステムプログラムの転送元は前記

20

高速シリアル通信インタフェースの入力バッファであることを特徴とする請求項 3 記載のマイクロコンピュータ。

【請求項 5】

前記高速シリアル通信インタフェース回路の入力バッファに受信されたシステムプログラムを一時的に蓄積可能なランダムアクセスメモリを有し、

前記ダイレクトメモリアクセスコントローラによるシステムプログラムの転送元は前記ランダムアクセスメモリであることを特徴とする請求項 3 記載のマイクロコンピュータ。

【請求項 6】

デバッグ専用低速シリアル通信インタフェース回路を有し、前記デバッグ専用低速シリアル通信インタフェース回路は、デバッグモードにおいて、前記高速シリアル通信インタフェース回路を制御する制御データの入力に利用可能であることを特徴とする請求項 1 記載のマイクロコンピュータ。

【請求項 7】

前記デバッグ専用低速シリアル通信インタフェース回路は、デバッグモードにおいて、前記高速シリアル通信インタフェース回路に代えてシステムプログラムの受信に利用可能であることを特徴とする請求項 6 記載のマイクロコンピュータ。

【請求項 8】

前記デバッグ専用低速シリアル通信インタフェース回路は J T A G に準拠しデータレジスタを有することを特徴とする請求項 6 又は 7 記載のマイクロコンピュータ。

【請求項 9】

トレース制御回路を有し、前記トレース制御回路は前記中央処理装置が前記システムプログラムを実行したときの内部状態をトレース情報として逐次蓄えることを特徴とする請求項 1 又は 6 記載のマイクロコンピュータ。

【請求項 10】

前記トレース情報の外部出力に前記高速シリアル通信インタフェース回路を利用可能であることを特徴とする請求項 9 記載のマイクロコンピュータ。

【請求項 11】

ホストコンピュータと、エミュレータと、ターゲットデバイスとを用いて、ターゲットデバイスが実行するシステムプログラムを開発する方法であって、

前記エミュレータによる処理として、ホストコンピュータが高速シリアル通信で出力するシステムプログラムを 2 面バッファの一方のバッファに蓄積する第 1 処理と、前記 2 面バッファの他方のバッファに蓄積されたシステムプログラムを前記第 1 処理に並行してターゲットデバイスに低速シリアル通信で送信する第 2 処理と、ターゲットデバイスとの間の前記低速シリアル通信のハンドシェイク制御を行なう第 3 処理を含み、

前記第 2 処理では、前記バッファから出力するシステムプログラムを前記一つのバッファの記憶容量以上の F I F O バッファを経由してターゲットデバイスに低速シリアル通信で送信し、

前記第 3 処理では、ターゲットデバイスからの送信許可に回答して F I F O バッファからターゲットデバイスへの送信を行ない、F I F O バッファのフル状態に回答して前記バッファから F I F O バッファへの転送を抑制することを特徴とするシステムプログラムの開発方法。

【請求項 12】

ユーザモードとデバッグモードを有するマイクロコンピュータであって、

中央処理装置と、ユニバーサルシリアルバスインタフェース回路と、第 1 のデバッグ用制御プログラムを保有する R O M と、R A M と、外部バスインタフェース回路とを備え、

前記ユニバーサルシリアルバスインタフェース回路は、前記デバッグモードで利用可能にされる所定のエンドポイントバッファ回路を有し、前記所定のエンドポイントバッファ回路は並列動作可能な一対のバッファを有し、前記一対のバッファは相互に一方が入力動作可能にされるのに並行して他方が出力動作可能とされ、

前記中央処理装置は、パワーオンリセット時に前記デバッグモードが指定されていると

10

20

30

40

50

き、前記第 1 のデバッグ用制御プログラムを実行して、前記ユニバーサルシリアルバスインタフェース回路を動作可能に初期化し、第 2 のデバッグ用制御プログラムを前記ユニバーサルシリアルバスインタフェース回路で受信し、受信した第 2 のデバッグ用制御プログラムを前記 R A M に格納し、R A M に格納した第 2 のデバッグ用制御プログラムの実行に移行するマイクロコンピュータ。

【請求項 1 3】

バッファ R A M とダイレクトメモリアクセスコントローラと更に有し、

中央処理装置は、前記第 2 のデバッグ用制御プログラムにしたがって、ユニバーサルシリアルバスインタフェース回路が受信したダウンロード要求コマンドにตอบสนองして、前記ダイレクトメモリアクセスコントローラに、ユニバーサルシリアルバスインタフェース回路が受信したプログラムを前記バッファ R A M に転送させる請求項 1 2 記載のマイクロコンピュータ。

10

【請求項 1 4】

前記中央処理装置は、前記第 2 のデバッグ用制御プログラムにしたがって、ユニバーサルシリアルバスインタフェース回路が受信した転送要求コマンドにตอบสนองして、前記ダイレクトメモリアクセスコントローラに、前記バッファ R A M に転送されたプログラムを外部バスインタフェース回路を介して外部に転送制御させる請求項 1 3 記載のマイクロコンピュータ。

【請求項 1 5】

中央処理装置は、前記第 2 のデバッグ用制御プログラムの実行状態において、モード制御コマンドにตอบสนองしてユーザモードに移行し、

20

前記ユーザモードにおいて中央処理装置は外部バスインタフェース回路を介して命令をフェッチする請求項 1 2 記載のマイクロコンピュータ。

【請求項 1 6】

中央処理装置と、ユニバーサルシリアルバスインタフェース回路と、第 1 のデバッグ用制御プログラムを保有する R O M と、バッファ R A M と、外部インタフェース回路とを備え、

前記ユニバーサルシリアルバスインタフェース回路は、所定のエンドポイントバッファ回路を有し、前記所定のエンドポイントバッファ回路は並列動作可能な一対のバッファを有し、前記一対のバッファは相互に一方が入力動作可能にされるのに並行して他方が出力動作可能とされ、

30

前記中央処理装置は、パワーオンリセット時に前記第 1 のデバッグ用制御プログラムを実行して、前記ユニバーサルシリアルバスインタフェース回路を動作可能に初期化し、第 2 のデバッグ用制御プログラムを前記ユニバーサルシリアルバスインタフェース回路で受信し、受信した第 2 のデバッグ用制御プログラムを前記バッファ R A M に格納し、バッファ R A M に格納した第 2 のデバッグ用制御プログラムを前記外部インタフェース回路を介して出力させるマイクロコンピュータ。

【請求項 1 7】

ダイレクトメモリアクセスコントローラを更に有し、

前記ダイレクトメモリアクセスコントローラは、前記中央処理装置による転送制御条件にしたがって、前記バッファ R A M から第 2 のデバッグ用制御プログラムを前記外部インタフェース回路を介して外部に転送する請求項 1 6 記載のマイクロコンピュータ。

40

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

本発明は、マイクロコンピュータ、特にデバッグサポート機能を有するマイクロコンピュータに関し、更にはマイクロコンピュータを動作させるシステムプログラムの開発方法に関する。

【背景技術】

【0 0 0 2】

50

デバッグサポート機能を有するマイクロコンピュータはデバッグに際してエミュレータもしくはホストコンピュータと通信を行なうデバッグ用インタフェースを有する。このデバッグ用インタフェースには、例えばJTAG (Joint Test Action Group, IEEE Std 1149.1, IEEE Standard Test Access Port and Boundary-Scan Architecture) プロトコルに準拠したシリアル入出力インタフェースが用いられている。特許文献1にもそのようなデバッグ用インタフェースを備えたマイクロコンピュータの記載がある。

【0003】

【特許文献1】特開2002-202900号公報(段落0017)

