

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第5016604号
(P5016604)

(45) 発行日 平成24年9月5日(2012.9.5)

(24) 登録日 平成24年6月15日(2012.6.15)

(51) Int.Cl. F 1
G 0 6 F 11/00 (2006.01) G 0 6 F 9/06 6 3 0 B

請求項の数 2 (全 19 頁)

(21) 出願番号	特願2008-537391 (P2008-537391)	(73) 特許権者	000005016 パイオニア株式会社 神奈川県川崎市幸区新小倉1番1号
(86) (22) 出願日	平成18年10月4日(2006.10.4)	(74) 代理人	100104190 弁理士 酒井 昭徳
(86) 国際出願番号	PCT/JP2006/319881	(72) 発明者	奥田 順二 埼玉県川越市山田字西町25番地1 パイ オニア株式会社 川越事業所内
(87) 国際公開番号	W02008/041337		
(87) 国際公開日	平成20年4月10日(2008.4.10)		
審査請求日	平成21年3月11日(2009.3.11)		
		審査官	林 毅

最終頁に続く

(54) 【発明の名称】 情報処理装置および情報処理方法

(57) 【特許請求の範囲】

【請求項1】

プログラムを保存するメモリおよびプログラムを実行する制御手段を備える情報処理装置であって、

前記メモリは、

実行対象となるプログラムのアドレスを保存する V e c t o r 領域、

リセットスタート時に実行される初期起動プログラムを保存する I P L 領域、

実行対象プログラムを保存する実行対象プログラム領域と、リセットスタート時に前記実行対象プログラムを実行させるための仮想初期起動プログラムを保存する仮想 V e c t o r 領域と、から構成される有効プログラム領域、および

前記有効プログラム領域のプログラムを更新するための更新用プログラムを一時保存する更新用プログラム保存領域、

を有し、

前記制御手段は、

リセットスタートを受け付けた場合に、前記 V e c t o r 領域を参照して前記 I P L 領域の初期起動プログラムを実行して、前記更新用プログラム保存領域に更新用プログラムが不足なく保存されて更新準備が完了しているか否かを判定し、

前記更新準備が完了している場合には前記更新用プログラムを前記有効プログラム領域に複製し、前記更新準備が完了していない場合には前記仮想 V e c t o r 領域の仮想初期起動プログラムを実行して前記実行対象プログラム領域の実行対象プログラムを実行する

10

20

ことを特徴とする情報処理装置。

【請求項 2】

プログラムを保存するメモリを備える情報処理装置が実行する情報処理方法であって、前記メモリは、

実行対象となるプログラムのアドレスを保存する V e c t o r 領域、

リセットスタート時に実行される初期起動プログラムを保存する I P L 領域、

実行対象プログラムを保存する実行対象プログラム領域と、リセットスタート時に前記実行対象プログラムを実行させるための仮想初期起動プログラムを保存する仮想 V e c t o r 領域と、から構成される有効プログラム領域、および

前記有効プログラム領域のプログラムを更新するための更新用プログラムを一時保存する更新用プログラム保存領域、

10

を有し、

リセットスタートを受け付けた場合に、前記 V e c t o r 領域を参照して前記 I P L 領域の初期起動プログラムを実行して、前記更新用プログラム保存領域に更新用プログラムが不足なく保存されて更新準備が完了しているか否かを判定し、

前記更新準備が完了している場合には前記更新用プログラムを前記有効プログラム領域に複製し、前記更新準備が完了していない場合には前記仮想 V e c t o r 領域の仮想初期起動プログラムを実行して前記実行対象プログラム領域の実行対象プログラムを実行する制御工程を含むことを特徴とする情報処理方法。

【発明の詳細な説明】

20

【技術分野】

【0001】

この発明は、実行対象となるプログラムの実行および更新が可能な情報処理装置および情報処理方法に関する。ただし、この発明の利用は、上述した情報処理装置および情報処理方法には限られない。

【背景技術】

【0002】

従来より、一つのマイコン（マイクロコンピュータ）における制御プログラムを更新する際には、データ設定器などの外部機器を使用して、データの更新をおこなっている。近年では、制御プログラムの更新の際に、他のマイコンから割り込み処理によって更新エラーを起こさないようにするため、制御プログラムの更新中に、他のマイコンからの割り込み処理を禁止する提案がされている（たとえば、下記特許文献 1 参照。）。

30

【0003】

ここで、図 1 を用いて、実行対象となるプログラムが実行可能に保存されたプログラム実行マイコン（マイクロコンピュータ）におけるプログラムの更新の概要について説明する。図 1 は、プログラム実行マイコンにおけるプログラムの更新の概要について示す説明図である。

【0004】

図 1 において、プログラム更新システム 100 は、プログラム実行マイコン 101 と、更新用プログラム送信側マイコン 102 と、フラッシュメモリ 103 と、から構成されている。

40

【0005】

プログラム実行マイコン 101 は、バッファ領域 110 と、プログラム領域 111 と、I P L (I n i t i a l P r o g r a m L o a d e r) 領域 112 と、V e c t o r 領域 113 と、から構成されている。

【0006】

バッファ領域 110 は、後述する更新用プログラム送信側マイコン 102 から送信される更新用プログラムを一時的に保存する。更新用プログラムは、たとえば、マイコン間通信によって更新用プログラム送信側マイコン 102 からバッファ領域 110 に送信され、プログラム実行マイコン 101 の実行対象となるプログラムの更新用のプログラムである

50

。

【0007】

バッファ領域110に一時保存された更新用プログラムは、図示しないCPU(Central Processing Unit)が、後述するIPL領域112を参照することによって、プログラム領域111に複製(コピー)される。換言すれば、後述するプログラム領域111に保存されたプログラムa, b, cは、更新用プログラムのプログラム領域111に複製されることによって更新がおこなわれる構成である。

【0008】

プログラム領域111は、プログラム実行マイコン101の実行対象となるプログラムa, b, cを保存する。具体的には、たとえば、プログラム領域111は、プログラムaを保存するA領域121と、プログラムbを保存するB領域122と、プログラムcを保存するC領域123と、から構成されている。

10

【0009】

各プログラムa, b, cは、たとえば、図示しないCPUによって読み込まれて実行される構成でもよい。具体的には、たとえば、プログラムa, b, cは、図示しないCPUが、後述するVector領域113に格納された、プログラムa, b, cのアドレスを参照することによって実行される。なお、図1の説明では、3つの領域(A領域121, B領域122, C領域123)を用いて説明したが、領域の数量および実行対象となるプログラムの数はこれに限るものではない。

