

(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局



(10) 国際公開番号

WO 2010/140403 A1

(43) 国際公開日

2010 年 12 月 9 日(09.12.2010)

PCT

- (51) 国際特許分類:
G06F 9/445 (2006.01) G06F 9/54 (2006.01)
- (21) 国際出願番号: PCT/JP2010/053627
- (22) 国際出願日: 2010 年 3 月 5 日(05.03.2010)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (30) 優先権データ:
特願 2009-132708 2009 年 6 月 2 日(02.06.2009) JP
- (71) 出願人 (米国を除く全ての指定国について): 株式会社ユビキタス(Ubiquitous Corporation) [JP/JP]; 〒1600023 東京都新宿区西新宿 1-23-7 新宿ファーストウエスト 16 F Tokyo (JP).
- (72) 発明者; および
- (75) 発明者/出願人 (米国についてのみ): 橋本 健一 (HASHIMOTO Kenichi) [JP/JP]; 〒1600023 東京都新宿区西新宿 1-23-7 新宿ファーストウエスト 16 F 株式会社ユビキタス内 Tokyo (JP).
- (74) 代理人: 稲本 義雄, 外 (INAMOTO Yoshio et al.); 〒1600023 東京都新宿区西新宿 7 丁目 1 1 番 1 8 号 7 1 1 ビルディング 4 階 Tokyo (JP).
- (81) 指定国 (表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) 指定国 (表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM, TR), OAPI (BF, BJ,

[続葉有]

(54) Title: PROGRAM, CONTROL METHOD, AND CONTROL DEVICE

(54) 発明の名称: プログラム、制御方法、並びに制御装置

図4

0	~207-0
1	~207-1
2	~207-2
3	~207-3
4	~207-4
5	~207-5
6	~207-6
7	~207-7
8	~207-8
9	~207-9
10	~207-10
11	~207-11
12	~207-12
13	~207-13
14	~207-14
15	~207-15
16	~207-16
17	~207-17
18	~207-18
19	~207-19
20	~207-20
...	
n	~207-n

(57) Abstract: Provided is a program, control method, and control device that can shorten start-up time. Page table entry is rewritten for a Memory Management Unit (MMU) table, on a computer system equipped with an MMU, so that a page fault will occur at every page, for all the pages necessary for the operation of a software program. Upon start-up, the stored memory image is loaded in page units for page faults that have occurred on the RAM to be accessed. Loading of unnecessary pages will not be executed, because such loading was executed, and the start-up time can be shortened worth that time. This program, control method, and control device can be applied to personal computers, and electronic devices equipped with built-in type computers.

(57) 要約: 本発明は、起動時間を短縮することができるプログラム、制御方法、並びに制御装置に関する。Memory Management Unit(MMU)を搭載したコンピュータシステム上で、MMUのテーブルに対し、ソフトウェアの動作に必要なページに対して、全てのページにてページフォルトが発生するようにページテーブルエントリが書き換えられる。起動時は、アクセスするRAMに対して発生したページフォルトに対し、保存したメモリーイメージのページ単位の読み込みが実行される。このような読み込みが行われことで、不要なページの読み込みが行われず、その分の起動時間を短縮することが可能となる。本発明は、パーソナルコンピュータや組み込み型のコンピュータを備える電子機器に適用できる。

WO 2010/140403 A1

WO 2010/140403 A1



CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, 添付公開書類:
TD, TG).

— 国際調査報告 (条約第 21 条(3))

明 細 書

発明の名称： プログラム、制御方法、並びに制御装置

技術分野

[0001] 本発明はプログラム、制御方法、並びに制御装置に関し、特に、ソフトウェアの起動を制御するときに好適なプログラム、制御方法、並びに制御装置に関する。

背景技術

[0002] パーソナルコンピュータにおいて、OS (Operating System) を起動させ、所望のソフトウェアを動作させるまでには数分単位の起動時間が必要であった。これを高速に起動させる手法として、ハイバネーション (hibernation) と呼ばれる手法が存在する (例えば、特許文献1 参照)。

[0003] 特許文献1 には、起動後のCPU (central processing unit) やI/O (input/output) レジスタ、RAM (Random Access Memory) イメージが、ハードディスクドライブ (HDD) やフラッシュメモリに格納されることが記載されている。そして、次に起動されるときに、格納されているRAMイメージが復帰され、その後、CPUやI/Oレジスタが再設定されることが記載されている。このように起動することで、OSの起動を高速化させることが、特許文献1 では提案されている。このような提案に基づくハイバネーションと称される手法は、既にパーソナルコンピュータで適用されている。

[0004] また、組み込み型のコンピュータ、例えば、テレビジョン受像器、ハードディスクレコーダといった電子機器に組み込まれているコンピュータにおいても、ハイバネーションの手法が応用されている。