10

【発明の開示】

【発明が解決しようとする課題】

【0004】

本発明者はデバッグサポート機能(オンチップデバッグ機能とも称する)を有するマイクロコンピュータをシステムデバッグするとき(デバッグ対象マイクロコンピュータをターゲットマイクロコンピュータと称する)、そのターゲットプログラムをホストコンピュータからターゲットシステムのプログラムメモリにダウンロードする機能におけるデータ転送の高速化について検討した。例えば、ターゲットマイクロコンピュータはJTAG準拠のデバッグ用シリアルインタフェース回路を有する。USB(Universal Serial Bus)インタフェース回路を持つパーソナルコンピュータ(PC)にUSBケーブルでエミュレータを接続し、エミュレータは専用のユーザーインタフェースケーブルでターゲットシステム(ユーザーシステムとも称する)に搭載されたターゲットマイクロコンピュータのデバッグ用シリアルインタフェースに接続する。ターゲットマイクロコンピュータはオンチップデバッグ機能を有する。オンチップデバッグ機能を有するターゲットマイクロコンピュータは、ユーザモードの他に、ターゲットプログラムの開発を支援するデバッグモードを有する。前記ユーザモードではターゲットシステムのためにユーザが開発するユーザプログラムとしてのシステムプログラム(ターゲットプログラムとも称する)を実行する。前記デバッグモードでは、ユーザプログラムの実行が停止されているとき、主にプログラムデバッグ支援用のプログラム(デバッグ支援プログラムとも称する)を実行する。デバッグ支援プログラムは、ターゲットマイクロコンピュータの電源投入毎にホストコンピュータ上のエミュレータソフトより転送され、ターゲットマイクロコンピュータ内部のデバッグ用アドレス空間上に書き込みされ、ホストコンピュータ上のエミュレータソフトウェアとリンクされる。ホストコンピュータから供給されるユーザプログラムは、ターゲットマイクロコンピュータがユーザモードでデバッグ支援プログラムを実行することによって、ターゲットシステム上の所定メモリ空間に書き込まれる。エミュレータとターゲットマイクロコンピュータとの間の通信には前記JTAG準拠のデバッグ用シリアルインタフェース回路が用いられる。このデバッグ用シリアルインタフェース回路は、所定バイト単位でクロック同期のシリアル転送を行なうため、転送スピードは前記クロック周波数に比例し、USBインタフェース回路に比べて低速である。例えば、USB規格1.1でも転送処理能力はフルスピードで1200キロバイト(KB)/秒であるが、JTAG準拠のインタフェースでは4バイトのような1回の最大データ転送毎にターゲットマイクロコンピュータからのアクセス許可を示すステータスを取得して次の転送データをセットするというデータ転送手順を必要とし、それによるオーバーヘッド故に、同期クロック周波数を上げて230KB/秒のような転送速度限界を生ずる。このような転送速度の相違によって、比較的容量の大きなターゲットプログラムのロード転送に長い時間を必要とし、システムデバッグ効率を低下させ、更にはターゲットプログラムの開発期間短縮を阻む要因の一つになっていることが本発明者によって明らかにされた。

20

30

40

【0005】

本発明の目的は、ターゲットマイクロコンピュータをシステムデバッグするとき、デバッグ対象のシステムプログラムをホストコンピュータからターゲットシステムにダウンロ

50

ードするときのデータ転送を高速化することにある。

【0006】

本発明の別の目的は、エミュレータを利用してシステムプログラムを開発するとき、デバッグ対象のシステムプログラムをホストコンピュータからターゲットシステムにダウンロードするときのデータ転送の高速化という点において、システムプログラムの開発期間の短縮に寄与することにある。

【0007】

本発明の前記並びにその他の目的と新規な特徴は本明細書の記述及び添付図面から明らかになるであろう。

【課題を解決するための手段】

【0008】

本願において開示される発明のうち代表的なものの概要を簡単に説明すれば下記の通りである。

【0009】

〔1〕本発明に係るマイクロコンピュータは、中央処理装置、デバッグ用インタフェースに利用可能な高速シリアル通信インタフェース回路(3)、及び外部メモリに接続可能な外部バスインタフェース回路(5)を有する。前記高速シリアル通信インタフェース回路はその内部に複数の入力バッファ(EP1, EP2)を有し、相互に一の入力バッファに対する入力動作に並行して他の入力バッファからデータを出力可能とされる。前記デバッグモードにおいて、前記高速シリアル通信インタフェース回路はシステムプログラムを受信し、受信されたシステムプログラムをメモリアクセス制御信号と共に前記外部バスインタフェース回路から出力可能である。前記高速シリアル通信インタフェース回路は、例えばユニバーサルシリアルバスインタフェース回路である。システムデバッグに際して前記高速シリアル通信インタフェース回路にホストコンピュータを直結することにより、ターゲットプログラムをホストコンピュータからターゲットシステムにダウンロードするときのデータ転送を高速化することができる。特に前記高速シリアル通信インタフェース回路は交互に入出力動作を切換えて並列動作可能な2面バッファを備えるという点でデータ転送の更なる高速化が実現される。ターゲットプログラムをホストコンピュータからターゲットシステムにダウンロードするときのデータ転送の高速化という点において、ターゲットプログラムの開発期間の短縮に寄与する。

【0010】

本発明の望ましい形態では、受信されたシステムプログラムを外部バスインタフェース回路に接続されたメモリに転送制御可能なダイレクトメモリアクセスコントローラを有する。中央処理装置の負担軽減とデータ転送の更なる高速化に好都合である。前記ダイレクトメモリアクセスコントローラによるシステムプログラムの転送元は、例えば前記高速シリアル通信インタフェース回路の入力バッファである。前記高速シリアル通信インタフェース回路の入力バッファに受信されたシステムプログラムを一時的に蓄積可能なランダムアクセスメモリ(7)を備える場合には、前記ダイレクトメモリアクセスコントローラによるシステムプログラムの転送元は前記ランダムアクセスメモリとしてよい。前記高速シリアル通信インタフェース回路の入力バッファ(EP1, EP2)とランダムアクセスメモリ(7)で多段バッファを構成してシステムプログラムを受け取るから、高速シリアル通信インタフェース回路による受信処理と受信データに対する転送処理との速度差に対して更に余裕ができる。

【0011】

本発明の望ましい形態では、デバッグ専用低速シリアル通信インタフェース回路(8)を有し、前記デバッグ専用低速シリアル通信インタフェース回路は、デバッグモードにおいて、前記高速シリアル通信インタフェース回路を制御する制御データの入力に利用可能である。デバッグ専用低速シリアル通信インタフェース回路は例えばJTAG準拠のシリアルインタフェース回路であり、中央処理装置の制御を必要とせずにインタフェース動作を行なうことができるようになっている。前記デバッグ専用低速シリアル通信インタフェ

10

20

30

40

50

ース回路は、例えばデバッグモードにおいて、前記高速シリアル通信インタフェース回路に代えてシステムプログラムの受信に利用可能である。

【 0 0 1 2 】

デバッグサポート機能の充実という観点より、トレース制御回路を内蔵してよい。前記トレース制御回路は前記中央処理装置が前記システムプログラムを実行したときの内部状態をトレース情報として逐次蓄え、蓄えたトレース情報は前記システムプログラムの実行停止後に外部に出力される。前記トレース情報の外部出力に前記高速シリアル通信インタフェース回路を利用可能である。

【 0 0 1 3 】

〔 2 〕本発明に係るシステムプログラムの開発方法は、ホストコンピュータ（ 2 5 ）と、エミュレータ（ 3 5 ）と、ターゲットデバイス（ 3 3 ）とを用いて、ターゲットデバイスが実行するシステムプログラムを開発する方法であって、前記エミュレータによる処理として、ホストコンピュータが高速シリアル通信で出力するシステムプログラムを 2 面バッファの一方のバッファに蓄積する第 1 処理と、 2 面バッファの他方のバッファに蓄積されたシステムプログラムを前記第 1 処理に並行してターゲットデバイスに低速シリアル通信で送信する第 2 処理と、ターゲットデバイスとの間の前記低速シリアル通信のハンドシェイク制御を行なう第 3 処理を含む。 2 面バッファの利用により、ターゲットプログラムをホストコンピュータからターゲットシステムにダウンロードするときのデータ転送を高速化することができるという点において、ターゲットプログラムの開発期間を短縮することができる。

【 0 0 1 4 】

本発明の具体的な形態として、前記第 2 処理では、前記バッファから出力するシステムプログラムを前記一つのバッファの記憶容量以上の F I F O バッファを経由してターゲットデバイスに低速シリアル通信で送信し、前記第 3 処理では、ターゲットデバイスからの送信許可にตอบสนองして F I F O バッファからターゲットデバイスへの送信を行ない、 F I F O バッファのフル状態にตอบสนองして前記バッファから F I F O バッファへの転送を抑制する。低速シリアル通信に F I F O バッファを用いることにより、第 1 処理から第 2 処理に渡されるデータの滞留による第 1 処理の中断を減らすことができ、この点においても、上記データ転送を高速化することができる。

【 0 0 1 5 】

〔 3 〕本発明に係るマイクロコンピュータ（ 1 ）は、ユーザモードとデバッグモードを有し、中央処理装置（ 2 ）と、ユニバーサルシリアルバスインタフェース回路（ 3 ）と、第 1 のデバッグ用制御プログラムを保有する R O M（ 7 1 ）と、 R A M（ 7 ）と、外部バスインタフェース回路（ 5 ）とを備える。前記ユニバーサルシリアルバスインタフェース回路は、前記デバッグモードで利用可能にされる所定のエンドポイントバッファ回路（ 2 0 ）を有し、前記所定のエンドポイントバッファ回路は並列動作可能な一対のバッファ（ E P 1 , E P 2 ）を有し、前記一対のバッファは相互に一方が入力動作可能にされるのに並行して他方が出力動作可能とされる。ここで対象とするマイクロコンピュータはターゲットマイクロコンピュータであり、ユニバーサルシリアルバスインタフェース回路を介して直接デバッグ支援用のホストコンピュータに接続する場合を想定する。