【0010】

IPL領域112は、リセットスタート時に図示しないCPUによって最初に実行されるプログラムを保存する領域である。具体的には、たとえば、IPL領域112は、バッファ領域110に一時保存された更新用プログラムをプログラム領域111に複製するか、プログラム領域111に保存された各プログラムa, b, cを実行するかを決定するプログラムを保存する。

20

【0011】

Vector領域113は、イベント発生時にプログラム実行マイコン101の実行対象となるプログラムのアドレスを格納する。具体的には、たとえば、図示しないCPUは、リセットスタート時には、Vector領域113を参照してIPL領域112に保存されたプログラムのアドレスを読み込む。また、図示しないCPUは、たとえば、通常起動時には、Vector領域113を参照してプログラム領域111に保存された各プログラムa, b, cのアドレスを読み込む。

30

【0012】

更新用プログラム送信側マイコン102は、プログラム実行マイコン101に対して、プログラム実行マイコン101の実行対象となるプログラムa, b, cの更新用プログラムを送信する。具体的には、たとえば、更新用プログラム送信側マイコン102は、フラッシュメモリ103が保存している更新用プログラムを読み込んで、読み込んだ更新用プログラムをプログラム実行マイコン101に送信する。

【0013】

フラッシュメモリ103は、各種データを保存することができる記録媒体で、図1では、プログラム実行マイコン101の実行対象となるプログラムa, b, cの更新用プログラムを保存する。具体的には、たとえば、フラッシュメモリ103は、更新用プログラム送信側マイコン102に対して、保存している更新用プログラムを出力する。

40

【先行技術文献】

【特許文献】

【0014】

【特許文献1】特開2004-118369号公報

【発明の概要】

【発明が解決しようとする課題】

【0015】

50

しかしながら、上記従来技術では、マイコンにおける制御プログラムを更新する場合、更新専用の外部機器が必要となるため、マイコンの利用者は、更新専用の外部機器を用意する負担が生じてしまうという問題が一例として挙げられる。

【0016】

また、更新専用の外部機器を有するマイコンの製造元などで、制御プログラムを更新する場合、マイコンの利用者は、製造元に対するマイコンの送付や受け取りに関する手間が発生するばかりか、更新に要する時間を浪費してしまうという問題が一例として挙げられる。

【0017】

一方、更新専用の外部機器を用いずに、マイコン間通信などによって受信される更新プログラムによって、実行対象となるプログラムを更新する場合、書き換え（複製）の途中で電源障害などによって更新動作が中断されると、マイコンが起動できなくなってしまうという問題が一例として挙げられる。

【0018】

特に、Vector領域に格納された、実行対象となるプログラムのアドレスを書き換える場合、書き換える途中で更新動作が中断されると、中断復旧後の起動時にVector領域を参照できないため、マイコンが起動できなくなってしまうという問題が一例として挙げられる。

【0019】

また、Vector領域に格納された、実行対象となるプログラムのアドレスを書き換えずに、実行対象となるプログラムだけを更新する場合、Vector領域に格納された、実行対象となるプログラムのアドレスが固定されるため、初期設定によるプログラムの配置を変更できず、汎用性に欠けるとともにプログラム領域の容量を効率的に使用できないという問題が一例として挙げられる。

【課題を解決するための手段】

【0020】

上述した課題を解決し、目的を達成するため、請求項1の発明にかかる情報処理装置は、プログラムを保存するメモリおよびプログラムを実行する制御手段を備える情報処理装置であって、前記メモリは、実行対象となるプログラムのアドレスを保存するVector領域、リセットスタート時に実行される初期起動プログラムを保存するIPL領域、実行対象プログラムを保存する実行対象プログラム領域と、リセットスタート時に前記実行対象プログラムを実行させるための仮想初期起動プログラムを保存する仮想Vector領域と、から構成される有効プログラム領域、および前記有効プログラム領域のプログラムを更新するための更新用プログラムを一時保存する更新用プログラム保存領域、を有し、前記制御手段は、リセットスタートを受け付けた場合に、前記Vector領域を参照して前記IPL領域の初期起動プログラムを実行して、前記更新用プログラム保存領域に更新用プログラムが不足なく保存されて更新準備が完了しているか否かを判定し、前記更新準備が完了している場合には前記更新用プログラムを前記有効プログラム領域に複製し、前記更新準備が完了していない場合には前記仮想Vector領域の仮想初期起動プログラムを実行して前記実行対象プログラム領域の実行対象プログラムを実行することを特徴とする。

【0021】

また、請求項2の発明にかかる情報処理方法は、プログラムを保存するメモリを備える情報処理装置が実行する情報処理方法であって、前記メモリは、実行対象となるプログラムのアドレスを保存するVector領域、リセットスタート時に実行される初期起動プログラムを保存するIPL領域、実行対象プログラムを保存する実行対象プログラム領域と、リセットスタート時に前記実行対象プログラムを実行させるための仮想初期起動プログラムを保存する仮想Vector領域と、から構成される有効プログラム領域、および前記有効プログラム領域のプログラムを更新するための更新用プログラムを一時保存する更新用プログラム保存領域、を有し、リセットスタートを受け付けた場合に、前記Vec

10

20

30

40

50

t o r 領域を参照して前記 I P L 領域の初期起動プログラムを実行して、前記更新用プログラム保存領域に更新用プログラムが不足なく保存されて更新準備が完了しているか否かを判定し、前記更新準備が完了している場合には前記更新用プログラムを前記有効プログラム領域に複製し、前記更新準備が完了していない場合には前記仮想 V e c t o r 領域の仮想初期起動プログラムを実行して前記実行対象プログラム領域の実行対象プログラムを実行する制御工程を含むことを特徴とする。

【図面の簡単な説明】

【 0 0 2 4 】

【図 1】プログラム実行マイコンにおけるプログラムの更新の概要について示す説明図である。

10

【図 2】本実施の形態にかかる情報処理装置の機能的構成の一例を示すブロック図である。

【図 3】本実施の形態にかかる情報処理装置の処理の内容を示すフローチャートである。

【図 4】本実施例にかかるワンチップマイコンのハードウェア構成の一例を示すブロック図である。

【図 5】本実施例にかかるワンチップマイコンにおける実行対象プログラムの更新の概要を示す説明図である。

【図 6】本実施例にかかる有効プログラム領域に保存されるプログラムについて示す説明図である。

【図 7】本実施例にかかるワンチップマイコンにおける更新用プログラムの受け付け処理手順を示すフローチャートである。

20

【図 8】本実施例にかかるワンチップマイコンの起動処理手順を示すフローチャートである。

【図 9】本実施例にかかるワンチップマイコンにおける割り込み処理（図 8 のステップ S 8 1 2）手順を示すフローチャートである。

【発明を実施するための形態】

【 0 0 2 5 】

以下に添付図面を参照して、この発明にかかる情報処理装置および情報処理方法の好適な実施の形態を詳細に説明する。

【 0 0 2 6 】

30

（実施の形態）

（情報処理装置の機能的構成）

図 2 を用いて、本実施の形態にかかる情報処理装置の機能的構成について説明する。図 2 は、本実施の形態にかかる情報処理装置の機能的構成の一例を示すブロック図である。