先行技術文献

特許文献

[0005] 特許文献1：特開2005-149225号公報

特許文献2：特開2007-334383号公報

発明の概要

発明が解決しようとする課題

- [0006] ハイバネーションを適用し、OSを起動させる場合とOSを通常通りに起動させる場合とを比較した場合、ハイバネーションを適用してOSを起動させる方が、より高速に起動させることができる。しかしながら、RAMの容量増加に伴って保存すべきRAMイメージのサイズも増加してしまい、起動時に、そのRAMイメージの展開時間も増加してしまう。結果として、RAMの大容量化に伴い高速起動が困難になってしまう。
- [0007] また、パーソナルコンピュータは、CPUの性能も比較的高いため、仮にRAMイメージのサイズが増加してしまっても、そのRAMイメージを処理する性能が確保される。しかしながら、組み込み型のコンピュータの場合、CPUの性能が比較的低いものが使われることが多い。そのため、組み込み型のコンピュータの場合、RAMイメージが増加すれば、ハイバネーションの手法を適用しても、起動時の速度は低下してしまう。すなわち、組み込み型のコンピュータの場合、RAMイメージの増加による速度の低下は、より顕著に表れてしまう。
- [0008] また、RAMイメージを圧縮することで、RAMイメージのサイズを小さくすることも提案されているが、起動時に伸張する処理が必要となる。この伸張処理にかかるCPUへの負荷や、伸張処理にかかる時間を考慮すると、起動の高速化という点では効果的な方法ではない。
- [0009] このようなことを考慮し、特許文献2では、ハイバネーションのイメージの全ての転送を完了する前に、OSの実行を開始する手法が提案されている。しかしながら、この手法では、特別なハードウェアを搭載し、先行転送するページを予め特定しておく必要があるため、その特別なハードウェアの分だけコストが高くなるなどの問題点があった。
- [0010] 本発明は、このような状況に鑑みてなされたものであり、起動時間を短縮することができるようにするものである。

課題を解決するための手段

- [0011] 本発明の一側面のプログラムは、メモリを管理する機能を有する制御装置

に、所定のソフトウェアの動作に必要なページに対して、全てのページでページフォルトが発生するようにページテーブルエントリを書き換え、前記ソフトウェアが起動時に、前記ページテーブルエントリで、ページフォルトが発生し、そのページフォルトが発生したページを順次読み出すステップを含む。

[0012] 前記所定のソフトウェアが起動された後、前記ページテーブルエントリを書き換え、その起動時のデータ、プログラムコード、テーブル、ページフォルトハンドラ、割り込みベクタ、およびレジスタを、前記メモリに記憶させるようにすることができる。

[0013] 前記メモリのうち、書き換えの対象となる前記ページテーブルエントリを記憶しているのはRAMであり、順次読み出される前記ページを記憶しているのは不揮発メモリであるようにすることができる。

[0014] 組み込み型のコンピュータが読み込むようにすることができる。

[0015] 本発明の一側面の制御方法は、メモリを管理する機能を有する制御装置の制御方法において、所定のソフトウェアの動作に必要なページに対して、全てのページでページフォルトが発生するようにページテーブルエントリを書き換え、前記ソフトウェアが起動時に、前記ページテーブルエントリで、ページフォルトが発生し、そのページフォルトが発生したページを順次読み出すステップを含む。

[0016] 本発明の一側面の制御装置は、メモリを管理する機能を有する制御装置において、所定のソフトウェアの動作に必要なページに対して、全てのページでページフォルトが発生するようにページテーブルエントリを書き換える書き換え手段と、前記ソフトウェアが起動時に、前記ページテーブルエントリで、ページフォルトが発生し、そのページフォルトが発生したページを順次読み出す読み出し手段とを備える。

[0017] 本発明の一側面のプログラム、制御方法、並びに制御装置は、所定のソフトウェアの動作に必要なページに対して、全てのページでページフォルトが発生するようにページテーブルエントリが書き換えられ、ソフトウェアが起

動時に、ページテーブルエントリで、ページフォルトが発生し、そのページフォルトが発生したページが順次読みだされる。

発明の効果

[0018] 本発明の一側面によれば、OSの起動時間を短縮することが可能となる。

図面の簡単な説明

[0019] [図1]本発明を適用した情報処理装置の一実施の形態の構成を示す図である。

[図2]MMUのモデルを示す図である。

[図3]ディスクリプタについて説明するための図である。

[図4]物理ページの読み込みについて説明するための図である。

[図5]物理ページの読み込みについて説明するための図である。

[図6]物理メモリマップについて説明するための図である。

[図7]起動の処理について説明するためのフローチャートである。

[図8]起動の処理について説明するためのフローチャートである。

[図9]起動の処理について説明するためのフローチャートである。

[図10]起動の処理について説明するためのフローチャートである。

[図11]起動の処理について説明するためのフローチャートである。

発明を実施するための形態

[0020] 以下に、本発明の実施の形態について図面を参照して説明する。

[0021] まず、本発明の概略を説明する。本発明は、Memory Management Unit(以下、MMUと略記する)を搭載したCPU (central processing unit) 上で動作するOS (Operating System) やアプリケーション等のソフトウェアを高速に起動させるための手法である。

[0022] 高速起動の対象となるソフトウェアが、一度、通常通りの方法で起動され、その状態のRAM (Random Access Memory) イメージが不揮発メモリ等に保存される。RAMイメージが、不揮発メモリ等に保存されるとき、MMUのテーブルが書き換えられ、全てのページでページフォルトが発生するように変更される。対象とされるソフトウェアには、ページフォルトハンドラが用意されており、ページフォルトが発生した場合、そのページフォルトが発

生したページのみが、不揮発メモリからロードされる。

[0023] 任意のソフトウェアが起動され、プログラムコードが実行されたり、そのソフトウェアが動作に必要なデータを読み出すために、RAMにアクセスしたりした場合、毎回ページフォルトが発生し、必要なページが不揮発メモリからRAMに逐次ロードされることになる。これにより、従来のハイバネーション起動とは異なり、全てのRAMイメージを不揮発メモリからメインRAMに予めロードする必要がなくなり、動作に必要な必要最小限のRAMイメージのみをロードすることが可能になる。よって、高速に所望のソフトウェアを起動・動作させることが可能となる。