【 0 0 1 6 】

前記中央処理装置は、パワーオンリセット時に前記デバッグモードが指定されているとき、前記第 1 のデバッグ用制御プログラムを実行して、前記ユニバーサルシリアルバスインタフェース回路を動作可能に初期化し、第 2 のデバッグ用制御プログラムを前記ユニバーサルシリアルバスインタフェース回路で受信し、受信した第 2 のデバッグ用制御プログラムを前記 R A M に格納し、 R A M に格納した第 2 のデバッグ用制御プログラムの実行に移行する。これにより、マイクロコンピュータの電源投入毎に例えばホストコンピュータ上のエミュレータソフトより転送される第 2 のデバッグ用制御プログラムを高速にダウンロードことができ、ホストコンピュータ上のエミュレータソフトウェアとリンクしたデバッグ制御可能な状態の立ち上げを高速化することができる。

## 【 0 0 1 7 】

本発明の具体的な形態として、バッファRAMとダイレクトメモリアクセスコントローラと更に有し、中央処理装置は、前記第2のデバッグ用制御プログラムにしたがって、ユニバーサルシリアルバスインタフェース回路が受信したダウンロード要求コマンドにตอบสนองして、前記ダイレクトメモリアクセスコントローラに、ユニバーサルシリアルバスインタフェース回路が受信したユーザプログラムを前記バッファRAMに転送させる。デバッグ制御可能な状態においてユーザプログラムのダウンロードを高速化することができる。更に、本発明の望ましい形態として、前記中央処理装置は、前記第2のデバッグ用制御プログラムにしたがって、ユニバーサルシリアルバスインタフェース回路が受信した転送要求コマンドにตอบสนองして、前記ダイレクトメモリアクセスコントローラに、前記バッファRAMに転送されたプログラムを外部バスインタフェース回路を介して外部のプログラムメモリに転送制御させる。この後、前記中央処理装置は、前記第2のデバッグ用制御プログラムの実行状態において、モード制御コマンドにตอบสนองしてユーザモードに移行し、前記ユーザモードにおいて中央処理装置は外部バスインタフェース回路を介して前記プログラムメモリなどから命令をフェッチして、ユーザプログラムを実行する。ユーザプログラムの実行状態をトレースし、それを解析したりすることによってユーザプログラムに対するデバッグが行われる。

10

## 【 0 0 1 8 】

〔 4 〕別の観点による本発明のマイクロコンピュータ(40)は、中央処理装置(44)と、ユニバーサルシリアルバスインタフェース回路(48)と、第1のデバッグ用制御プログラムを保有するROM(45)と、バッファRAM(47)と、外部インタフェース回路(41)とを備え、前記ユニバーサルシリアルバスインタフェース回路は、所定のエンドポイントバッファ回路(20)を有し、前記所定のエンドポイントバッファ回路は並列動作可能な一対のバッファ(EP1, EP2)を有し、前記一対のバッファは相互に一方が入力動作可能にされるのに並行して他方が出力動作可能とされる。前記中央処理装置は、パワーオンリセット時に前記第1のデバッグ用制御プログラムを実行して、前記ユニバーサルシリアルバスインタフェース回路を動作可能に初期化し、第2のデバッグ用制御プログラムを前記ユニバーサルシリアルバスインタフェース回路で受信し、受信した第2のデバッグ用制御プログラムを前記バッファRAMに格納し、バッファRAMに格納した第2のデバッグ用制御プログラムを前記外部インタフェース回路を介して出力させる。ここで対象とするマイクロコンピュータはターゲットマイクロコンピュータとデバッグ支援用のホストコンピュータとの間に配置されるエミュレータ(33)の制御用マイクロコンピュータとされる。

20

30

## 【 0 0 1 9 】

本発明の具体的な形態として、マイクロコンピュータはダイレクトメモリアクセスコントローラ(46)を更に有し、前記ダイレクトメモリアクセスコントローラは、前記中央処理装置による転送制御条件にしたがって、前記バッファRAMから第2のデバッグ用制御プログラムを前記外部インタフェース回路を介して外部に転送する。

## 【 発明の効果 】

## 【 0 0 2 0 】

本願において開示される発明のうち代表的なものによって得られる効果を簡単に説明すれば下記の通りである。

40

## 【 0 0 2 1 】

すなわち、ターゲットマイクロコンピュータをシステムデバッグするとき、システムプログラムをホストコンピュータからターゲットシステムにダウンロードするときのデータ転送を高速化することができる。要するに、システムデバッグにおいて、システムプログラムダウンロード機能の性能を向上させることができる。

## 【 0 0 2 2 】

エミュレータを利用してシステムプログラムを開発するとき、システムプログラムをホストコンピュータからターゲットシステムにダウンロードするときのデータ転送の高速化

50

という点において、システムプログラムの開発期間の短縮に寄与することができる。

【図面の簡単な説明】

【0023】

- [図1] 本発明に係るマイクロコンピュータを例示するブロック図である。
- [図2] USBIF内蔵のマイクロコンピュータを搭載したターゲットシステムとホストコンピュータとの接続形態を示す説明図である。
- [図3] USBIFを備えていないマイクロコンピュータを搭載したターゲットシステムとホストコンピュータとの接続形態を示す説明図である。
- [図4] 図3のマイクロコンピュータとエミュレータの一例を示すブロック図である。
- [図5] FPGAで構成されるJTAGインタフェースロジックの詳細を示す論理回路図である。 10
- [図6] 図5の構成による動作を示すタイミングチャートである。
- [図7] RAMの2面バッファBUF1, BUF2を切り換えてデータをFIFO TDOに転送する制御手順を示すフローチャートである。
- [図8] 図5と図9の夫々の場合におけるプログラムダウンロードの動作タイミングの相違を示すタイミングチャートである。
- [図9] 図5の比較例に係るエミュレータを例示するブロック図である。
- [図10] 図5の例と図9の比較例のそれぞれにおいてホストコンピュータからユーザープログラムファイルをダウンロードするときのダウンロード性能を例示する説明である。
- [図11] 本発明に係るマイクロコンピュータの更に具体的な例を示すブロック図である 20
- [図12] マイクロコンピュータによるオンチップデバッグの動作タイミングを例示するタイミングチャートである。
- [図13] USBの通信フォーマットの説明図である。
- [図14] ホストコンピュータとUSBインタフェース回路との通信ハンドシェック制御の基本形を例示するフローチャートである。
- [図15] ユーザープログラムの実行中に強制ブレークさせる場合のホストコンピュータとUSBインタフェース回路との間でハンドシェック制御内容を例示するフローチャートである。
- [図16] ホストコンピュータからUSBインタフェース回路へのソフトウェアのダウンロードの制御内容を例示するフローチャートである。 30
- [図17] 図1のマイクロコンピュータの変形例を示すブロック図である。
- [図18] 図17のマイクロコンピュータの変形例を示すブロック図である。

【符号の説明】

【0024】

- 1 マイクロコンピュータ
- 2 CPU
- 3 USBインタフェース回路
- 4 外部メモリ
- 5 外部バスインタフェース回路 40
- 6 DMAC
- 7 RAM
- 8 JTAG回路
- 9 トレース回路
- 11 エミュレーションRAM
- EP1, EP2 2面バッファ
- 20 バッファ部
- 25 ホストコンピュータ
- 30 ターゲットシステム
- 33 マイクロコンピュータ 50

34 ターゲットシステム  
 35 エミュレータ  
 47 RAM  
 BUF1, BUF2 2面バッファ  
 FIFO TDO  
 48 USBインタフェース回路  
 71 ROM  
 73 パワーオンリセット回路  
 74 システムコントローラ

【発明を実施するための最良の形態】

10

【0025】

図1には本発明に係るマイクロコンピュータ1が例示される。マイクロコンピュータ1は、単結晶シリコンのような1個の半導体基板(半導体チップ)に相補型MOS集積回路製造技術等によって形成される。マイクロコンピュータ1は、中央処理装置(CPU)2、デバッグ用インタフェースに利用可能な高速シリアル通信インタフェース回路としてのUSBインタフェース回路(USBIF)3、外部メモリ(EXMRY)4に接続可能な外部バスインタフェース回路(EXIF)5、ダイレクトメモリアクセスコントローラ(DMAC)6、ランダムアクセスメモリ(RAM)7、デバッグ専用低速シリアル通信インタフェースとしてのJTAGインタフェース回路(JTAGIF)8、トレース制御回路(TRCNT)9、トレース情報の外部出力制御用のAUD(アドバンスト・ユーザ・デバッグ)インタフェース回路(AUDIF)10、及びエミュレーションRAM(EMMRY)11を有し、特に制限されないが、それらは内部バス(IBUS)12を共有する。尚、内部バス12にはタイマカウンタなどその他の回路が接続されていてよい。前期RAM7は転送データなどを一時的に保持するバッファRAMなどとして利用される。