【 0 0 2 7 】

図 2 において、プログラム実行領域に保存された実行対象プログラムを実行する情報処理装置 2 0 0 は、更新用プログラム入力部 2 0 1 と、一時保存部 2 0 2 と、判定部 2 0 3 と、生成部 2 0 4 と、更新情報保存部 2 0 5 と、再稼働情報入力部 2 0 6 と、判断部 2 0 7 と、更新部 2 0 8 と、実行部 2 0 9 と、から構成されている。

【 0 0 2 8 】

40

更新用プログラム入力部 2 0 1 は、実行対象プログラムに対する更新用プログラムの入力を受け付ける。具体的には、たとえば、更新用プログラムの入力は、図示しない更新用プログラムが送信可能な外部装置から入力を受け付けることとしてもよい。

【 0 0 2 9 】

更新用プログラムは、たとえば、情報処理装置 2 0 0 における不図示のメモリ領域などに割り当てられたプログラム実行領域に複製（コピー）されて実行対象プログラムとなる。また、更新用プログラムは、実行対象プログラムとして実行される際に、初期起動プログラムによって実行される仮想初期起動プログラムを含む構成でもよい。具体的には、たとえば、仮想初期起動プログラムは、プログラム実行領域が備える仮想プログラム領域に保存され、初期起動プログラムによって実行される。ここで、初期起動プログラムは、た

50

例えば、情報処理装置 100 の再稼働の際に起動されるプログラムである。

【0030】

さらに、更新用プログラム入力部 201 は、後述する判定部 203 によって判定された結果、更新用プログラムに過不足がある場合、更新用プログラムの出力元に対して取得要求をおこなうことによって更新用プログラムの入力を受け付けることとしてもよい。

【0031】

一時保存部 202 は、更新用プログラム入力部 201 によって入力された更新用プログラムを一時保存領域に一時保存する。具体的には、たとえば、一時保存部 202 は、情報処理装置 200 における不図示のメモリ領域などに割り当てられたバッファ領域など、データの一時保存が可能な領域に更新用プログラムを一時保存することとしてもよい。

10

【0032】

判定部 203 は、一時保存部 202 によって一時保存された更新用プログラムに過不足がないか否かを判定する。具体的には、たとえば、判定部 203 は、複数回に分割されて入力される更新用プログラムが、すべて一時保存領域に一時保存されたか否かを判定することとしてもよい。

【0033】

生成部 204 は、判定部 203 によって判定された結果、更新用プログラムに過不足がない場合、プログラム実行領域に保存された実行対象プログラムの更新準備が完了した旨を示す更新情報を生成する。更新情報は、たとえば、実行対象プログラムの更新準備が完了した旨を示す代わりに、更新準備が完了していない旨を示す情報であってもよい。

20

【0034】

更新情報保存部 205 は、生成部 204 によって生成された更新情報を更新情報保存領域に保存する。具体的には、たとえば、更新情報保存部 205 は、情報処理装置 200 における不図示のメモリ領域などに割り当てられた更新情報保存領域に更新情報を保存する。なお、更新情報保存部 205 は、生成部 204 によって更新準備が完了した旨を示す更新情報の代わりに、更新準備が完了していない旨を示す情報を保存していてもよい。

【0035】

再稼働情報入力部 206 は、自装置の再稼働に関する再稼働情報の入力を受け付ける。再稼働情報の入力は、たとえば、リセットスタートの処理要求などの入力であってもよい。

30

【0036】

判断部 207 は、再稼働情報入力部 206 によって再稼働情報が入力された場合、初期起動プログラム保存領域に保存された初期起動プログラムを起動させて更新情報を参照することによって、更新準備が完了しているか否かを判断する。具体的には、たとえば、判断部 207 は、情報処理装置 200 における不図示のメモリ領域などに割り当てられた初期起動プログラム保存領域に保存された初期起動プログラムを起動させる構成である。

【0037】

更新部 208 は、判断部 207 によって判断された結果、更新準備が完了している場合、更新用プログラムをプログラム実行領域に複製することによって実行対象プログラムを更新する。具体的には、たとえば、実行対象プログラムの更新は、更新準備が完了している場合、一時保存部 202 によって保存された更新用プログラムを実行対象プログラムに対して上書き保存する構成でもよい。

40

【0038】

また、更新部 208 は、たとえば、更新用プログラムを実行対象プログラムに上書きする際、更新用プログラムに含まれる仮想初期起動プログラムを用いて、プログラム実行領域が備える仮想プログラム保存領域の仮想初期起動プログラムもあわせて上書きすることとなる。

【0039】

また、更新部 208 は、プログラム実行領域に複製した更新用プログラムを検証することによって、実行対象プログラムを実行可能な形式に更新することとしてもよい。具体的

50

には、たとえば、更新用プログラムの検証は、更新用プログラムが実行対象プログラムとして過不足なく複製されたか否かを確認する構成でもよい。

【0040】

実行部209は、判断部207によって判断された結果、更新準備が完了していない場合、プログラム実行領域に保存された実行対象プログラムを実行する。具体的には、たとえば、実行部209は、仮想初期起動プログラムの実行にしたがって、プログラム実行領域に保存された実行対象プログラムを実行する。すなわち、実行部209は、更新準備が完了していない場合、初期起動プログラム保存領域に保存された初期起動プログラムの起動によって実行される仮想初期起動プログラムによって、実行対象プログラムを実行することができる。

10

【0041】

換言すれば、初期起動プログラムは、再稼働情報が入力されると起動して、更新準備が完了しているか否かを判断させる。そして、初期起動プログラムは、判断の結果、更新準備が完了している場合は、更新用プログラムを用いて、実行対象プログラムを更新させることとなる。一方、初期起動プログラムは、判断の結果、更新準備が完了していない場合は、仮想プログラム保存領域に保存されている仮想初期起動プログラムを実行させることにより、実行対象プログラムを実行させることとなる。

【0042】

このように、情報処理装置200は、実行対象プログラムを更新する場合、仮想プログラム保存領域に保存された仮想初期起動プログラムを、実行対象プログラムとあわせて更新する構成である。また、情報処理装置200は、初期起動プログラムによって仮想初期起動プログラムを実行させることによって、実行対象プログラムを実行することができる。

