

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2007-213292

(P2007-213292A)

(43) 公開日 平成19年8月23日(2007.8.23)

(51) Int. Cl.	F I	テーマコード (参考)
G06F 9/445 (2006.01)	G06F 9/06 640A	5B076
G06F 13/38 (2006.01)	G06F 13/38 340	5B077
	G06F 13/38 350	5B176

審査請求 未請求 請求項の数 6 O L (全 8 頁)

(21) 出願番号 特願2006-32038 (P2006-32038)
 (22) 出願日 平成18年2月9日(2006.2.9)

(71) 出願人 302062931
 NECエレクトロニクス株式会社
 神奈川県川崎市中原区下沼部1753番地
 (74) 代理人 100103894
 弁理士 冢入 健
 (72) 発明者 植村 真理子
 神奈川県川崎市中原区下沼部1753番地
 NECエレクトロニクス株式会社内
 (72) 発明者 須田 敬偉
 神奈川県川崎市中原区下沼部1753番地
 NECエレクトロニクス株式会社内
 Fターム(参考) 5B076 AA06 BB06 BB11
 5B077 BA09 NN02
 5B176 AA06 BB06 BB11

(54) 【発明の名称】 マルチプロセッサシステム及びスレーブシステムの起動方法

(57) 【要約】

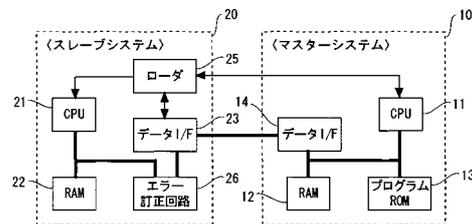
【課題】

簡易な構成で部品点数を低減できるマルチプロセッサシステム及びスレーブシステムの起動方法を提供すること。

【解決手段】

本発明にかかるマルチプロセッサシステムは、マスターシステム10と、マスターシステム10とプロセッサ間通信インターフェースを介して接続されたスレーブシステム20を備えている。マスターシステム10は、スレーブシステム20の起動プログラムを格納するプログラムROM13を有する。また、スレーブシステム20は、プログラムROM13に格納された起動プログラムをプロセッサ間通信インターフェースを介して取得するローダ25を有する。