20

【0026】

前記USBIF3は、例えばUSB2.0規格に準拠し、USBバッファ部(BEP)20とUSBインタフェース制御部(UCNT)21を有する。USBバッファ部20は夫々512バイトのUSBバッファEP1、EP2によるダブルバッファ構造を持つ。USBバッファEP1、EP2は例えばFIFOによって構成される。UCNT21は所謂USBデバイスコントロールとデータ転送制御を行なう。UCNT21はUSBホストとして例えば図1のホストコンピュータ(パーソナルコンピュータ)に搭載されるUSBホストにUSBケーブル23で接続され、USBデバイスコントロールとして、USBホストからのコマンドに回答して所定のプロトコルでデータのシリアル送受信制御を行なう。USBホストからの受信データはバッファ部20に送られ、USBホストへの送信データはバッファ部20から供給される。UCNT21はデータ転送制御としてバッファ部20に対するリード・ライト制御と、DMAC6に対する転送要求の制御を行なう。特にバッファ部20に対するリード・ライト制御では、相互に一のUSBバッファに対する入力動作(USBホストからの受信データ入力動作)に並行して他のUSBバッファからデータ出力動作(DMA転送のための内部バス12への出力動作)を可能とする。

30

【0027】

40

DMAC7はCPU2により、転送元アドレス、転送先アドレスなどのデータ転送制御条件が設定され、USBIF3等からのDMA転送要求に回答して、転送元から転送動作へのデータ転送を制御する。データ転送モードとしてデュアルアドレッシングモード及びシングルアドレッシングモードの双方をサポートする。例えば、USBバッファEP1、EP2とEMMRY11との間のシングルアドレスによるデータ転送、USBバッファEP1、EP2とRAM7との間のシングルアドレスによるデータ転送、RAM7とEXMRY4との間のデュアルアドレスによるデータ転送を制御することができる。

【0028】

JTAGIF8は受信データを入力する入力レジスタTDI、送信データを出力する出力レジスタTDO、前記レジスタTDIと内部バス12を接続するデータレジスタSDD

50

R、図示を省略するコマンドレジスタ、JTAG制御回路(JCNT)24を有する。J CNT 24はクロック信号TCKに同期してレジスタTDOからのシリアル出力とレジスタTDIへのシリアル入力を制御する。その制御には所謂TAP(テスト・アクセス・ポート)コントロールを用い、クロック信号TCKに同期してシリアル入力される図示を省略する1ビットのモードセレクト信号のパターンによって制御を行なう。外部からJTAGIF8へのデータ入力はレジスタTDOから外部にアクセス許可ビットが出力される毎に可能にされる。例えばJTAGIF8は4バイト受信毎に外部にアクセス許可ビットを出力する。この意味において、JTAGIF8によるシリアル通信速度は遅い。規格ではTCK周波数は数十MHzであり、USB2.0規格では転送処理能力がハイスピード時480MB/秒であるのに対して低速である。

10

**【0029】**

前記CPU2は図示を省略する命令制御部と演算部を備える。命令制御部は命令フェッチを制御すると共にフェッチした命令のデコードを行なう。演算部は命令のデコード信号及び命令で指定されるオペランドを用いてデータ演算やアドレス演算を行なって命令を実行する。マイクロコンピュータ1は通常モードとしてのユーザモードの他に、ターゲットプログラムの開発を支援するデバッグモードを有する。前記デバッグモードの指定は、リセット時にモード端子から指定し、或いはユーザモードにおいてブレーク割り込みで指定することができる。前記ユーザモードではEXMR Y4に格納されているシステムプログラム(ユーザプログラムとも称する)を実行する。前記デバッグモードでは、ユーザプログラムの実行が停止されているとき、主にプログラムデバッグ支援用のプログラム(デバ

20

**【0030】**

図1では、ターゲットシステムのマイクロコンピュータ1とホストコンピュータ25との間の通信にはUSBIF3が利用可能にされる。JTAGIF8を利用することも可能であるが、通信速度の点でUSBIF3の方が優れているので、USBIF3を利用するのが得策である。特にUSBIF3は交互に入出力動作を切換えて並列動作可能な2面バッファEP1, EP2を備えるという点でデータ転送の更なる高速化が実現される。ユーザプログラムをホストコンピュータ25からターゲットシステムにダウンロードするときのデータ転送の高速化という点において、ユーザプログラムの開発期間を短縮可能になる。

30

**【0031】**

USBIF3で受信されたシステムプログラムをEXIF5に接続されたEXMR Y4に転送するには、DMAC6を用いればよい。前記DMAC6によるシステムプログラムの転送元は、例えば前記USBIF3の入力バッファEP1, EP2である。前記USBIF3の入力バッファEP1, EP2に受信されたシステムプログラムをRAM7に一時的に蓄積する場合には、前記DMAC6により先ずバッファEP1, EP2からRAM7に転送し、タイミングを見計らって、RAM7からEXMR Y4にシステムプログラムを転送すればよい。前記USBIF3の入力バッファEP1, EP2とRAM7で多段バッファを構成してシステムプログラムを受けることにより、USBIF3による受信処理と受信データに対する外部への転送処理との速度差に対して更に余裕ができる。

40

**【0032】**

前記JTAGIF8は、デバッグモードにおいて、前記USBIF3を制御する制御データの入力に利用可能である。JTAGIF8は、例えばデバッグモードにおいて、前記

50

USBIF3に代えてシステムプログラムの受信に利用可能である。その必要性は、ユーザモードにおいて前記USBIF3の使用が予約されていて、プログラムダウンロードに利用することができない環境にある場合などである。前記USBIF3は複数転送チャネルを備える構成であってもよい。その一つをオンチップデバック専用にすれば、プログラムダウンロードをJTAGIF8による代替通信で行なうことを一切要しないから、そのデータ転送の高速化を保証することができる。

【0033】

前記TRCNT9はCPU2がユーザプログラムを実行しているとき、CPU2の内部状態をトレースバス13を介して順次格納する回路である。格納場所はFIFOバッファ(FBUF)26とされ、FIFOバッファ26に対するアドレス制御はアドレスカウンタ(ACOUNT)27が行なう。FIFOバッファ26に蓄えられたトレース情報は、デバッグモードにおいて前記USBIF3又はJTAGIF8によってホストコンピュータ25に転送可能である。但しFIFOバッファ26の容量は小さいので、大量のデータをトレースするには不向きである。大量のデータをトレース情報として収集する場合には前記AUDIF10を用いる。内部バス12のアドレス情報及びデータ情報等をバスアクセスサイクル毎にトレースバッファ(TRBUF)28に蓄え、TRBUF28のデータをクロック信号AUDCLKに同期して外部に出力すればよい。AUDATAは出力データ、AUDSYNCはデータ出力同期信号である。

【0034】

または、FIFOバッファ26に蓄えられたトレース情報をRAM7やEXMRY4へ一時的に格納し、USBIF3を用いて一括で出力することも可能である。

【0035】

図2にはマイクロコンピュータ1を搭載したターゲットシステム30とホストコンピュータ25との接続形態を示す。ターゲットシステム30にはマイクロコンピュータ1のUSBIF3に接続するUSBコネクタ31が設けられ、このUSBコネクタ31とホストコンピュータ25のUSBコネクタをUSBケーブル23で直結すればよい。マイクロコンピュータ1はデバッグ機能を有するから、ホストコンピュータ25とマイクロコンピュータ1の間にエミュレータを介在させることなく、ターゲットプログラムとしてのシステムプログラムとデバッグ支援プログラムのダウンロードを行なうことができ、その後のユーザモードでは、システムプログラムを実行しながらトレース情報を収集する。システムプログラムの実行がブレークされてデバッグモードに遷移されたとき、トレース情報を参照したりしてターゲットシステムに対する評価及びシステムプログラムの修正等が行なわれる。

【0036】

図3には別の例として前記USBIF3を備えていないマイクロコンピュータ33を搭載したターゲットシステム34とホストコンピュータ25との接続形態を示す。ターゲットシステム34とホストコンピュータ25の間にはエミュレータ35が配置される。エミュレータ35とホストコンピュータ25はUSBケーブル23で接続される。ターゲットシステム34とエミュレータ35はJTAGインタフェースフェースケーブル36で接続される。

【0037】

図4には前記マイクロコンピュータ33とエミュレータ35の一例が示される。マイクロコンピュータ33は図1のマイクロコンピュータ1に対してUSBIF3を備えていない点が相異される。同一機能を有する回路要素には同一符号を付してその詳細な説明は省略する。

【0038】

エミュレータ35は、マイクロコンピュータ40、フィールド・プログラマブル・ゲート・アレイ(FPGA)41、シンクロナス・スタティック・ランダム・アクセス・メモリ(SSRAM)42から成る。

【0039】

10

20

30

40

50

マイクロコンピュータ40はCPU44、ROM45、DMAC46、RAM47及びUSBIF48を有し、1個の半導体基板に形成されている。USBIF48は前記USBIF3と同様に、2面でUSBバッファEP1、EP2を有するUSBバッファ部20とUCNT21によって構成され、USB2.0規格に準拠する。RAM47はUSBバッファEP1、EP2の記憶情報に対する後段の2面RAMバッファを構成し、第1RAMバッファ領域BUF1と第2RAMバッファ領域BUF2が割当てられる。望ましくは、第1RAMバッファ領域BUF1と第2RAMバッファ領域BUF2に対して夫々個別のアクセスポートを持つデュアルポートであるのがよい。即ち、一方のUSBバッファEP1がホストコンピュータ25から受信したデータを格納しているとき、他方のUSBバッファEP2から出力されるデータをRAM47の第1RAMバッファ領域BUF1に格納し、これに並行して第2バッファ領域BUF2に格納されているデータを所定のポートから外部に出力することができる。また、他方のUSBバッファEP2がホストコンピュータ25から受信したデータを格納しているとき、USBバッファEP1から出力されるデータをRAM47の第2RAMバッファ領域BUF2に格納し、これに並行して第1RAMバッファ領域BUF1に格納されているデータを所定のポートから外部に出力することができる。USBIF48からRAM47へのデータ転送にはDMAC46を用いることが望ましい。