20

【0043】

したがって、情報処理装置200は、仮想プログラム保存領域に仮想初期起動プログラムを保存することによって、実行対象プログラムを更新する際に初期起動プログラムを更新しなくても、実行対象プログラムの更新あるいは実行を確実におこなうことができる。換言すれば、実行対象プログラムの更新中に、電源障害などに起因して更新処理に不具合が発生しても、電源復旧時には、再度初期起動プログラムの実行によって、実行対象プログラムの更新あるいは実行をおこなうことができる。

30

【0044】

(情報処理装置200の処理の内容)

つぎに、図3を用いて本実施の形態にかかる情報処理装置200の処理の内容について説明する。図3は、本実施の形態にかかる情報処理装置の処理の内容を示すフローチャートである。図3のフローチャートにおいて、まず、更新用プログラム入力部201によって、更新用プログラムの入力を受け付けたか否かを判断する(ステップS301)。

【0045】

ステップS301において、更新用プログラムの入力を受け付けた場合(ステップS301:Yes)は、一時保存部202によって、一時保存領域にステップS301において入力された更新用プログラムを一時保存する(ステップS302)。

40

【0046】

また、ステップS301において、更新用プログラムの入力を受け付けない場合(ステップS301:No)は、ステップS306へ移行して、再稼働情報入力部206によって、再稼働情報の入力を受け付けたか否かを判断する(ステップS306)。

【0047】

つぎに、判定部203によって、ステップS302において一時保存された更新用プログラムに過不足がないか否かを判定する(ステップS303)。

【0048】

そして、生成部204によって、ステップS303において判定された結果に基づいて、更新情報を生成する(ステップS304)。更新情報は、たとえば、ステップS303

50

において判定された結果、更新用プログラムに過不足がない場合、プログラム実行領域に保存された実行対象プログラムの更新準備が完了した旨を示す情報である。また、更新情報は、たとえば、ステップS303において判定された結果、更新用プログラムに過不足がある場合、更新準備が完了していない旨を示す情報であってもよい。

【0049】

つづいて、更新情報保存部205によって、更新情報保存領域にステップS304において生成された更新情報を保存する(ステップS305)。

【0050】

つぎに、再稼働情報入力部206によって、再稼働情報の入力を受け付けたか否かを判断する(ステップS306)。再稼働情報は、たとえば、リセットスタートの処理要求などであってもよい。

【0051】

ステップS306において、再稼働情報の入力を受け付けた場合(ステップS306: Yes)は、判断部207によって、初期起動プログラム保存領域に保存された初期起動プログラムを起動させて更新情報を参照することによって、更新準備が完了しているか否かを判断する(ステップS307)。

【0052】

また、ステップS306において、再稼働情報の入力を受け付けない場合(ステップS306: No)は、ステップS301に戻って処理を繰り返す。

【0053】

ステップS307において、更新準備が完了している場合(ステップS307: Yes)は、更新部208によって、更新用プログラムをプログラム実行領域に複製することによって実行対象プログラムを更新する(ステップS308)。そして、実行部209によって、ステップS308において更新された実行対象プログラムを実行して(ステップS309)、一連の処理を終了する。

【0054】

また、ステップS307において、更新準備が完了していない場合(ステップS307: No)は、ステップS309へ移行して、実行部209によって、プログラム実行領域に保存された実行対象プログラムを実行する(ステップS309)。そして、一連の処理を終了する。

【0055】

換言すれば、ステップS307～ステップS309では、更新準備が完了している場合は、実行対象プログラムを更新してから、更新した実行対象プログラムを実行する。また、更新準備が完了していない場合は、実行対象プログラムを更新せずに、プログラム実行領域に保存されている実行対象プログラムを実行することとしてもよい。

【0056】

以上説明したように、本実施の形態によれば、一時保存部に保存された更新用プログラムによって実行対象プログラムを更新するため、専用の外部機器を用意しなくても実行対象プログラムの更新をおこなうことができる。

【0057】

また、本実施の形態によれば、更新用プログラムの更新準備が完了してから、実行対象プログラムを更新するため、更新用プログラムの入力途中に電源障害などが発生しても、実行対象プログラムに対する弊害の防止を図ることができる。

【0058】

また、本実施の形態によれば、仮想初期起動プログラムを含む更新用プログラムによって実行対象プログラムを更新するため、初期起動プログラムの更新をしなくてもよい。したがって、更新途中に電源障害などが発生しても、復旧後には初期起動プログラムによって自装置を起動することができる。

【0059】

すなわち、本実施の形態によれば、電源障害の復旧後であっても、初期起動プログラム

10

20

30

40

50

によって実行対象プログラムの更新あるいは実行をおこなうことができる。具体的には、たとえば、本実施の形態によれば、電源障害の復旧後に再稼働情報が入力されると、電源障害によって更新途中であるため、初期起動プログラムによって更新を継続あるいはやり直しすることができる。また、本実施の形態によれば、電源障害があっても更新が終了していれば、初期起動プログラムによって仮想初期起動プログラムを実行させて、実行対象プログラムを実行することができる。

【0060】

くわえて、実行対象プログラムは、仮想初期起動プログラムによって実行されることとなるため、汎用性に富んだ実行対象プログラムの配置を設計でき、プログラム実行領域の利用の効率化を図ることができる。換言すれば、初期起動プログラムの代わりに、実行対象プログラムとあわせて更新される仮想初期起動プログラムによって、実行対象プログラムを実行するため、初期起動プログラムによって制限される配置ではなく、仮想初期起動プログラムに設定される自由度のある配置を実現できる。

10

【0061】

また、本実施の形態によれば、更新用プログラムを検証することによって、実行対象プログラムを実行可能な形式に更新するため、的確な実行対象プログラムの更新を図ることができる。

【実施例】

【0062】

以下に、本発明の実施例について説明する。本実施例では、たとえば、フラッシュROM (Read Only Memory) 内蔵のワンチップマイコンによって、本発明の情報処理装置を実施した場合の一例について説明する。

20

【0063】

(ワンチップマイコンのハードウェア構成)

まず、図4を用いて、本実施例にかかるワンチップマイコンのハードウェア構成について説明する。図4は、本実施例にかかるワンチップマイコンのハードウェア構成の一例を示すブロック図である。

【0064】

図4において、ワンチップマイコン400は、CPU401と、フラッシュROM402と、RAM (Random Access Memory) 403と、入出力I/F (インターフェース) 404と、を備えている。また、各構成部401~404は、バス410によってそれぞれ接続されている。