【選択図】 図1



【特許請求の範囲】**【請求項 1】**

マスターシステムと、前記マスターシステムとプロセッサ間通信インターフェースを介して接続されたスレーブシステムを備えたマルチプロセッサシステムであって、

前記マスターシステムは、

前記スレーブシステムの起動プログラムを格納するプログラムメモリを有し、

前記スレーブシステムは、

前記プログラムメモリに格納された起動プログラムを前記プロセッサ間通信インターフェースを介して取得するローダを有するマルチプロセッサシステム。

【請求項 2】

マスターシステムと、前記マスターシステムとプロセッサ間通信インターフェースを介して接続されたスレーブシステムを備えたマルチプロセッサシステムであって、

前記マスターシステムは、

マスターシステム側プロセッサと、

前記スレーブシステムの起動プログラムを格納するプログラムメモリとを有し、

前記スレーブシステムは、

スレーブシステム側プロセッサと、

前記スレーブシステム側プロセッサにより実行されるプログラムを格納するプログラム実行用メモリと、

前記マスターシステムのプログラムメモリに格納された起動プログラムを前記プロセッサ間通信インターフェースを介して取得するローダとを有し、

前記マスターシステム側プロセッサは、

前記ローダに対して起動プログラムのダウンロード命令を発行する手段と、

前記ローダに対して起動プログラムを転送する手段とを有し、

前記ローダは、

前記ダウンロード命令の発行に応じて前記マスターシステム側プロセッサに対して命令確認を発行する手段と、

転送された起動プログラムを受け取り、前記プログラム実行用メモリに格納する手段とを有するマルチプロセッサシステム。

【請求項 3】

前記プロセッサ間通信インターフェースは、メモリを介さない通信インターフェースであることを特徴とする請求項 1 又は 2 記載のマルチプロセッサシステム。

【請求項 4】

前記プロセッサ間通信インターフェースは、シリアル通信インターフェースであることを特徴とする請求項 3 記載のマルチプロセッサシステム。

【請求項 5】

マスターシステムとプロセッサ間通信インターフェースを介して接続されたスレーブシステムの起動方法であって、

前記マスターシステムより、前記スレーブシステムの起動プログラムを前記スレーブシステムに対して前記プロセッサ間通信インターフェースを介して転送し、

前記スレーブシステムにおいて、前記転送された起動プログラムをプログラム実行用メモリに格納し、

前記プログラム実行用メモリに格納された起動プログラムに基づいてスレーブシステムを起動するスレーブシステムの起動方法。

【請求項 6】

マスターシステムとプロセッサ間通信インターフェースを介して接続されたスレーブシステムの起動方法であって、

マスターシステム側プロセッサが、前記ローダに対して起動プログラムのダウンロード命令を発行し、

前記スレーブシステムのローダが、前記ダウンロード命令の発行に応じて前記マスター

10

20

30

40

50

システム側プロセッサに対して命令確認を発行し、

前記マスターシステム側プロセッサが、前記ローダに対して起動プログラムを転送し、前記ローダが、転送された起動プログラムを受け取り、プログラム実行用メモリに格納し、

前記プログラム実行用メモリに格納された起動プログラムに基づいてスレーブシステムを起動するスレーブシステムの起動方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、複数のプロセッサを有するマルチプロセッサシステムに関するものであり、特に、マスターシステムとスレーブシステムを備えたマルチプロセッサシステムにおけるスレーブシステムの起動方法に関する。

10

【背景技術】

【0002】

組み込み機器の性能向上と低消費電力化という要求に応える一つの解として、複数のプロセッサ・コアを搭載したマルチプロセッサシステムが注目を浴びている（例えば、非特許文献1参照）。

【0003】

従来のマルチプロセッサシステムの構成例を図4に示す。図4に示されるように、従来のマルチプロセッサシステムは、マスターシステム10とスレーブシステム20を備えている。そして、マスターシステム10は、CPU（Central Processor Unit：中央制御装置）11、RAM（Random Access Memory）12、プログラムROM（Read Only Memory）13、データインターフェース（I/F）回路14を備えている。また、スレーブシステム20もマスターシステム10と同様にCPU21、RAM22、プログラムROM24、データインターフェース（I/F）回路23を備えている。

20

【0004】

図4に示す従来例において、マスターシステム10とスレーブシステム20は、ともにプログラムROMを備え、それぞれ、プログラムROMに格納された起動プログラム（ブートプログラム）に従って、それぞれのCPU11, 21が起動処理を実行する。そのため、図4に示す従来例では、マスターシステム10とスレーブシステム20の双方に、マスクROMやFLASH等のプログラムROMを備えているので、部品点数が増えるという問題があった。特に、システム構成の高度化に伴って、最近のマルチプロセッサシステムには、2つ以上のスレーブシステムを持つものも珍しくなくなり、スレーブシステムの数と同じ数のプログラムROMが必要となり、部品点数の増加がさらに問題化している。

30

【0005】

特許文献1には、マルチバスラックを用いた制御システムに関する技術が開示され、特に、マスター基板にあるROMから他のスレーブ基板のRAMに対してプログラムを転送する点が開示されている。また、特許文献2には、マスターCPUから共通起動プログラムをスレーブCPUに転送し、これによりスレーブCPUが起動する点が開示されている。特許文献3には、複数のCPUを有するマルチCPU用メモリシステムにおいてプログラムROMを共有する点が開示されているが、マスターシステムとスレーブシステムより構成されたマルチプロセッサシステムに関するものではない。また、特許文献4には、複数のCPUを有する通信装置が他の通信装置からプログラムをダウンロードする技術が開示されているが、マスターシステムとスレーブシステムより構成されたマルチプロセッサシステムに関するものではない。

40

【特許文献1】特開2000-137671号公報

【特許文献2】特開平10-320366号公報

【特許文献3】特開平4-98448号公報

【特許文献4】特開2005-31796号公報

【非特許文献1】「組み込み分野の『マルチコア』ハード/ソフトの課題，OSの概要と

50

実装事例」, デザインウェブマガジン 2005年8月号, 吉田正康, 本田晋也, CQ出版株式会社

【発明の開示】

【発明が解決しようとする課題】

【0006】

特許文献1では、プログラムはマスター基板からスレーブ基板に対して共通メモリバスで転送されるため、プログラムを転送するために複雑なハードウェア構成が必要となるという課題がある。

【0007】

また、特許文献2では、マスタCPUから共通起動プログラムを一端デュアルポートRAMに格納しておき、これをスレーブCPUがローディングして起動することとしているので、高価なデュアルポートRAMを設け、かつその両側にメモリバスを設けなければならない、高コスト化及び部品点数の増大という課題が生じる。