10

#### 【0040】

FPGA41はフラッシュメモリセルのような不揮発性記憶素子を多数有し、それら不揮発性記憶素子のプログラム状態に応じて論理機能を所望に設定することができる回路である。ここではFPGA41は、マイクロコンピュータ33のJTAGIF8とデータ送受信を行なうためのJTAGインタフェースロジック41Aと、SSRAM42のアクセス制御を行なうアドレス生成ロジック41Bを構成する。

20

#### 【0041】

JTAGインタフェースロジック41Aは、出力FIFOバッファFIFOTDO、入力バッファTDI、インタフェース制御回路(JCNT)50を有する。RAM47から出力FIFOバッファFIFOTDOへのデータライトの際、USBインタフェース制御プログラムを実行するCPU44は出力FIFOバッファFIFOTDOへの書き込み可能状態(TDOST)と空き状態(TDOF)等を管理し、出力FIFOバッファFIFOTDOへ連続して転送データをセットする。マイクロコンピュータ33のJTAGIF8からのアクセス許可ビットはJCNT50が監視する。アクセス許可ビットがイネーブルになったとき出力FIFOバッファFIFOTDOから4バイトの情報送信が行なわれる。出力FIFOバッファFIFOTDOから送信された情報はマイクロコンピュータ33のJTAGIF8で受信され、データレジスタSDDRに格納される。データレジスタSDDRに格納されると、DMAC6が起動され、その情報はDMA転送にてEXMRY14に書き込まれる。

30

#### 【0042】

前記SSRAM42はマイクロコンピュータ33のAUDIF10から出力されるトレース情報の格納に利用される。前記FPGA41で実現されるSSRAM42のアクセス制御を行なうアドレス生成ロジックはアドレスカウンタ(ACOUNT)52と、セレクタ(ASEL)53を構成する。セレクタ53はSSRAM42のアドレッシングをCPU44で行なうかアドレスカウンタ52で行なうかを選択する。どちらを選択するかはCPU44の指示に従って決定される。

40

#### 【0043】

このように、ホストコンピュータ25に接続するUSBIF48と、ターゲットコンピュータ33のJTAGIF8の間には、受信データの入力と出力を並列可能にするバッファBUF1、BUF2による2面バッファと、2面バッファの一方のバッファから出力される受信データをFIFO形式で蓄えることができる出力FIFOバッファFIFOTDOが設けられているから、ホストコンピュータ25とマイクロコンピュータ33がUISBインタフェースで直結されていないが、ある程度データ転送効率を上げることができる

50

## 【 0 0 4 4 】

特に、内蔵RAM47を数キロバイトのダブルバッファ構造として、ホストコンピュータ25からのダウンロードデータをバッファBFU1またはBFU2の一方に受信中に、他方のバッファが保持しているデータを出力FIFOバッファFIFOTDOにライトするように、処理を並列化したから、バッファEP1、EP2のデータサイズとバッファFIFOTDOのデータサイズが異なる場合であっても、また、バッファEP1、EP2の動作とバッファFIFOTDOの動作が独立しているためDMAC転送を途中で中断させないという制約が必要になる場合であっても、対応することが可能になる。

## 【 0 0 4 5 】

図5にはFPGA41で構成されるJTAGインタフェースロジック41Aの詳細が示される。図6には図5の動作タイミングが示される。

## 【 0 0 4 6 】

バッファFIFOTDOは4バイト×61段(244バイト)の多段構成とされる。55はマイクロコンピュータ40に接続されるインタフェースである。バッファFIFOTDOはFIFO56とシフトレジスタ57によって構成される。コントロールブロック58とバッファコントロールブロック59はJCNT50を構成する。59A、59Bは所定のロジックである。CPUDATA(REGDATA.D)はRAM47からの出力データ、CPUWR\_N(TDOWR\_N)はFIFO56に対するライトリクエスト、TDOREG.QはFIFO56からのリードデータ、READQはFIFO56に対するリードリクエスト、SHIFTRREG.はシフトレジスタ57の出力データ(TDO出力)である。SBUF\_LOADはシフトレジスタ57のデータロード信号、S\_Pはシフトレジスタ57のシフト信号である。TDOemはFIFO56のエンpty信号、TDOflはFIFO56のフル信号である。TDOSTはバッファFIFOTDOにライト可能か否かを示し、1でライト可、0でライト不可を意味する。TDOFは1でバッファFIFOTDOのデータがすべてシフトアウトしたこと、0でバッファFIFITDOのデータがすべてシフトアウトしていない(初期値)ことを示す。TDOINTはバッファFIFITDOのデータが全てシフトアウトした時に割込要求信号IRQを出力させるための信号であり、1で割込要求を可能とし、0で割込要求を抑止する(初期値)。

## 【 0 0 4 7 】

マイクロコンピュータ40のRAM47より、ライトイネーブル(TDOWR\_N)に応答してCPUクロックCPUCLK同期でデータ(REGDATA.D)がFIFO56に書き込まれる。FIFO56がフル状態のときは、CPUクロックCPUCLKに同期してTDOflが出力される。FIFO56がエンpty状態の時は、クロックTCKに同期してTDOemが出力される。シフトレジスタ57へのデータ転送開始ロードパルス(SBUF\_LOAD)は、FIFO56が空の時はライトイネーブル(TDOWR\_N)に同期して生成され、空でない時は転送終了パルス(DONE\_P)に同期して生成される。この作用はロジック59A、59Bにより実現される。DONE\_Pは、ターゲットマイクロコンピュータ33からのアクセス許可ビットに基づいて生成される。アクセス許可ビットは、レジスタTDIを介して入力される。バッファコントロールブロック59ではシフトレジスタ57のデータ転送終了毎にFIFO56のデータ書き込み可能状態を示すTDOflをチェックする。TDOflがバッファFIFITDOの書き込み可能状態を示すTDOSTビットを生成し、TDOemとDONE\_PによりバッファFIFOTDOの空き状態を示すTDOFビットを生成し、コントロールブロック58のJTAGレジスタに反映する。TDOFをマイクロコンピュータ40への割込み要求に使用する場合、割込み許可を示すTDOINTビットとTDOFによって割込み要求信号IRQを生成する。CPU44がこの割込み要求を受付けると、例えば図7のTDOデータセット処理実行における、次のRAMバッファへの切り換え(S7)を行う。

## 【 0 0 4 8 】

図7にはRAM47の2面RAMバッファBUF1、BUF2を切り換えてデータをバ

10

20

30

40

50

バッファFIFO TDOに転送する制御手順が示される。まず、この制御手順を可能にするTDOINTが1にされ(S1)、TDOST=1かによってバッファFIFO TDOの空きがチェックされ(S2)、空きがあればバッファFIFO TDOにデータが転送される(S3)。転送されたデータバイト数がチェックされる。即ち、バッファBUF1, BUF2の夫々に割当てられる領域の最大容量数が否かがチェックされる(S4)。その転送バイト数に到達すると、TDOF=1かの判定によってFIFO TDOの空きチェックを行ない(S5)、全てシフトアウトしたことが判定されると、CPU44にIRQにて割り込みを要求し、TDOFを0にクリアする(S6)。その後、転送元RAMバッファ領域が次のバッファBUF1又はBUF2に切り換えられる(S7)。上記処理を送信完了まで繰返す(S8)。完了後、TDINTを0にクリアして(S9)、制御を終了する。

10

#### 【0049】

このように、バッファBUF1からバッファFIFO TDOへのデータライト転送は、バッファFIFO TDOのバッファの書き込み可能状態(TDOSTビット)を参照して、RAM47の最大容量数キロバイト数を転送し、次にバッファFIFO TDOの空き状態(TDOF)を確認した後、次のバッファBUF2又はBUF1へ切り替え、指定したデータ量を転送完了する迄その処理を繰り返す。これにより、RAM47からバッファFIFO TDOへのデータライトの際、TDOSTビットとTDOFビットを管理し連続して転送データをセットする。マイクロコンピュータ33からバッファTDIに送られるアクセス許可ビットはハードロジック58が監視する。それらの結果、図8の(A)に示すように、アクセス許可ビットを検出してから転送データを転送するまでのオーバーヘッドを無くし、最大数百バイトの連続転送が可能になり、ユーザプログラムなどのダウンロード転送の高速化を実現することができる。ターゲットマイクロコンピュータがUSBIF3を持たない場合にもプログラムダウンロード転送速度をある程度高速化することが可能である。尚、図8の(A)において、マイコン書き込みとはRAM47からバッファFIFO TDOへのデータ転送書き込みを意味する。JTAG出力とはバッファFIFO TDOからターゲットマイクロコンピュータ33への転送データ出力を意味する。SPはターゲットマイクロコンピュータ33からのアクセス許可ビットのステータスポーリング処理を意味する。