30

【0065】

まず、CPU401はワンチップマイコン400の全体の制御を司る。具体的には、たとえば、CPU401は、フラッシュROM402に記録されたプログラムを読み込み、RAM403をワークエリアとして読み込んだプログラムを実行する構成でもよい。

【0066】

フラッシュROM402は、ブートプログラムや実行対象となる実行対象プログラムなどの各種プログラムを記録している。詳細は図5を用いて説明するが、フラッシュRAM402は、たとえば、各種プログラムを書き換え可能に記録し、電源を切ってもデータが消えない不揮発性半導体メモリなどである。

40

【0067】

RAM403は、CPU401のワークエリアとして使用される。具体的には、たとえば、RAM403は、データの書き込みと読み出しが随時可能な半導体メモリでもよく、CPU401によって読み込まれたプログラムを実行するためのワークエリアとなる。

【0068】

入出力I/F404は、図示しない外部機器に接続され、ワンチップマイコン400とCPU401とのインターフェースとして機能する。入出力I/F404は、さらに、有線あるいは無線などを介してインターネットなどの通信網に接続され、この通信網とCPU401とのインターフェースとしても機能する。

50

【0069】

また、詳細は図5に後述するが、入出力I/F404は、ワンチップマイコン400の外部に接続されたマイコンから、マイコン間通信によってフラッシュROM402に記録された実行対象プログラムに対する更新用プログラムの入力を受け付けることとしてもよい。

【0070】

なお、図4では説明を省略したが、本実施例にかかるワンチップマイコン400は、CPU401、フラッシュROM402、RAM403、入出力I/F404のほかに、タイマ、LCD(Liquid Crystal Display)ドライバ、MMU(Memory Management Unit)、DMAC(Direct Memory Access Contoroller)、A/D(Analog/Digital)コンバータなどの周辺機能を備えていてもよい。

10

【0071】

なお、実施の形態にかかる情報処理装置200の機能的構成のうち、更新用プログラム入力部201や再稼働情報入力部206は入出力I/F404によって、一時保存部202や判定部203や生成部204や更新情報保存部205や判断部207や更新部208や実行部209はCPU401がフラッシュROM402やRAM403に記録された各種プログラムを実行することによって、それぞれその機能を実現する。

【0072】

(ワンチップマイコン400における実行対象プログラムの更新の概要)

20

つぎに、図5を用いて、本実施例にかかるワンチップマイコン400における実行対象プログラムの更新の概要について説明する。図5は、本実施例にかかるワンチップマイコンにおける実行対象プログラムの更新の概要を示す説明図である。

【0073】

図5において、プログラム更新システム500は、ワンチップマイコン400と、更新用プログラム送信側マイコン502と、フラッシュメモリ503と、から構成されている。

【0074】

ワンチップマイコン400における図4に示したフラッシュROM402は、有効プログラム情報保存領域510と、空き領域511と、更新用プログラム保存領域512と、有効プログラム領域513と、IPL領域514と、Vector領域515と、から構成されている。

30

【0075】

有効プログラム情報保存領域510は、後述する有効プログラム領域513に記録された実行対象プログラムの更新準備が完了した旨を示す更新情報を保存する領域である。具体的には、たとえば、更新情報は、後述する更新用プログラム保存領域512に、更新用プログラム送信側マイコン502から複数回に分割されて入力される更新用プログラムに過不足がない場合、更新準備が完了したこととする情報でもよい。

【0076】

また、更新情報は、たとえば、実行対象プログラムの更新準備が完了した旨を示す代わりに、更新準備が完了していない旨を示す情報であってもよい。

40

【0077】

空き領域511は、本実施例におけるフラッシュROM402の未使用領域であり、初期設定によって変更することができる。

【0078】

更新用プログラム保存領域512は、後述する更新用プログラム送信側マイコン502から送信される更新用プログラムを一時的に保存する領域である。更新用プログラムは、たとえば、マイコン間通信によって更新用プログラム送信側マイコン502から図示しない入出力I/F404を介して入力され、ワンチップマイコン400の実行対象となる実行対象プログラムの更新用のプログラムである。

50

【 0 0 7 9 】

また、更新用プログラムは、たとえば、図示しないCPU401によって有効プログラム情報保存領域510に保存された更新情報が参照されて更新準備が完了している場合に、有効プログラム領域513に複製（コピー）される。換言すれば、更新用プログラムは、たとえば、CPU401によって実行対象プログラムに対して上書き保存され、その実行対象プログラムが更新されることとなる。

【 0 0 8 0 】

このように、更新情報を参照して実行対象プログラムの更新をおこなう構成とすれば、更新用プログラムが不完全な状態で実行対象プログラムを更新して不具合を引き起こすことを防ぐことができる。すなわち、電源障害など、何らかの障害によって更新用プログラムの一時保存がストップしてしまった場合、リセットスタートによってストップした時点の更新用プログラムによって中途半端な状態の実行対象プログラムの更新はおこなわずに、一つ前の手順に戻って、更新用プログラムの一時保存を開始することができる。

10

【 0 0 8 1 】

また、更新用プログラムは、実行対象プログラムとして実行される際に、初期起動プログラムによって実行される仮想初期起動プログラムを含む構成でもよい。初期起動プログラムは、たとえば、再稼働（リセットスタート）の際に起動されるプログラムである。

【 0 0 8 2 】

有効プログラム領域513は、ワンチップマイコン400の実行対象となる実行対象プログラムを保存する領域である。詳細は図6を用いて説明するが、有効プログラム領域513に保存された実行対象プログラムは、複数でもよく、図示しないCPU401に読み込まれて実行されることとしてもよい。

20

【 0 0 8 3 】

また、有効プログラム領域513に更新用プログラムが複製された場合、CPU401は、複製された更新用プログラムを検証することによって、実行対象プログラムを実行可能な形式に更新することとしてもよい。具体的には、たとえば、更新用プログラムの検証は、更新用プログラムが実行対象プログラムとして過不足なく複製されたか否かを確認する構成でもよい。

【 0 0 8 4 】

IPL領域514は、リセットスタート時に図示しないCPU401によって最初に実行される初期起動プログラムを保存する領域である。具体的には、たとえば、IPL領域514は、更新用プログラム保存領域512に一時保存された更新用プログラムを有効プログラム領域513に複製するか、有効プログラム領域513に保存された各実行対象プログラムを実行するかを決定するプログラムを保存する。