【課題を解決するための手段】

【0008】

本発明にかかるマルチプロセッサシステムは、マスターシステムと、前記マスターシステムとプロセッサ間通信インターフェースを介して接続されたスレーブシステムを備えたマルチプロセッサシステムであって、前記マスターシステムは、前記スレーブシステムの起動プログラムを格納するプログラムメモリを有し、前記スレーブシステムは、前記プログラムメモリに格納された起動プログラムを前記プロセッサ間通信インターフェースを介して取得するローダを有するものである。本発明は、このような構成を有するので、スレーブシステムにおいて起動プログラムを格納するためのプログラムメモリを設ける必要がないため部品点数を減らすことができる。特に、起動プログラムをプロセッサ間通信インターフェースを介して取得することとしたので、ローダの構成を簡易にすることができる。

【発明の効果】

【0009】

本発明によれば、簡易な構成で部品点数を低減できるマルチプロセッサシステム及びスレーブシステムの起動方法を提供することができる。

【発明を実施するための最良の形態】

【0010】

図1は、本発明の実施の形態にかかるマルチプロセッサシステムの構成図である。本例のマルチプロセッサシステムは、例えば携帯電話機に搭載することができる。図において、当該マルチプロセッサシステムは、マスターシステム10とスレーブシステム20により構成されている。本発明の実施の形態にかかるマルチプロセッサシステムは、1チップ上に集積されており、ヘテロジニアス構成を有している。図1に示す例では、それぞれ一つのマスターシステム10とスレーブシステム20より構成されているが、複数のスレーブシステム20を備えていてもよい。

【0011】

マスターシステム10は、CPU11、RAM12、プログラムROM13及びデータインターフェース(I/F)回路14を備えている。CPU11は、RAM12、プログラムROM13及びデータインターフェース(I/F)回路14のそれぞれとデータバスにより接続されている。

【0012】

本例におけるプログラムROM13は、マスターシステム10の起動プログラムに加えて、スレーブシステム20の起動プログラムを格納している。

【0013】

スレーブシステム20は、CPU21、RAM22、データインターフェース(I/F)回路23、ローダ25、エラー訂正回路26を備えている。本例にかかるスレーブシステム20は、プログラムROM24を備えていない。CPU21は、RAM22及びエラ

10

20

30

40

50

ー訂正回路26と接続され、さらに、エラー訂正回路26を介してデータI/F回路23と接続されている。ローダ25は、CPU21及びデータI/F回路23と制御線により接続されている。

【0014】

ローダ25は、スレーブシステム20においてCPU21を起動する前に、マスターシステム10からスレーブシステム20の起動プログラムをダウンロードするオートイニシャルプログラムローダ(Auto Initial Program Loader)であり、ソフトウェアではなく、ハードウェアにより構成されている。このローダ25は、簡易通信プロトコルを実現し、かつRAM22への転送機能を有する。

【0015】

エラー訂正回路26は、マスターシステム10からダウンロードした起動プログラムのエラー訂正処理を実行する回路である。

【0016】

マスターシステム10のデータI/F回路14とスレーブシステム20のデータI/F回路23は、それぞれ、プログラム転送フェーズで通常モードで使用するCPU間通信I/Fを実現するインターフェース回路である。本例では、特に、データI/F回路14とデータI/F回路23は、メモリを介さずにシリアルデータバスにより直接接続されている。つまり、本例におけるデータI/F回路14とデータI/F回路23は、共通メモリバスにより接続されるものではない。データI/F回路14とデータI/F回路23は、シリアルデータバスでなくとも、2~4本のパラレルデータバスにより接続するようにしてもよい。

【0017】

また、マスターシステム10のCPU11とローダ25は、制御線により接続されている。

【0018】

続いて、図2のフローチャートを用いて、本発明の実施の形態にかかるマルチプロセッサシステムにおける処理の流れについて説明する。このフローチャートを実行する直前の状態は、マスターシステム10が起動され、スレーブシステム20が起動されていない。

【0019】

最初にマスターシステム10のCPU11は、スレーブシステム20のローダ25に対して制御線を介してスレーブシステムの起動プログラムのダウンロード命令を発行する(S101)。

【0020】

スレーブシステム20のローダ25は、当該ダウンロード命令を受領し、これに応じて、スレーブシステム20のデータI/F回路23を起動する。これに応じて、データI/F回路23は起動する。さらに、ローダ25は、マスターシステム10のCPU11に対してダウンロード命令に関して命令確認を発行する(S102)。

【0021】

マスターシステム10のCPU11は、ローダ25から命令確認を受領し、これに応じて、データI/F回路14を介してさらにローダ25に対して転送コード数及び転送モードを転送する(S103)。