20

#### 【0050】

図9には比較例に係るエミュレータが例示される。ホストコンピュータとのインタフェースにはUSBを用いる。USBドライバ61及びUSBコントローラマイコンチップ62が設けられる。USBコントローラマイコンチップ62のプログラムはROM63に格納され、SDRAM64がワークメモリとして利用される。受信したUSBデータはSDRAM64に一旦格納され、USBパケット解析等を経てSDRAM64に格納してからJTAGコントローラ65のデータ出力レジスタからターゲットマイクロコンピュータにデータを送信する。JTAG準拠のインタフェースではソフトダウンロードデータ転送方式は、送信データの終了をターゲットCPUからアクセス許可ビットを受け取ることができるステータス取得モードで行っている。また1回のデータ転送量は最大で4バイトである。エミュレータは転送の都度ターゲットマイクロコンピュータ33からステータス取得モード終了を示すアクセス許可ビットを取得する。要するに、アクセス許可ビットをポーリングしながら取得して、データ出力レジスタ(TDO)へ転送データをセットしなければならない。この場合には図8の(B)に示されるように、アクセス許可ビットの検出から転送データセットまでにオーバーヘッドT1が発生する。要するに、図8の(B)は4バイト転送毎にデータ出力レジスタ(TDO)へのデータセットを行い、図8の(A)はデータ出力レジスタ(TDO)へのデータセットを予め行い、ハードロジック58でアクセス許可ビット取得毎に4バイト単位でデータ出力を行う。

30

40

#### 【0051】

図10には図5の例と図9の比較例に関しホストコンピュータPCよりターゲットマイクロコンピュータの外付メモリに1メガバイトのユーザプログラムファイルをダウンロ

50

ードするときのダウンロード性能が示される。傾向線 a は図 5 に対応され、傾向線 b は図 9 に対応される。これによればホストコンピュータからターゲットマイクロコンピュータへデータ転送する際、図 9 の比較例におけるプログラムダウンロード性能は、 $TCCK = 10\text{MHz}$  以上にしても  $230\text{KB/秒}$  より変化はなかった。

【0052】

図 5 の例では、 $TCCK = 10\text{MHz}$  以上でもダウンロード性能がリニアに上がる。また、 $TCCK = 20\text{MHz}$  である場合、ダウンロード性能が約 2 倍の  $400\text{KB/秒}$  となる。

【0053】

図 11 にはマイクロコンピュータ 1 の更に具体的な例が示される。図 1 との相違点は、USB インタフェース回路 3 に複数の USB バッファ部 (BEP0 ~ BEP6) 20 を示している。更に、ブートプログラムを保有する ROM 71 を図示している。前記エミュレーションメモリ 11 は SRAM で構成されるのに対し、ROM 71 はマスク ROM 又は EEPROM やフラッシュメモリなどの電氣的に書き換え可能な不揮発性メモリによって構成される。その他にブレーク回路 72 が設けられる。ブレーク回路 72 は、デバッグモードにおいて CPU 2 を介してブレーク条件が設定され、ユーザモードにおいてブレーク条件に一致する状態の発生を検出して、CPU 2 にブレーク例外を要求する。

【0054】

前記 USB バッファ部 20 の BEP0 ~ BEP6 の夫々は USB 規格におけるエンドポイントと称される FIFO (First-In First-Out) バッファを意味する。FIFO バッファは入力用 (イン)、出力用 (アウト) 夫々別々に備えたダブルバッファ構造とされ、図 1 に示される EP1、EP2 のように構成される。BEP0 はエンドポイント 0、BEP6 はエンドポイント 6 と理解されたい。エンドポイントの番号は例である。一つの USB デバイスが持つことができるエンドポイントの最大数は USB 規格で規定されている。エンドポイント 0 (BEP0) はコントロール転送に使用され、USB デバイスにとって必須とされる。ここでは USB インタフェース回路 3 はユーザモードとデバッグモードの双方で利用可能とされ、USB バッファ部 (BEP0 ~ BEP6) 20 のうち、BEP1、BEP2 はユーザプログラムによる使用に専用化され、BEP3 ~ BEP6 はデバッグ支援に専用化される。BEP0 は双方で共用される。BEP0 はオンチップデバッグ専用のディスクリプタ情報をコントロール転送するのに使用される。BEP3 は USB データのバルクアウト転送に利用される BEP4 は USB データのバルクイン転送用に利用される。BEP5 は命令コマンドの入力に利用される。BEP6 はステータス情報の出力に利用される。BEP1、BEP2 はユーザモードにおいてユーザの設定に従ってデータ入出力などに利用される。USB インタフェース回路 3 は外部から与えられる  $48\text{MHz}$  のような周波数のクロック信号 CLK に同期動作される。CLK はマイクロコンピュータ 1 の内部で PLL 回路などを介して生成されてもよい。

【0055】

前記 ROM 71 が保有するブートプログラムは、特に制限されないが、USB インタフェース回路 3 を初期化するための USB 初期化制御プログラム及び転送制御プログラムとされる。転送制御プログラムは、USB インタフェース回路 3 を介して受信したデバッグ支援プログラム (ASEファームソフトとも称する) をエミュレーションメモリ 11 に格納するためのプログラムとされる。

【0056】

システムコントローラ (SYSC74) は代表的に示された外部端子としてデバッグモード端子 ASEMD 及びリセット端子 RES に接続され、また、パワーオンリセット回路 (PORES) 73 から出力されるパワーオンリセット信号が供給され、それら入力に従ってマイクロコンピュータ 1 の動作モード等を制御する。SYSC74 は、デバッグモード端子 ASEMD によってユーザモードが指定され、パワーオンリセット回路 73 からのパワーオンリセット指示又はリセット端子 RES からのリセット指示があると、制御信号 1 によって CPU が初期化され、CPU 2 は ROM 71 が保有する前記 USB 初期化制御プログラムを実行してエンドポイント BE0、BEP1、BEP2 の動作を可能とし、

10

20

30

40

50

最後にCPU 2はプログラム格納領域の先頭番地から命令を実行開始可能にされる。一方、SYSC 74は、デバッグモード端子ASEMDによってデバッグモードが指定され、パワーオンリセット回路73からのパワーオンリセット指示があると、制御信号 2によってCPU 2が初期化され、CPU 2はROM 71が保有する前記USB初期化制御プログラムを実行してエンドポイントBE0～BEP6の動作を可能とし、その後CPU 2はROM 71が保有する前記転送制御プログラムを実行して、USBインタフェース回路3を介して受信したデバッグ支援プログラム(ASEファームソフトとも称する)をエミュレーションメモリ11に格納し、最後にCPU 2は転送したASEファームソフトの実行が可能にされる。尚、デバッグモードにおいてユーザモードに分岐したユーザプログラムの実行状態では、RESによるリセット指示があってもデバッグモードそれ自体のリセット指示はマスクされる。デバッグモードからユーザモードに分岐してユーザプログラムを実行しているとき、リセット入力でリセットベクタへジャンプするかを検証し足りるのを可能にするためである。

10

#### 【0057】

ASEファームソフトによってユーザプログラムをホストコンピュータ25からダウンロードする場合、CPU 2は、ASEファームソフトにしたがって、USBIF 3が受信したダウンロード要求コマンドに回答して、前記DMAC 6に、USBIF 3が受信したユーザプログラムを前記バッファRAM 7に転送させる。例えば、USBIF 3との通信が完了した後、CPU 2は、前記ASEファームソフトにしたがって、USBIF 3が受信した転送要求コマンドに回答して、前記DMAC 6に、前記バッファRAM 7に転送されたプログラムを外部バスインタフェース回路5を介して外部メモリ4に転送制御させる。CPU 2は、前記ASEファームソフトの実行状態において、モード制御コマンドに回答してユーザモードに移行し、前記ユーザモードにおいてCPU 2は外部メモリ4から命令をフェッチして実行することができる。

20

#### 【0058】

図12にはマイクロコンピュータによるオンチップデバッグの動作タイミングが例示される。時刻t0にASEMDがローレベルにされてデバッグモードが指示され、時刻t1に電源が投入されて、パワーオンリセット回路73によりパワーオンリセットが指示されると(時刻t2)、制御信号 2が活性化され、CPU 2が初期化され、CPU 2はROM 71が保有する前記USB初期化制御プログラムを実行してUSBブート処理を行なってエンドポイントBE0～BEP6の動作を可能とし、その後CPU 2は、ROM 71が保有する前記転送制御プログラムを実行する。すなわち、USBインタフェース回路が書き込みコマンドを受信すると、これに回答して、USBインタフェース回路3を介して受信したデバッグ支援プログラム(ASEファームソフトとも称する)をエミュレーションメモリ11に格納する(ASEファーム書き込み処理)。ASEファーム書き込み処理が終わった後、USBインタフェース回路3がホストコンピュータからブレイクコマンドを受信することにより、CPU 2はEMMR 11上のASEファームソフトを実行可能にされる。この後、USBインタフェース回路3がホストコンピュータからユーザプログラム実行コマンドを受信すると、当該ユーザプログラム実行コマンドで指定されるユーザプログラムのアドレスに処理が分岐し、CPU 2はユーザプログラムを実行するユーザモードに移行される。このユーザプログラムの実行状態において、例えばUSBインタフェース回路がブレイクコマンドを受信すると、CPU 2にブレイク例外要求BERQが発行され、CPU 2によるユーザプログラムの実行が停止され、CPU 2は再びASEファームソフトを実行するデバッグモードに遷移される。