30

【 0 0 8 5 】

Vector領域515は、イベント発生時にワンチップマイコン400の実行対象となるプログラムのアドレスを格納する。具体的には、たとえば、図示しないCPU401は、リセットスタート時には、Vector領域515を参照してIPL領域514に保存されたプログラムのアドレスを読み込む。また、CPU401は、たとえば、通常起動時には、Vector領域515を参照して有効プログラム領域513に保存された各実行対象プログラムのアドレスを読み込むこととしてもよい。

40

【 0 0 8 6 】

このように、IPL領域514およびVector領域515に保存されたデータは、更新用プログラムによって書き換えをおこなわないため、更新用プログラムの一時保存中や、実行対象プログラムの更新中に障害が発生しても、リセットスタートによって、CPU401がIPL領域514およびVector領域515に保存されたデータを参照すれば、ワンチップマイコン400の動作を正常に再開することができる。

【 0 0 8 7 】

更新用プログラム送信側マイコン502は、ワンチップマイコン400に対して、ワンチップマイコン400の実行対象となる実行対象プログラムの更新用プログラムを送信す

50

る。具体的には、たとえば、更新用プログラム送信側マイコン502は、フラッシュメモリ503が保存している更新用プログラムを読み込んで、読み込んだ更新用プログラムをワンチップマイコン400に送信する。

【0088】

フラッシュメモリ503は、各種データを保存することができる記録媒体で、図5では、ワンチップマイコン400の実行対象となる実行対象プログラムの更新用プログラムを保存する。具体的には、たとえば、フラッシュメモリ503は、更新用プログラム送信側マイコン502に対して、保存している更新用プログラムを出力する。

【0089】

ここで、図6を用いて、有効プログラム領域513に保存されるプログラムについて説明する。図6は、本実施例にかかる有効プログラム領域に保存されるプログラムについて示す説明図である。

10

【0090】

図6において、有効プログラム領域513は、仮想初期起動プログラムを保存する仮想Vector領域601と、実行対象となる実行対象プログラムa, b, cを保存する実行対象プログラム領域602と、から構成されている。

【0091】

仮想Vector領域601に保存された仮想初期起動プログラムは、リセットスタート時に、通常起動である場合に図示しないCPU401によって読み込まれる。仮想初期起動プログラムは、たとえば、通常起動時に実行対象となる実行対象プログラムa, b, cを実行させるプログラムで、CPU401に対して、実行対象プログラム領域602に保存された実行対象プログラムa, b, cのアドレスを指定する構成でもよい。

20

【0092】

具体的には、たとえば、CPU401は、リセットスタート時に、Vector領域515を参照してIPL領域514に保存されたプログラムのアドレスを読み込むことによって、更新用プログラムの複製（実行対象プログラムの更新）、あるいは、実行対象プログラムを実行する。

【0093】

より具体的には、たとえば、CPU401は、有効プログラム情報保存領域510に保存された更新情報に基づいて、更新用プログラムの複製、あるいは、実行対象プログラムの実行を選択し、実行対象プログラムを実行する場合、Vector領域515を参照して、仮想Vector領域601のアドレスを読み込む。

30

【0094】

そして、CPU401は、仮想Vector領域601に保存された仮想初期起動プログラムによってアドレスが指定される実行対象プログラムa, b, cを実行することとしてもよい。

【0095】

このように、IPL領域514およびVector領域515は更新せずに、有効プログラム領域513における仮想初期起動プログラムを保存する仮想Vector領域601および実行対象プログラム領域602を更新することで、実行対象プログラムa, b, cの配置に関する制限を最小限に抑制することができる。

40

【0096】

換言すれば、たとえば、図1に示すように、仮想初期起動プログラムがないと、Vector領域113を更新しない場合は、各実行対象プログラムa, b, cを保存する領域に余裕を持たせた配置としなければならない。一方で、Vector領域113を更新用プログラムによって更新する場合は、プログラムの更新中に電源障害などが発生すると、Vector領域113に不具合が発生してしまい、プログラム実行マイコン101が動作しなくなってしまう。

【0097】

(ワンチップマイコン400の処理の内容)

50

つぎに、図7～図9を用いて、本実施例にかかるワンチップマイコン400の処理の内容について説明する。図7は、本実施例にかかるワンチップマイコンにおける更新用プログラムの受け付け処理手順を示すフローチャートである。図7のフローチャートにおいて、まず、入出力I/F404によって、更新用プログラム送信側マイコン502からプログラム更新準備完了通知を受け付けたか否かを判断する(ステップS701)。プログラム更新準備完了通知は、たとえば、更新用プログラム送信側マイコン502がフラッシュメモリ503やその他外部機器から更新用プログラムを取得した場合、更新用プログラム送信側マイコン502からワンチップマイコン400に送信されることとしてもよい。

【0098】

ステップS701において、プログラム更新準備完了通知を受け付けるのを待って、受け付けた場合(ステップS701: Yes)は、ワンチップマイコン400は、プログラム更新モードへ移行し(ステップS702)、CPU401によって、更新用プログラム保存領域512を初期化する(ステップS703)。更新用プログラム保存領域512の初期化は、たとえば、更新用プログラム保存領域512の容量を空きとすることでもよく、すなわち、更新用プログラム保存領域512に中途半端なデータがあった場合は消去することとしてもよい。

【0099】

つぎに、入出力I/F404によって、ステップS703において更新用プログラム保存領域512が初期化されたら、更新用プログラム送信側マイコン502に対して、受け入れ可能通知を出力する(ステップS704)。

【0100】

そして、入出力I/F404によって、ステップS704において出力された受け入れ可能通知に応じて更新用プログラム送信側マイコン502から出力される、更新用プログラムのデータをパケット受信する(ステップS705)。

【0101】

つぎに、入出力I/F404によって、ステップS705においてパケット受信された更新用プログラムのデータについて、更新用プログラム送信側マイコン502に対して、パケット受信完了通知を出力する(ステップS706)。

【0102】

つづいて、入出力I/F404によって、更新用プログラムをすべて受信したか否かを判断する(ステップS707)。更新用プログラムをすべて受信したか否かの判断は、たとえば、ステップS705においてパケット受信される最後の更新用プログラムのデータに付与されるフラグを受信した場合に、すべて受信したと判断する構成でもよい。

【0103】

ステップS707において、更新用プログラムをすべて受信した場合(ステップS707: Yes)は、CPU401によって、更新用プログラム保存領域512に一時保存される更新用プログラムのチェックサムを確認する(ステップS708)。チェックサムは、たとえば、複数回に分割されてパケット受信される更新用プログラムが、すべて更新用プログラム保存領域512に一時保存されたか否かを判定することとしてもよい。