【0022】

スレーブシステム20のローダ25は、マスターシステム10のCPU11よりデータI/F回路14及びシリアルデータ配線を介して転送された転送コード数及び転送モードをデータI/F回路23を介して受領し、所定の記憶手段に格納する。ローダ25は、転送コード数及び転送モードの受け取りに応じて、マスターシステム10のCPU11に対して制御線を介して受取確認を発行する(S104)。

【0023】

マスターシステム10のCPU11は、制御線を介して受取確認を受領すると、これに応じて、起動プログラムを構成するプログラムコードをスレーブシステム20のローダ2

10

20

30

40

50

5 に対してデータ I / F 回路 14、23 を介して転送する (S 1 0 5)。このとき、プログラムコードに対して E C C (Error Correcting Code : 誤り訂正符号) ビットを付加することも可能であるが、付加しなくてもよい。

【 0 0 2 4 】

スレーブシステム 20 のローダ 25 は、マスターシステム 10 から転送されたプログラムコードをデータ I / F 回路 23 を介して R A M 2 2 に格納する。このとき、マスターシステム 10 から転送されたプログラムコードに E C C ビットが付加されている場合には、エラー訂正回路 26 によってエラー訂正処理を実行した後に、R A M 2 2 に格納する (S 1 0 6)。そして、上述のステップ S 1 0 5 及び S 1 0 6 を、スレーブシステム 20 の起動プログラムを構成するプログラムコード数分繰り返す。

10

【 0 0 2 5 】

スレーブシステム 20 のローダ 25 は、ステップ S 1 0 3 において C P U 1 1 より転送された転送コード数分のプログラムコードを受領したか否かを判断する。ローダ 25 は、当該転送コード数分のプログラムコードを受領したものと判断した場合には、転送受領コード数、転送ステータス情報を転送終了確認としてマスターシステム 10 の C P U 1 1 に対して制御線を介して発行する (S 1 0 7)。

【 0 0 2 6 】

マスターシステム 10 の C P U 1 1 は、転送終了確認を制御線を介して受領し、これに応じて、転送終了確認の内容をチェックする。C P U 1 1 は、転送終了確認の内容のチェックの結果、正常転送終了と判断した場合には、スレーブシステム 20 のローダ 25 に対して制御線を介してスレーブシステム 20 の C P U 2 1 の起動コマンドを発行する (S 1 0 8)。

20

【 0 0 2 7 】

スレーブシステム 20 のローダ 25 は、当該起動コマンドを制御線を介してマスターシステム 10 の C P U 1 1 から受け取り、これに応じてスレーブシステム 20 の C P U 2 1 のリセットを解除する (S 1 0 9)。

【 0 0 2 8 】

そして、リセットを解除されたスレーブシステム 20 の C P U 2 1 は、スレーブシステム 20 の R A M 2 2 に格納された起動プログラムに従って動作を開始する (S 1 1 0)。

【 0 0 2 9 】

以上、説明したように、本発明の実施の形態によれば、プログラム R O M をマスターシステム 10 にのみ設ければよく、スレーブシステム 20 に設けなくてもよいため、部品点数の削減及びコスト低減を図ることができる。

30

【 0 0 3 0 】

さらに、本発明の実施の形態によれば、マスターシステム 10 のプログラム R O M 1 3 に格納されたスレーブシステム 20 の起動プログラムは、メモリを介さない C P U 間通信 I / F によりスレーブシステム 20 の R A M 2 2 に転送されるので、部品点数を削減でき、かつ低コスト化を図ることができる。特に、スレーブシステム 20 の起動プログラムを共通メモリバスを介して転送する場合と比較すると、転送処理のための回路構成を簡易化できる。

40

【 0 0 3 1 】

なお、上述の例では、スレーブシステム 20 のローダ 25 は、転送コード数をチェックすることで起動プログラムの最後となるプログラムコードを認識したが、これに限らず、最後であることを示すエンドコードをマスターシステム 10 の C P U 1 1 が付加するようにしてもよい。また、スレーブシステム 20 のローダ 25 がタイマーによりプログラムコードを受け取ってからの経過時間を測定し、経過時間が所定時間を越える場合に転送終了と判断させる仕組みを持つようにしてもよい。

【 0 0 3 2 】

また、上述の例では、1つのマスターシステム 10 に対して1つのスレーブシステム 20 を設けた構成であったが、これに限らず、図 3 (a) のようにマスターシステム 10 に

50

対して並列にそれぞれがローダ 25 を有する複数のスレーブシステム 201, 202 · · · 20n が接続される構成であってもよく、また、図 3 (b) のようにマスターシステム 10 に対して直列にそれぞれがローダ 25 を有する複数のスレーブシステム 201, 202 · · · 20n が接続される構成であってもよい。さらには、スレーブシステムがマトリクス状に接続される構成であってもよい。