30

40

#### 【0059】

図13にはUSBの通信フォーマットが例示される。1フレームの先頭はSOF(Start of frame)とされる。USB規格におけるUSB通信フォーマットのフレーム内ではDATA0/DATA1がオンチップデバック用データパケットとされる。DATA0/DATA1のリード/ライトは各BEPで行う。オンチップデバック用USBパケットデータはヘッダーとデータから成る。ヘッダーは、識別ID、データサイズ、

50

ステータス、命令コマンドの情報を含む。データ情報にダウンロードデータ等の情報を含む。

【0060】

図14にはホストコンピュータ25(PC側)とUSBインタフェース回路3との通信ハンドシェック制御の基本形が例示される。ホストコンピュータ25から送信されてくるUSBデータパケットをEP3に受信する。次にCPU2がASEファームプログラムに従ってEP3に受信したUSBデータパケットを読み込み、解析して、解析結果に応ずる制御を行い、その制御結果をEP4よりホストコンピュータ25に送信する。

【0061】

図15にはユーザプログラムの実行中に強制ブレークさせる場合のホストコンピュータ25とUSBインタフェース回路3との間でハンドシェック制御内容が例示される。EP5を命令コマンド受信専用バッファとして使用する。EP5で受信した命令コマンドパケットを解析し、これに応じてUSB制御回路21からCPU2へブレーク例外処理を要求し、ASEファームのベクタアドレスが与えられる。CPU2はユーザプログラムの実行状態からASEファームプログラムのブレークのアドレスへジャンプする。USBインタフェース制御回路21はCPU2よりブレークアクナリッジBACKを受信し、EP6にステータス情報(BACK)をセットしホストコンピュータ25に送信する。ホストコンピュータ25はステータスがブレーク状態か否かを確認する。

10

【0062】

図16にはホストコンピュータ25からUSBインタフェース回路3へのソフトウェア(ユーザプログラム)のダウンロードの制御内容が例示される。ホストコンピュータよりソフトダウンロード送信要求と共に転送サイズ、ロード先アドレスがBEP3へ送信する。これに対し、CPU2からBEP4を介して送信許可がホストコンピュータ25に転送され、それが完了次第、ホストコンピュータ25より指定サイズ分だけダウンロードデータがUSBインタフェース回路3に向けて送信される。このとき、USBインタフェース回路のBEP3に受信した場合、USB割り込みにより、DMA転送し、直接外部メモリ4に書き込む。指定サイズ分送信終了たらEP4より送信完了をホストコンピュータに送信する。

20

【0063】

図17にはマイクロコンピュータ1の更に別の例が示される。図11との相違点は、JTAGIF8及びAUDIF10を削除し、更に、USBIF3をデバッグモードに専用化した点である。USBIF3はBEP1, BEP2を備えていない。ユーザ資源としてUSBIF3を利用することはできない。USBIF3はオンチップデバックのときだけ使用可能にされる。AUDIF10は大容量のトレースデータの格納にユーザ資源のオンチップRAM7やEXMRY4を使用することが可能な場合は必要ないためである。

30

【0064】

図18にはマイクロコンピュータ1の更に別の例が示される。図17との相違点は、ユーザ専用資源としてUSBIF3Aを設けた点である。ユーザ専用資源としてUSBIF3AはBEP0, BEP1, BEP2を構成するバッファ20とUSBインタフェース制御回路21Aを備える。オンチップデバックとユーザモードの夫々において全く専用化されたUSBIFを利用可能である。

40

【0065】

特に図示はしないが図11においてAUDIF10だけを廃止することも可能である。USBIF3はオンチップデバックとユーザモードで兼用に使用されるから、ユーザ資源としてUSBIF3をデバックする場合は、デバッグ用インタフェースとしてJTAGIF8を利用すればよい。

【0066】

以上本発明者によってなされた発明を実施形態に基づいて具体的に説明したが、本発明はそれに限定されるものではなく、その要旨を逸脱しない範囲において種々変更可能であることは言うまでもない。

50

【 0 0 6 7 】

例えば、CPU 2 と内部バス 1 2 の間にはキャッシュメモリが配置されてもよい。仮想アドレスをサポートする場合にはCPU 2 と内部バス 1 2 の間にアドレス変換バッファが配置されていてよい。ロジック 4 1 A , 4 1 B はFPGAで構成することに限定されない。

【 0 0 6 8 】

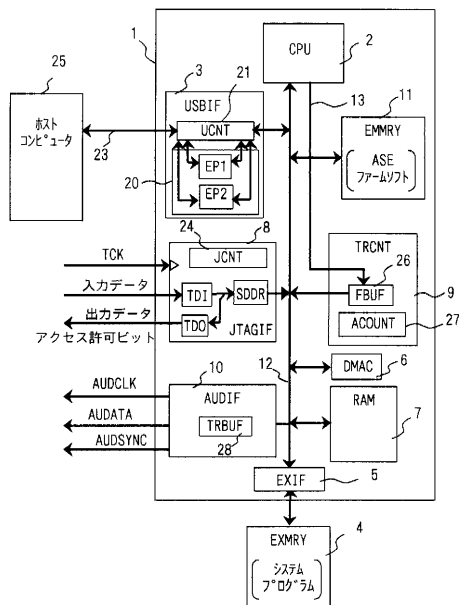
図 4 で説明したマイクロコンピュータ 4 0 が保有するUSBIF 4 8 についても図 1 1 で説明したUSBIF 3 を適用可能である。この場合、ユーザ専用のBEPは必要ない。

【 産業上の利用可能性 】

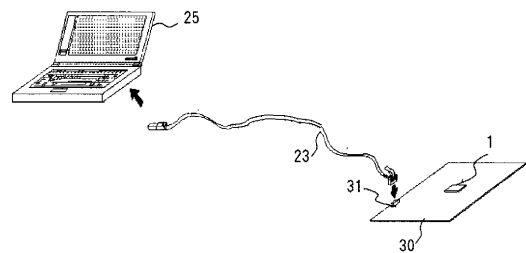
【 0 0 6 9 】

本発明はマイクロコンピュータ、特にデバッグサポート機能を有するマイクロコンピュータ、更にはマイクロコンピュータを動作させるシステムプログラムの開発方法などに広く適用することができる。