【0104】

また、ステップS707において、更新用プログラムをすべて受信しない場合(ステップS707: No)は、ステップS705に戻って処理を繰り返す。

【0105】

つぎに、ステップS708においてチェックサムを確認したら、CPU401によって、有効プログラム情報保存領域510に更新用プログラム準備完了状態であることを書き込む(ステップS709)。更新用プログラム準備完了状態の書き込みは、たとえば、実行対象プログラムの更新準備が完了した旨を示す更新情報を有効プログラム情報保存領域510に保存することである。

【0106】

なお、図7のフローチャートでは説明を省略するが、ステップS708においてチェッ

10

20

30

40

50

クサムが確認できない場合は、エラーとして、再度更新用プログラム送信側マイコン502に対して、更新用プログラムを要求することとしてもよい。

【0107】

そして、入出力I/F404によって、更新用プログラム送信側マイコン502に対してシャットダウン要求を出力する(ステップS710)。その後、入出力I/F404によって、更新用プログラム送信側マイコン502からステップS710におけるシャットダウン要求に応じたシャットダウン完了通知を受信して(ステップS711)、一連の更新用プログラムの受け付け処理を終了する。

【0108】

つづいて、図8を用いて、本実施例にかかるワンチップマイコン400の起動処理手順について説明する。図8は、本実施例にかかるワンチップマイコンの起動処理手順を示すフローチャートである。図8のフローチャートにおいて、まず、入出力I/F404によって、リセットスタートを受け付ける(ステップS801)。

【0109】

ステップS801において、リセットスタートを受け付けると、CPU401によって、IPL領域514に保存されたIPLを起動して(ステップS802)、有効プログラム情報保存領域510を参照する(ステップS803)。

【0110】

そして、CPU401によって、ステップS803において参照された有効プログラム情報保存領域510の更新情報より、更新用プログラム準備は完了しているか否かを判断する(ステップS804)。更新用プログラム準備は、たとえば、有効プログラム情報保存領域510に更新用プログラム準備完了状態であることが書き込まれている(たとえば、図7に示したステップS709)か否かを判断することとしてもよい。

【0111】

ステップS804において、更新用プログラム準備が完了している場合(ステップS804:Yes)は、CPU401によって、コピープログラムを起動させる(ステップS805)。コピープログラムは、たとえば、更新用プログラム保存領域512に一時保存された更新用プログラムを有効プログラム領域513に保存された実行対象プログラムなど各種プログラムに上書き保存するプログラムでもよい。

【0112】

そして、ステップS805において起動されたコピープログラムによるコピーが終了するのを待って(ステップS806:No)、終了した場合(ステップS806:Yes)は、CPU401によって、有効プログラム領域513にコピーされたプログラムを検証する(ステップS807)。プログラムの検証は、たとえば、更新用プログラムが実行対象プログラムとして過不足なくコピーされたか否かを確認する構成でもよい。

【0113】

そして、ステップS807においてプログラムの検証をおこなったら、CPU401によって、有効プログラム情報保存領域510に通常起動状態であることを書き込んで(ステップS808)、ステップS801へ戻って処理を繰り返す。

【0114】

なお、図8のフローチャートでは説明を省略するが、ステップS807においてプログラムの検証の結果、実行対象プログラムとして過不足なくコピーされていない場合は、エラーとして、更新用プログラム保存領域512からのコピーを再度おこなうこととしてもよい。

【0115】

また、ステップS804において、更新用プログラム準備が完了していない場合(ステップS804:No)は、CPU401によって、ステップS803において参照された有効プログラム情報保存領域510への書き込みが通常起動状態であるか否かを判断する(ステップS809)。

【0116】

10

20

30

40

50

ステップS809において、通常起動状態である場合（ステップS809：Yes）は、CPU401によって、有効プログラム領域513のプログラムを起動させる（ステップS810）。

【0117】

具体的には、たとえば、有効プログラム領域513のプログラムの起動は、CPU401によって、Vector領域515を参照して、仮想Vector領域601のアドレスを読み込む。そして、CPU401によって、仮想Vector領域601に保存された仮想初期起動プログラムによってアドレスが指定される実行対象プログラムa, b, cを実行することとしてもよい。

【0118】

また、ステップS809において、通常起動状態でない場合（ステップS809：No）は、ステップS801へ戻って処理を繰り返す。

【0119】

つぎに、入出力I/F404によって、割り込み処理を受け付けたか否かを判断する（ステップS811）。ステップS811において、割り込み処理を受け付けた場合（ステップS811：Yes）は、CPU401によって、割り込み処理をおこなって（ステップS812）、プログラムが終了したか否かを判断する（ステップS813）。なお、割り込み処理の詳細は、図9に後述する。

【0120】

また、ステップS811において、割り込み処理を受け付けない場合（ステップS811：No）は、ステップS813へ移行して、プログラムが終了したか否かを判断する（ステップS813）。

【0121】

ステップS813において、プログラムが終了した場合（ステップS813：Yes）は、ステップS801へ戻って処理を繰り返す。また、ステップS813において、プログラムが終了しない場合（ステップS813：No）は、ステップS810へ戻って処理を繰り返す。

【0122】

つぎに、図9を用いて、本実施例にかかるワンチップマイコン400における割り込み処理（図8のステップS812）手順について説明する。図9は、本実施例にかかるワンチップマイコンにおける割り込み処理（図8のステップS812）手順を示すフローチャートである。図9のフローチャートにおいて、まず、CPU401によって、仮想Vector領域601を参照する（ステップS901）。

【0123】

つぎに、CPU401によって、ステップS901において参照された仮想Vector領域601に保存された仮想初期起動プログラムを起動させて、実行対象プログラム領域602を参照する（ステップS902）。

【0124】

そして、CPU401によって、ステップS902において参照された実行対象プログラム領域602に保存されたプログラムの中から実行対象となるプログラムを起動させて（ステップS903）、一連の割り込み処理を終了する。

【0125】

以上説明したように、本実施例にかかるワンチップマイコンによれば、更新用プログラム保存領域に保存された更新用プログラムによって実行対象プログラムを更新するため、更新専用の外部機器を用意することなく、適切に実行対象プログラムを更新することができる。

【0126】

また、有効プログラム情報保存領域に更新用プログラム準備完了状態であることが書き込まれてから、実行対象プログラムを更新する構成であるため、更新用プログラム準備完了状態となる前に電源障害などが発生しても、復旧後にはワンチップマイコンは正常に起