[0024] 本発明を使用することで、従来は数十秒～数分を要していたOSやソフトウェア（以下、単にソフトウェアとする）の起動時間を数秒程度に短縮することが可能となることが、本出願人により確認されている。以下に、具体的に説明する。

[0025] [情報処理装置の構成について]

図1は、本発明を適用した情報処理装置の一実施の形態の構成を示す図である。本発明を適用した情報処理装置は、パーソナルコンピュータ（PC）に適用できることは勿論のこと、組み込み型のコンピュータを有する装置にも適用できる。組み込み型のコンピュータを含む装置としては、テレビジョン受像器、ハードディスクレコーダといった電子機器がある。ここでは、ハードディスクレコーダに対して本発明を適用したときを例にあげて説明する。

[0026] 図1は、本発明を適用した情報処理装置としてのハードディスクレコーダの構成を示す図である。図に示したハードディスクレコーダ100は、CPU101、RAM102、ROM（Read Only Memory）103、不揮発メモリ104、MPEG（Moving Picture Experts Group）エンコード・デコード部105、チューナ106、HDDインターフェース107、HDD108、I/O部109、起動モード切替部110を備える。

[0027] CPU101は、ハードディスクレコーダ100の各部を制御する。このCPU101は、Memory Management Unit(以下、MMUと記述する)を搭載

し、RAM 102を小分割して小単位（ページ）で管理できる仕組みを有している。なお、ここでは、MMU 131がCPU 101に含まれるとして説明を続けるが、MMU 131が、CPU 101に含まれず、外部に備えられている構成とすることも可能である。また、MMU 131の形式には特に制限はないが、ページ単位にアクセスの許可・禁止の属性が設定可能で、アクセス禁止のページにアクセスした場合は、ページフォルトの例外が発生できる構成とされている。また、CPU 101のMMU 131は、4キロバイト（以下、4KBと記述する）を1ページとして管理するとして説明を続ける。

[0028] RAM 102は、SRAM（Static Random Access Memory）やDRAM（Dynamic Random Access Memory）などで構成することが可能である。RAM 102は、CPU 101が使用する主記憶装置として機能し、そのような機能を有していれば、RAM 102として用いることが可能である。

[0029] ROM 103は、FLASH ROMやMask ROMなどの読み込み専用のメモリである。ROM 103は、OSやアプリケーションソフトウェアが格納され、そのような格納が行えれば、どのような種類のROMが用いられても本発明においては良い。

[0030] 不揮発メモリ 104は、ハードディスクレコーダ 100の電源が切られても、記憶している内容が保持されるメモリである。例えば、FLASH ROMや、バックアップ機能付きのSRAM、DRAMなどで構成することが可能である。

[0031] 不揮発メモリ 104には、後述するソフトウェアが起動した後に、RAM 102に格納されたソフトウェアのメモリイメージが格納される。そのため、不揮発メモリ 104の容量は、RAM 102の容量以上の容量とされることが好ましい。ただし、データ圧縮等によりデータを縮小するようにした場合、不揮発メモリ 104の容量は、RAM 102の容量以下の容量でも良い。また、不揮発メモリ 104は、HDD 108と兼用することも可能である（HDD 108を不揮発メモリ 108として利用することも可能である）。

[0032] MPEGエンコード・デコード部 105は、動画の圧縮・伸張を行う。動

画は、チューナ106を介して供給される。チューナ106は、ユーザの指示に基づき、複数の番組（動画）から、1つの動画を選択し、MPEGエンコード・デコード部105に供給する。MPEGエンコード・デコード部105は、必要に応じ、HDDインターフェース107を介して、HDD108にチューナ106からのデータを供給したり、HDD108からのデータをHDDインターフェース107を介して、受信したりする。また、その際、必要に応じ、エンコードまたはデコードの処理を実行する。

[0033] I/O部109は、起動モード切替部110の状態をCPU101が読みとるために設けられている。以下に説明するソフトウェアは、高速起動イメージを取得するための通常起動モードと、高速起動イメージが取得された後の起動モードとしての高速起動モードを有する。I/O109と起動モード切替部110は、これらの起動モードの切り替えに用いられる。通常起動モードと高速起動モードの切り換えのために、起動モード切替部110をスイッチで構成することが可能である。また、通常起動モードと高速起動モードの切り換えのために、起動モード切替部110を、ブートローダ等からコマンドで切り替える構成とすることも可能である。

[0034] また、一度高速起動のイメージが作成されれば、通常起動モードを不要とすることができるため、高速起動モードのみが実装され、通常起動モードと高速起動モードの切り換えを不要とする構成としても良い。そのような構成とした場合、I/O部109と起動モード切替部110を省略した構成とすることも可能である。

[0035] [MMUについて]

図2は、CPU101が装備するMMU131のモデルを示す図である。図2に示したMMU131は、32bitクラス以上のCPU101が装備するモデルである。MMU131の構成や、物理アドレス、仮想アドレス、各テーブルのインデックスに使用するbit数、テーブルの段数等は、CPU101のメーカーに依存するものであるが、特にメーカーのアーキテクチャに依存はしない。ここでは説明の都合上、32bit等の具体的な数字を例にあげて説明