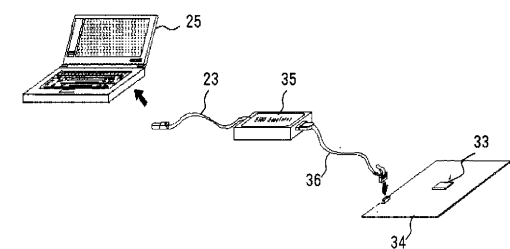
【 図 1 】  
図1



【 図 2 】  
図2



【 図 3 】  
図3





【 図 8 】

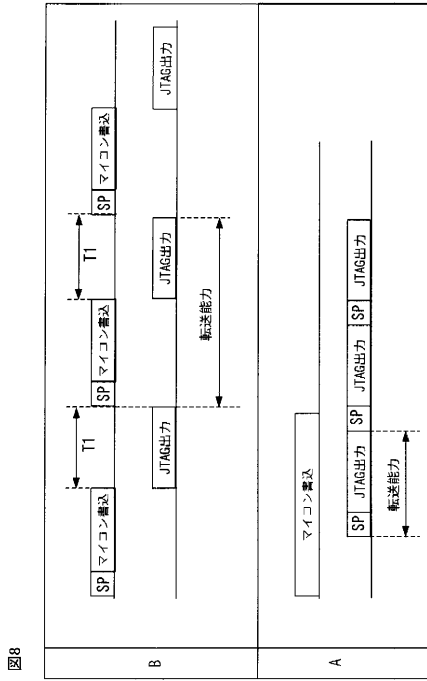


図8

【 図 9 】

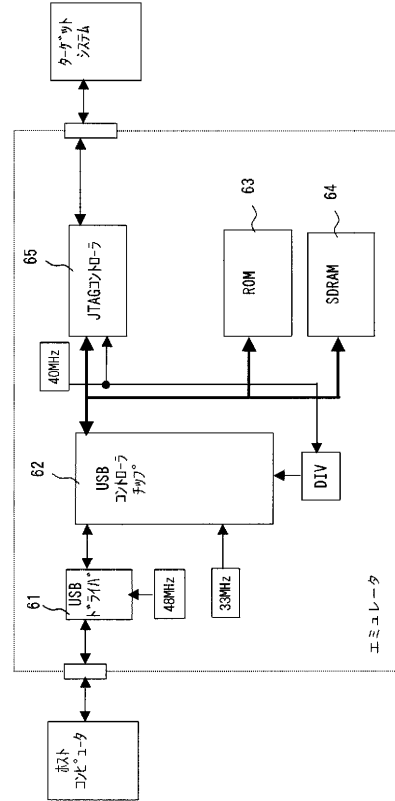


図9

【 図 10 】

図10

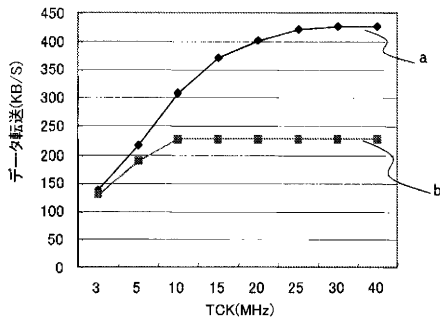


図10

【 図 11 】

図11

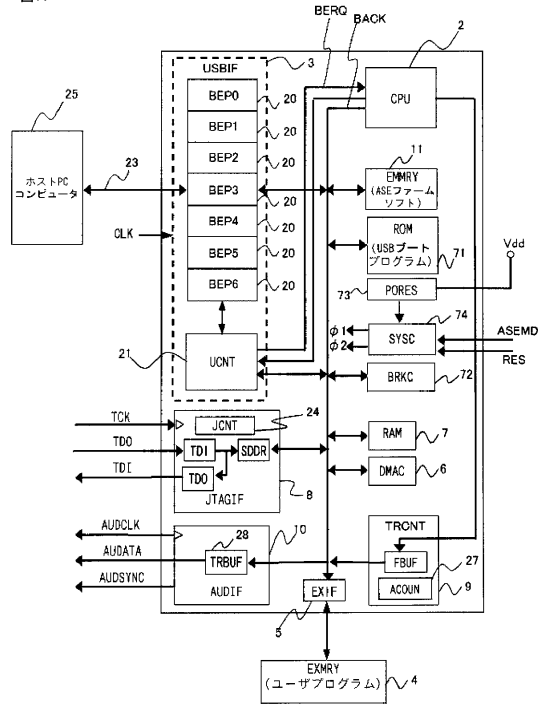


図11

【 図 1 2 】

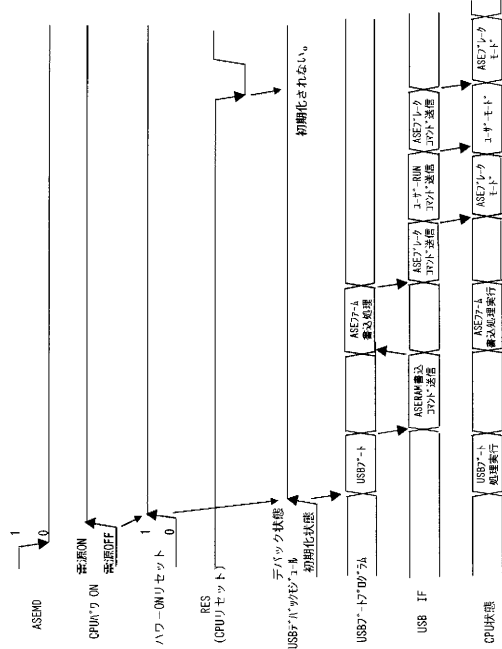
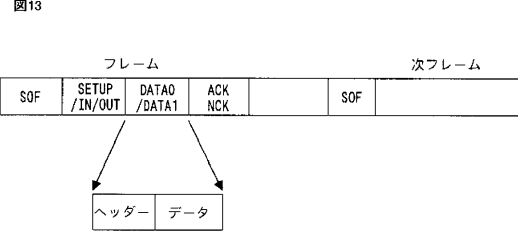
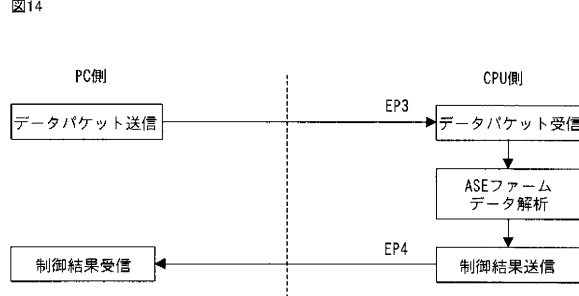


図 12

【 図 1 3 】

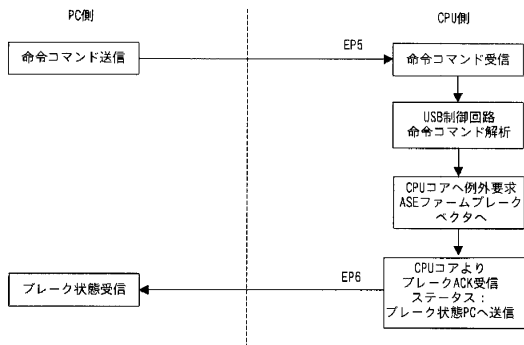


【 図 1 4 】



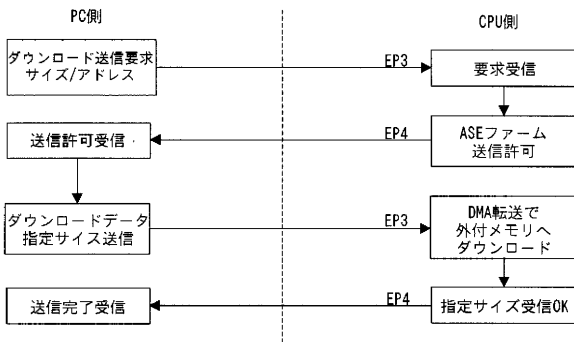
【 図 1 5 】

図 15



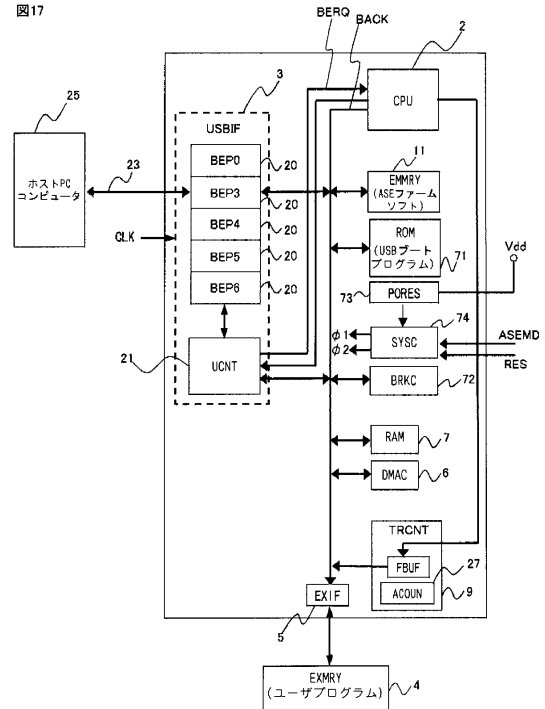
【 図 1 6 】

図 16

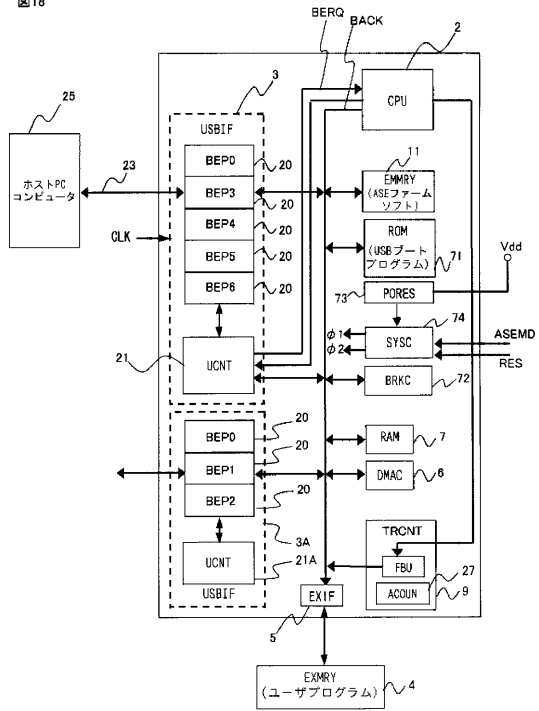


【 図 1 7 】

図 17



【 図 18 】  
図18



---

フロントページの続き

- (56)参考文献 特開平11-039186(JP,A)  
特開2002-268911(JP,A)  
特開平11-039184(JP,A)  
宇賀神孝, バウンダリースキャンテスト“JTAG”の考え方・使い方, エレクトロニクス, 株式会社オーム社, 2001年 4月 1日, 第46巻 第4号, p.42-53

(58)調査した分野(Int.Cl., DB名)

G06F9/445  
G06F11/22  
G06F11/28  
G06F13/38  
G06F15/78  
G01R31/28