10

20

30

40

50

動することができる。

【0127】

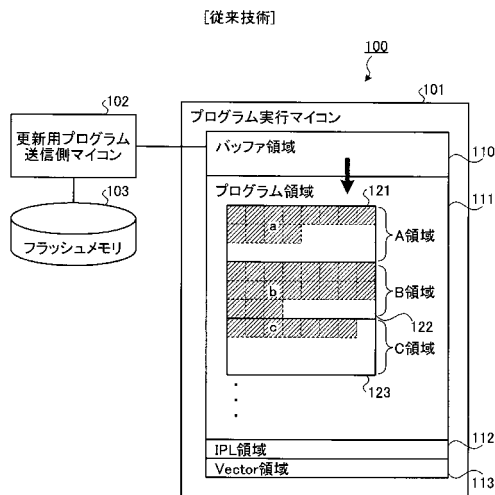
さらに、Vector領域の書き換えをおこなわずに、仮想Vector領域を含む有効プログラム領域を更新する構成であるため、更新途中で電源障害などが発生しても、復旧後にはワンチップマイコンは正常に起動することができ、有効プログラム領域の効率的な利用を図ることができる。

【符号の説明】

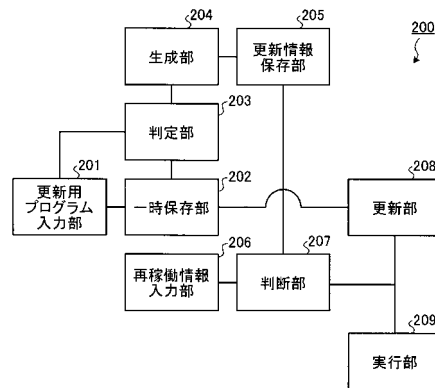
【0129】

- 200 情報処理装置
- 201 更新用プログラム入力部
- 202 一時保存部
- 203 判定部
- 204 生成部
- 205 更新情報保存部
- 206 再稼働情報入力部
- 207 判断部
- 208 更新部
- 209 実行部

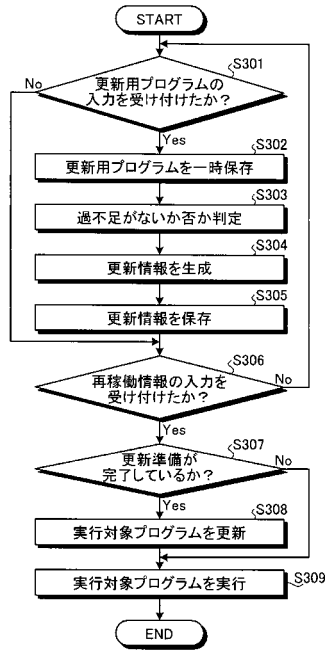
【図1】



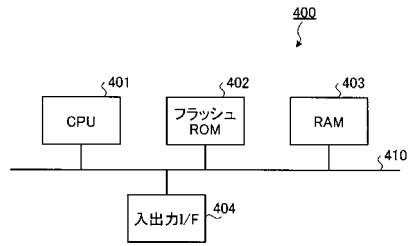
【図2】



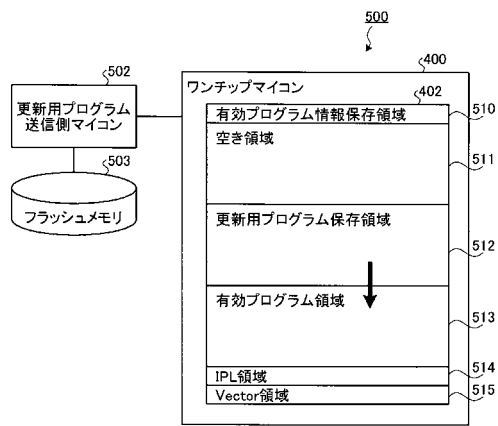
【図3】



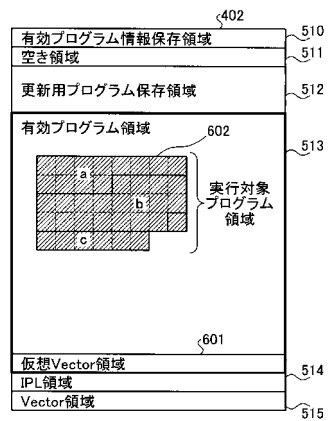
【図4】



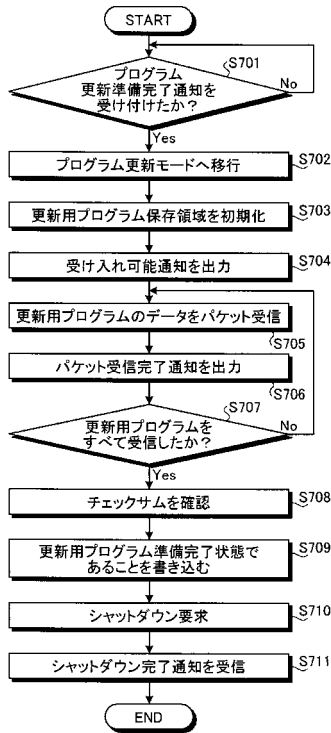
【図5】



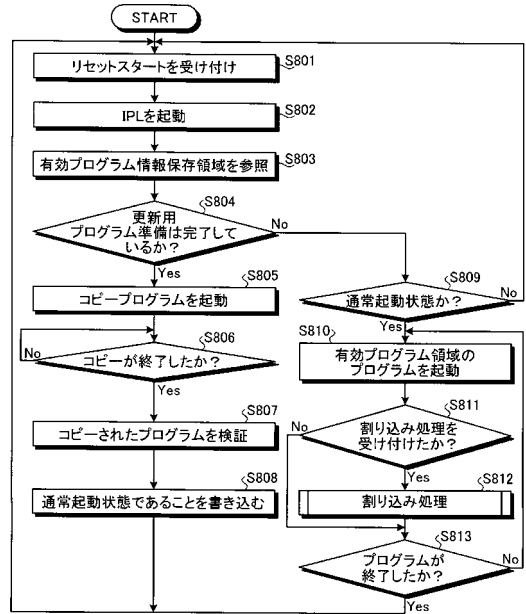
【図6】



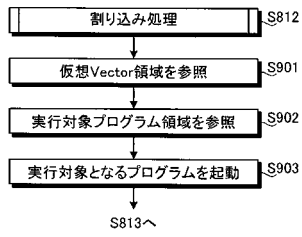
【図7】



【図8】



【図9】



フロントページの続き

- (56)参考文献 特開2002-063044(JP,A)
特開2002-007128(JP,A)
特開平09-238213(JP,A)
特開2002-244877(JP,A)
特開2006-060355(JP,A)
特開2005-332228(JP,A)
国際公開第97/038367(WO,A1)

(58)調査した分野(Int.Cl., DB名)

G06F 11/00