するが、その数値は、本発明の適用範囲の限定を示すものではない。

[0036] MMU 131は、最終的な物理ページを指し示すテーブル内エントリの属性にアクセス許可もしくはアクセス許可相当の機能を実装し、不許可の場合にアクセスしたときには、ページフォルトもしくはページフォルト相当の例外処理が発生できる機能を少なくとも有する。

[0037] MMUレジスタ200は、MMU 131に装備されるレジスタである。このレジスタにレベル1ディスクリプタテーブル201の先頭アドレスが代入される。レベル1ディスクリプタテーブル201は、メモリ上に置かれたレベル1メモリテーブルである。物理アドレス空間が32bitの場合、1つのアドレスを指し示すのに32bit、即ち4バイトが使用されるため、レベル1ディスクリプタテーブル201のサイズは、12bit空間＝4KB空間×4バイトで16KBのサイズとなる。

[0038] 仮想アドレス202は、レベル1メモリテーブルのインデックスとして使われる仮想アドレスの31bitから20bitを示す。VAはVirtual Address（仮想アドレス）を意味する。所定の仮想アドレスにアクセスがあった場合、レベル1ディスクリプタテーブル201の先頭アドレスから仮想アドレスの31bitから20bitがインデックスとされ、レベル1ディスクリプタ203にアクセスがされる。

[0039] 物理アドレス空間が32bitのCPU 101である場合、1つのアドレスを示すのに32bit、即ち4バイトが使用されるため、レベル1ディスクディスクリプタのアドレスは以下のような式になる。

レベル1ディスクリプタ203のアドレス

$$\begin{aligned} &= \text{レベル1ディスクリプタテーブル201の先頭アドレス} \\ &\quad + \text{仮想アドレス202 (VA[31:20])} \times 4 \end{aligned}$$

[0040] レベル1ディスクリプタ203は、レベル2ディスクリプタテーブル204の先頭アドレスを指し示すポインタや属性から構成されるディスクリプタである。レベル2ディスクリプタテーブル204の先頭アドレスから仮想アドレスの19bitから12bitがインデックスとされ、レベル2ディスクリプタ

206にアクセスされる。

- [0041] 物理アドレス空間が32bitのCPU101である場合、1つのアドレスを示すのに32bit、即ち4バイトが使用されるため、レベル2ディスクディスクリプタのアドレスは以下のような式になる。

レベル2ディスクリプタ206のアドレス

$$\begin{aligned} &= \text{レベル2ディスクリプタテーブル204の先頭アドレス} \\ &\quad + \text{仮想アドレス205 (VA[19:12])} \times 4 \end{aligned}$$

- [0042] レベル2ディスクリプタ206は、4KBの物理ページ207を示すポインタや属性から構成されるディスクリプタである。物理ページ207は、最終的に仮想アドレスから物理アドレスに変換された1ページの物理メモリである。物理ページ207の4KB内のアドレスは、仮想アドレス208 (VA[11:0])で指定される。

- [0043] 図3は、レベル1ディスクリプタ203やレベル2ディスクリプタ206の一例である。図3におけるディスクリプタはあくまでも一例であり、本発明は、このような一例で示した特定のCPU101やアーキテクチャに依存することを示すものではない。

- [0044] ベースアドレス301は、次のテーブルや物理ページ先頭アドレスを指すポインタである。属性302乃至304は、それぞれ、実行可・不可、特権モード・ユーザモードといった属性を示す属性bitである。アクセス許可bit305は、このディスクリプタが示す物理ページに対してのアクセスが許可されているか否かを示すbitである。このアクセス許可bit305で、アクセスを禁止するように設定されている物理ページにアクセスがあった場合、一般的には、ページフォルトと呼ばれる例外処理、即ち割込み処理が実行され、そのような機構が必要である。よって、CPU101のアーキテクチャに依存はしないが、ページフォルト、もしくはページフォルト相当の機能は、CPU101に装備されている必要がある。

- [0045] [物理ページについて]

図4は、物理ページ207が、RAM102（図1）上に並んでいる状態を

擬似的に示している。RAM 102には、1ページ分の物理ページ207が、物理ページ207-0から順に、物理ページ207-nまで配列されている。MMU 131搭載のCPU 101上でソフトウェアが動作される場合、1ページの物理ページ207は、このように4KB乃至64KB単位で1ページとして管理されるように構成されていることが多い。

[0046] 図5は、ソフトウェアが所定の動作状態であるときの物理ページの使用状況について説明するための図である。図5中、数字が記載されているページが使用されているページであり、数字が記載されていないページは使用されていないページを示す。ソフトウェアが使うプログラムコードやデータが、図4に示したように、全ての領域を使っていたとしても、ソフトウェアの状態をある単位時間でみた場合、図5で示すように使われていることが多い。すなわち、使用されているページと使用されていないページとがあり、全てのページが使用されているわけではない。

[0047] このソフトウェアが、所定の動作状態のときには、物理ページ207-0、物理ページ207-2、物理ページ207-4、物理ページ207-5、物理ページ207-9、物理ページ207-16、および物理ページ207-18が使用される。すなわち、図4に示したように、RAM 102には、1ページ分の物理ページ207が、物理ページ207-0から順に、物理ページ207-nまで配列されているが、そのうち所定のソフトウェアが、所定の動作状態のときには、全てのページが使われるわけではなく、図5に示したように、複数のページのみが使用される。