【 0 0 3 3 】

図 3 (a) の構成の場合には、マスターシステム 10 のプログラム ROM 13 には、それぞれのスレーブシステム 201, 202 · · · 20n の起動プログラムが個別に格納されていてもよいが、各スレーブシステムで共通の起動プログラムとすることも可能である。

10

【 0 0 3 4 】

図 3 (b) の構成の場合には、マスターシステム 10 に接続されたスレーブシステム 201 は、さらにスレーブシステム 202 ~ スレーブシステム 203 n の起動プログラムをマスターシステム 10 より受け取り、一旦、RAM 22 に格納した後に、順次転送を繰り返して中継するようにしてもよい。

【 図面の簡単な説明 】

【 0 0 3 5 】

【 図 1 】 本発明によるマルチプロセッサシステムの構成図である。

【 図 2 】 本発明によるマルチプロセッサシステムにおける処理の流れを示すフローチャートである。

20

【 図 3 】 本発明によるマルチプロセッサシステムの構成図である。

【 図 4 】 従来のマルチプロセッサシステムの構成図である。

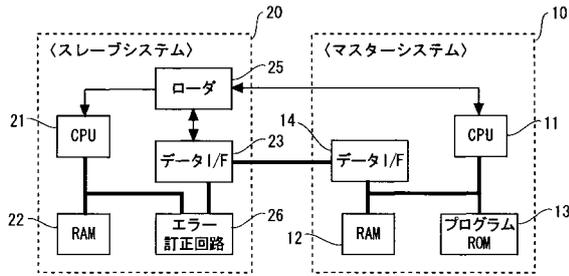
【 符号の説明 】

【 0 0 3 6 】

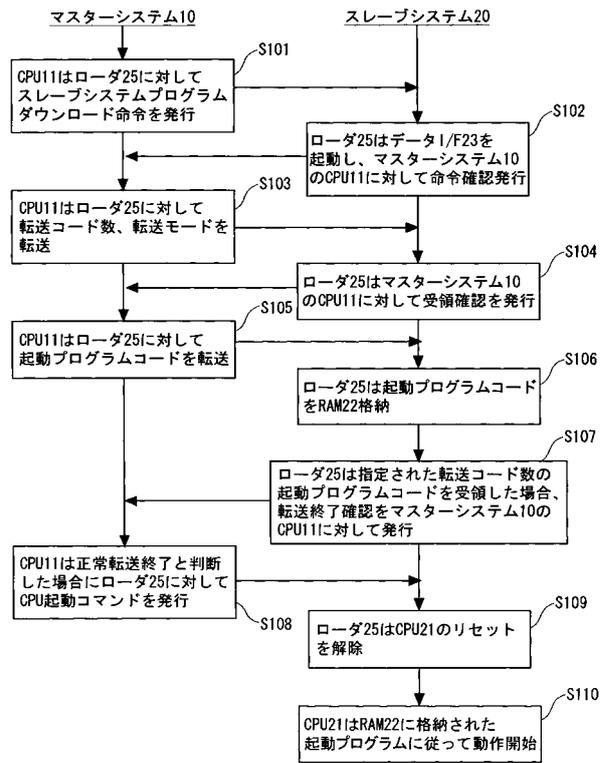
- 10 マスターシステム
- 11 CPU
- 12 RAM
- 13 プログラム ROM
- 14 データ I / F 回路
- 20 スレーブシステム
- 21 CPU
- 22 RAM
- 23 データ I / F 回路
- 24 プログラム ROM
- 25 ローダ
- 26 エラー訂正回路

30

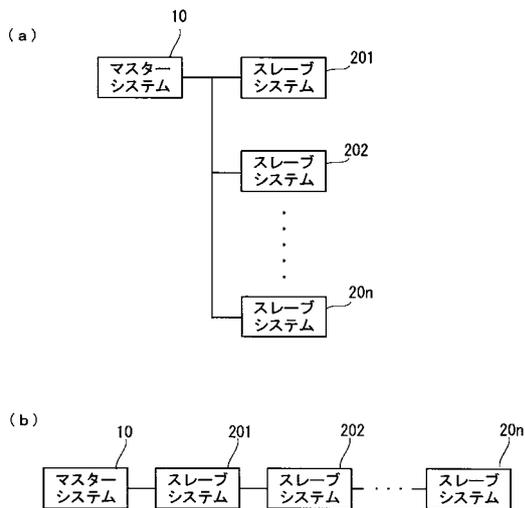
【図1】



【図2】



【図3】



【図4】