[0048] 従来のハイバネーション起動による起動の場合、ソフトウェアが動作する前に、図4に示したように、物理ページ207-0から物理ページ207-nまで、順次読み出しが行われてから復帰動作が始まり、所定の動作状態にされる。しかしながら実際に、そのソフトウェアが所定の動作状態になるためには、図5に示したように、所定の複数の物理ページ207が読み出されるだけでよい。そこで、本発明においては、後述するように読み出しを制御することで、図5に示したように、必要とされる物理ページ207のみが読

み出されるようにする。

[0049] 従来のハイバネーション起動による起動の場合、図4に示したように物理ページ207-0から順次読み出しが行われるため、換言すれば、必要のない物理ページ207も読み出されるため、読み出しに時間がかかり、結果として、所定のソフトウェア（OSなども含む）の起動が遅くなるということがあった。しかしながら、本発明によれば、図5に示したように、必要とされる物理ページ207のみが読み出されるため、読み出しにかかる時間を短縮することができ、所定のソフトウェア（OSなども含む）の起動を早くすることが可能となる。

[0050] 図6は、ソフトウェアの物理メモリマップを示す図である。なお、以下に説明する処理は、CPU101やOSに依存するもので、これらの機能が実装されていれば構成やメモリ配置に制限はなく、以下の説明だけに、本発明が適用されることを示すものではない。

[0051] 不揮発メモリ104は、電源を切っても記憶内容が保持されるメモリである。図6に示した例では、不揮発メモリ104としてFLASH ROMをイメージしている。不揮発メモリ104は、メインメモリ上にマッピングされているが、電源を切っても内容が保持され、且つ、RAM102以上の容量を装備している。しかしながら、本発明を適用できる不揮発メモリ104は、I/O経由でのアクセス等、必ずしもメモリマップ上にマッピングされている必要はなく、そのアーキテクチャに制限はない。

[0052] データ401は、プログラムコード402が使用する読み書きができるデータ領域である。データ401は物理ページ207として、ある特定の大きさに分割されて格納される。データ401は、読み書きできる必要があるのでRAM102上に置かれることが好ましい。

[0053] プログラムコード402は、起動・実行させる所望のプログラムを示す。一般的なパーソナルコンピュータでWindows（登録商標）やLinuxと言ったOSが搭載され、その上で動作するソフトウェアである場合、プログラムとは、そのOSとソフトウェアを含む。プログラムコード402は、物理ページ

207として、ある特定の大きさに分割されて格納される。このプログラムコード402は、RAM102またはROM103上に置かれる。

[0054] MMUテーブル403は、図2に示したレベル1ディスクリプタテーブル201およびレベル2ディスクリプタテーブル204を示している。ページフォルトハンドラ404は、MMU131のレベル2ディスクリプタ206がアクセス禁止の属性であり、ページフォルトが発生したとき、割込ベクタ経由で例外処理を行うためのプログラムである。ここではプログラムコード402と分けて、ページフォルトハンドラ404を記述しているが、プログラムコード402に含まれる場合もある。

[0055] 割込ベクタ405は、一般的なCPUが持つ割込ベクタである。ページフォルトが発生した場合、この割込ベクタ内のページフォルトにプログラムコードがジャンプし、結果的にページフォルトハンドラ404が呼び出される。

[0056] データ401、プログラムコード402、MMUテーブル403、および割込ベクタ405の論理アドレスに対する物理アドレスは、任意のアドレスにマッピングできる。

[0057] イメージ保存プログラム406は、所望のプログラムが起動された後、所望の状態でメモリイメージが不揮発メモリ104に保存されるようにするためのプログラムである。イメージ保存プログラム406の論理アドレスと物理アドレスは、同一アドレスにマッピングされる必要がある。

[0058] イメージ復帰プログラム407は、イメージ保存プログラム406の処理により保存された物理メモリイメージを、必要に応じて物理ページ単位に、データ401やプログラムコード402を、対応する物理ページに、不揮発メモリ104から読み込み、復帰させるためのプログラムである。イメージ復帰プログラム407の論理アドレスと物理アドレスは、同一アドレスにマッピングされる必要がある。

[0059] ブートローダ408は、電源投入もしくはリセット後に最初に起動されるブートローダである。主にブートローダ408は、起動に必要な最低限のI

／〇の初期化を行う。このような構成をソフトウェアは有する。

[0060] [ソフトウェアの動作について]

次に、本発明を適用したソフトウェアの動作について説明する。まず、概要を説明し、その後詳細を説明する。本発明によれば、ソフトウェアの局所性を利用し、ソフトウェアを高速に起動することができる。ソフトウェアには〇Ｓ等も含まれる。例えば、容量４ＧＢのＲＡＭを備えるハードウェア上で、所定のソフトウェアを動作させると仮定する。その所定のソフトウェアのプログラムおよびデータの容量の合計が、仮に４ＧＢであったとする。一般的にソフトウェアは、様々なモードや機能を有し、所定の単一の機能だけで、全容量の４ＧＢを使う可能性は極めて低い。

[0061] 例えばソフトウェアが、起動後、特定の状態でユーザからのキー入力进行待つとする。一般的にはハードウェアのリセット後、ブートローダが起動され、このソフトウェアが立ち上がってユーザからのキー入力待ちの状態とされる。このソフトウェアを既知の技術であるハイバネーションを適用して高速に起動させた場合、メモリイメージを作成する準備として、ユーザからのキー入力待ちの状態でＣＰＵや各Ｉ／〇のレジスタが保存され、プログラムコードやデータの合計４ＧＢが、何らかの不揮発メモリに格納されることになる。起動時はこの逆で、通常の起動プロセスは通らず、４ＧＢのメモリが展開され、ＣＰＵやＩ／〇のレジスタが復歸され、キー入力の処理に戻るようになる。

[0062] 上記ソフトウェアにおいて、「ユーザからの入力を待つ」という状態を考える。この状態では、キー入力の処理が繰り返されており、このようなキー入力に係わる処理に係わるプログラムコードやデータは比較的小さい。本発明は、この原理を利用して高速起動を実現する。動作は大きく分けると以下のようなになる。

[0063] （Ａ）通常起動モードで〇Ｓや所望のソフトウェアが起動され、そのソフトウェアが所望の状態にされる

 （Ｂ）イメージ保存プログラムが起動され、その起動されたイメージ保存プ

ログラムにより、MMU 131の全てのページテーブルに対してのアクセスを禁止するために、アクセス禁止を示す情報に所定の情報が書き換えられた後、上記（A）の状態のメモリイメージがレジスタに保存されて終了される

（C）次回以降は、高速起動モードにされることで、（A）の所望の状態で起動される

[0064] このような高速起動を実現するにあたり、基本的な準備として、通常通りにソフトウェアが起動され、そのソフトウェアが所望の状態にされ、その後、メモリイメージやレジスタが保存される。高速起動時は、ハイバネーション起動とは異なり、全てのメモリイメージがメインメモリに展開されるのではなく、実際に使われるメモリ、即ちプログラムコードやデータが、必要に応じて一部分のみ小刻みに展開される。

[0065] 小刻みにプログラムコードやメモリが、メインメモリに展開される方法として、本実施の形態としてはCPUが持つMemory Management Unit(MMU 131)が利用される例をあげて説明している。OSが、MMU 131を使用する場合もあるが、OSが利用する前に、本発明が適用されたソフトウェアが元に戻し、OSは、本発明が適用されたソフトウェアがMMU 131を操作したことには関知しない。

[0066] 具体的には、メモリイメージが保存される前に、MMU 131のテーブルの内容が書き換えられ、全てのページがアクセス禁止に設定される。また、本発明が適用されたソフトウェアは、前記ページをアクセス禁止にしたというマークを付与する機能も有する。

[0067] 仮に、所望のソフトウェアがOS上で動作し、そのOSがMMU 131を使用する場合も同じである。高速起動時は、MMU 131のテーブルおよびCPUのレジスタのみ先行で復帰される。そして、メモリイメージが作成された後のアドレスに戻される。MMU 131のテーブルは、全てアクセス禁止に設定されているので、戻りアドレスにジャンプした時にページフォルトが発生する。データアクセス時も同じである。ページフォルトを処理するページフォルトハンドラ404は、ページフォルトが発生したアドレスからペ

ージを計算し、且つ、本発明が適用されたソフトウェアが、マークしたマークをチェックし、そのページを不揮発メモリ 104 からメインメモリ（例えば、RAM 102）へ読み込み、MMU 131 のテーブルを書き換え前の元の通りに書き戻す。

[0068] このような処理が繰り返されることで、大容量のメモリイメージであっても、所望の状態に復帰させるには最低限のメモリイメージの読み込みだけで済み、高速に起動することが可能となる。

[0069] 〔動作の詳細について〕

上記してきたように、本発明によれば、所望のソフトウェアをある状態まで通常の起動と比較して高速に起動することが可能である。その高速起動を行う手順を大きく分けると、簡便に上記したが、（Ａ）、（Ｂ）、（Ｃ）の３つに分類される。さらに（Ａ）、（Ｂ）、（Ｃ）を、フローチャートを参照した説明の前に、さらに説明を加える。（Ａ）、（Ｂ）の一連の流れは、一度実行されれば、毎回行う必要がない処理である。通常、（Ｃ）から実行されるようにすることで、高速起動することが可能となる。

[0070] （Ａ）通常起動

（Ａ－１） 起動モード切替部 110 を通常起動モードに設定し、通常の方法により、OS や所望のプログラムが起動される。

（Ａ－２） 所望のプログラムが起動された後、ソフトウェアが操作されて、ソフトウェアが所望の状態にされる。高速起動時は、この状態で起動される。

[0071] （Ｂ）状態の保存

（Ｂ－１） 何らかのキーやコマンドなどによりイメージ保存プログラム 406 が起動される。この起動に関する起動方法には、特に制限はない。

（Ｂ－２） イメージ保存プログラム 406 は、次回以降に高速に起動したい状態で、メモリイメージとレジスタを保存する。具体的には、イメージ保存プログラム 406 は、MMU 131 のテーブルを全てアクセス禁止状態に設定し、その時点でのデータ 401、プログラムコード 402、MMU テー

ブル４０３、ページフォルトハンドラ４０４、割込ベクタ４０５、およびレジスタ類を不揮発メモリ１０４に保存させる。

[0072] (C) 高速起動

(C-1) 起動モード切替部１１０が、高速起動モードに設定される。ブートローダ４０８が、起動モードを判断し、高速起動モードの場合は、イメージ復帰プログラム４０７を呼び出す。イメージ復帰プログラム４０７は、イメージ保存プログラム４０６が保存したMMUテーブル４０３、ページフォルトハンドラ４０４、割込ベクタ４０５を復帰させる。

(C-2) B-1の処理でイメージ保存プログラム４０６が起動された後のアドレスに戻る、即ちジャンプする。MMU１３１が全てアクセス禁止状態に設定されているので、プログラムコード４０２やデータ４０１にアクセスされる毎に、対応するアドレスにてページフォルトが発生し、ページフォルトハンドラ４０４が呼ばれる。

(C-3) ページフォルトハンドラ４０４は、対応する物理ページ２０７を不揮発メモリ１０４から１ページ分読み出し、MMU１３１を元に戻す。

(C-4) ページフォルトが次々に発生し、A-2の状態になるまで、必要なページフォルトが発生し続ける。

(C-5) A-2の状態になるまで物理ページ２０７が読み込まれる。この処理で読み込まれる物理ページ２０７は、実行されるソフトウェアや、その状態にも依存するが、極めて少なく、従来のハイバネーション技術で行われていたように、全ての物理ページ２０７を読み込むのと比較して起動時間は著しく短くすることが可能となる。

[0073] 図７乃至１１のフローチャートを参照し、上記(A)、(B)、(C)の各動作についてさらに説明を加える。

[0074] 図７のフローチャートは、上記(A)、(B)の処理に該当する。すなわち、主に、電源投入からイメージ保存までの処理に係わるフローチャートである。ステップS１０１において、ハードディスクレコーダ１００(図１)の電源が投入、もしくはリセットが発生してシステムが起動される。

- [0075] ステップS 1 0 2において、ブートローダ4 0 8（図6）が起動される。このステップS 1 0 2で起動されるブートローダ4 0 8は、OSや所望のソフトウェアを動作させるために最低限のハードウェアの初期化や、必要に応じてROM 1 0 3やHDD 1 0 8に格納されているソフトウェアをRAM 1 0 2に転送させたりする処理を想定し、そのような処理を実行できるブートローダであればよい。ブートローダ4 0 8は、システムに依存するものであり必須のものではない。よって、システムによっては、このステップS 1 0 2が省略される場合もある。
- [0076] ステップS 1 0 3において、起動モード切替部1 1 0の状態がチェックされ、通常起動モードか、高速起動モードかの遷移が切り替えられる（通常起動スイッチがONであるか否かが判断される）。起動モード切替部1 1 0が通常起動モードの場合、ステップS 1 0 4に処理が進められ、高速起動モードの場合、ステップS 1 6 1（図9）に進められる。
- [0077] ステップS 1 0 3において、通常起動スイッチがONであると判断されると、通常起動モードで起動するため、通常起動フラグがONに設定される。通常起動フラグがONに設定されると、ステップS 1 0 5に処理が進められ、OSが搭載されているシステムであれば、OSが起動される。一般的なシステムの場合、OSが起動される時にMMU 1 3 1が初期化され、図2のMMU 1 3 1のテーブルが作成される。本発明においては、OSの搭載は必須ではないが、仮にOSを搭載しないシステムの場合、MMU 1 3 1の初期化を行う必要がある。また、OS搭載の場合であっても、OSの種類等に制限はない。
- [0078] ステップS 1 0 6において、高速起動させたい所望のソフトウェアが起動される。ステップS 1 0 7において、起動されたソフトウェアが動作する。この処理は、上記したA-2の処理に該当する。高速起動により起動した状態と同じ状態にソフトウェアを遷移させる。例えば、所望のソフトウェアに複数のモードがあったと仮定し、その中である特定のモードで高速起動させたいのであれば、ソフトウェアを操作し、そのモードにまで遷移させる。例

例えば、ハードディスクレコーダ１００の場合、予約するモード、再生するモード、設定するモードなどがあるが、ユーザが再生するモードをよく利用する場合、再生モードにまで遷移される。

[0079] ステップＳ１０８において、イメージ保存プログラム４０６（図６）の処理が開始されたか否かが判断される。この処理は、上記したＢ－１の処理に該当する。コマンドやキー操作、スイッチ等によってイメージ保存プログラム４０６による処理が開始される。このイメージ保存プログラム４０６の実行手段に制限はない。ステップＳ１０８において、イメージ保存プログラム４０６（図６）の処理は開始されていないと判断された場合、ステップＳ１０７に処理が戻され、それ以降の処理が繰り返される。すなわちこの場合、ソフトウェアの動作が継続される。

[0080] 一方、ステップＳ１０７において、イメージ保存プログラム４０６（図６）の処理が開始されたと判断された場合、換言すれば、ソフトウェアの動作は終了したと判断された場合、ステップＳ１０９に処理が進められる。以下のステップＳ１０９乃至Ｓ１１６の処理は、上記したＢ－２の処理に該当する。また、ステップＳ１０９乃至Ｓ１１６の処理は、イメージ保存プログラム４０６が実行する処理である。

[0081] ステップＳ１０９において、図１に示したＩ／Ｏ部１０９のレジスタが保存される。基本的には、設定されている値が取得され、保存される。Ｉ／Ｏに関しては、全てのレジスタが読み込める仕様のＩ／Ｏとは限らないので、その場合は個別に対応する必要がある。なお、Ｉ／Ｏの種類や仕様は任意であり、特に本発明を適用するうえでの制限はない。

[0082] ステップＳ１１０において、ＣＰＵ１０１のレジスタが保存される。基本的にはＣＰＵ１０１の全レジスタが保存される。ＣＰＵ１０１やレジスタの種類は任意であり、特に本発明を適用するうえでの制限はない。

[0083] ステップＳ１１１において、アドレス空間の切り換えが行われる。ＣＰＵ１０１は、通常、仮想アドレスモードで動作している。このモードが、仮想アドレスモードから物理アドレスモードに遷移される。仮想アドレスモード

から物理アドレスモードへの遷移方法は、MMU 131のアーキテクチャに依存するため、本発明を適用するうえでの遷移方法に制限はない。また、仮想アドレスモードから物理アドレスモードに遷移させ場合、アドレス空間が変化するため、ステップS 111における処理では、論理アドレスと物理アドレスとで同一アドレス空間にマッピングされる必要がある。

[0084] ステップS 112において、キャッシュフラッシュが実行される。CPU 101が、TLB (Translation Look-aside Buffer)、一次キャッシュ、二次キャッシュを搭載し、それらが有効だった場合、TLBおよびキャッシュがフラッシュされる必要がある。これは、次のステップS 113において、RAM 102上に置かれたMMUテーブル403の内容を書き換える必要があり、キャッシュに格納されているデータが、全てRAM 102に反映されている必要があるからである。このステップS 112におけるキャッシュフラッシュの処理は、必要に応じて行われ、場合によっては省略されても良い処理である。

[0085] ステップS 113において、MMU 131のMMUテーブル403が、全ての物理ページ207へのアクセスを禁止する情報に書き換えられる。このステップS 113におけるMMUテーブル書き換え処理については、図8のフローチャートを参照して後述する。

[0086] ステップS 113において、MMU 131のMMUテーブル403が書き換えられると、ステップS 114に処理が進められる。ステップS 114において、キャッシュフラッシュが実行される。CPU 101がTLB、一次キャッシュ、二次キャッシュを搭載し、それらが有効だった場合、TLBおよびキャッシュがフラッシュされる必要がある。これは前段のステップS 113の処理で書き換えたMMU 131のMMUテーブル403の内容を、確実にRAM 102へ反映させるためである。このステップS 114におけるキャッシュフラッシュの処理は、必要に応じて行われ、場合によっては省略されても良い処理である。

[0087] ステップS 115において、不揮発メモリ104に対して、RAM 102

の全容量の内容が全て保存される。RAM 102のアドレスに対する不揮発メモリ 104の相対的なアドレス位置が一致している必要がある。例えば、RAM 102の物理アドレスが 0x10000000 から 0xffffffff にマッピングされていたとする。この場合、例えば不揮発メモリ 104に対しては、0x40000000 から 0x4fffffffff のアドレスにより、データが読み込める必要がある。

[0088] この例の場合、RAM 102のアドレスに対する不揮発メモリ 104のオフセットは 0x30000000 となるため、RAM 102に対するアドレスであっても、0x30000000 のオフセットを加えるだけで不揮発メモリ 104内のアドレスに変換することが可能となる。不揮発メモリ 104は、必ずしもメモリマップ上にマッピングされる必要はない。上記オフセットを加えたアドレスをキーにして読み込みができれば良い。また、不揮発メモリ 104に対する保存方法は、アーキテクチャに依存するが、本発明の適用するうえで、その保存方法に制限はない。

[0089] ステップS 116で、イメージ保存プログラム 406での処理が終了される。イメージ保存プログラム 406での処理が終了されることで、電源がOFFまたはリセット(RESET)することができる状態となる。

[0090] 図7に示したフローチャートにおいて、ステップS 103において、通常起動スイッチがONではないと判断されたときの処理と、ステップS 113でのMMUテーブル書き換え処理についての詳細な説明が残っているが、まずここでは、図8のフローチャートを参照し、ステップS 113でのMMUテーブル書き換え処理についての詳細な説明を行う。

[0091] 図8に示したフローチャートに基づく処理は、MMU 131のMMUテーブル 403が、図2で示した構成になっており、このMMUテーブル 403を書き換える処理である。

[0092] ステップS 131において、MMU 131のMMUテーブル 403の書き換えが開始されると、まず、変数レベル1ディスクリプタポインタに、レベル1ディスクリプタテーブル 201の先頭アドレスが代入される。ステップS

132において、変数レベル1ディスクリプタポインタが指すアドレスからレベル1ディスクリプタ203が取得される。

[0093] ステップS133において、ステップS132の処理で取得されたレベル1ディスクリプタ203内にレベル2ディスクリプタテーブル204へのポインタが存在するか否かが判断される。ステップS133において、レベル1ディスクリプタ203内にレベル2ディスクリプタテーブル204へのポインタが存在すると判断された場合、ステップS136に処理が進められ、レベル1ディスクリプタ203内にレベル2ディスクリプタテーブル204へのポインタが存在しないと判断された場合、ステップS134に処理が進められる。

[0094] ステップS134において、変数レベル1ディスクリプタポインタが次のレベル1ディスクリプタポインタのアドレスに移動される。そして、ステップS135に処理が進められ、レベル1ディスクリプタポインタが最終に到達したか否かが判断される。

[0095] ステップS135において、レベル1ディスクリプタポインタが最終に到達したと判断されるまで、ステップS134に処理が戻され、変数レベル1ディスクリプタポインタが次のレベル1ディスクリプタポインタのアドレスに移動されるといった処理が繰り返される。そして、ステップS135において、レベル1ディスクリプタポインタが最終に到達したと判断されると、処理はステップS114(図7)に進められる。すなわち、MMUテーブルの書き換えが終了したと判断され、図7に示したフローチャートの処理に、処理が戻される。

[0096] 一方、ステップS133において、レベル1ディスクリプタ203内にレベル2ディスクリプタテーブル204へのポインタが存在すると判断された場合、ステップS136に処理が進められる。ステップS136において、変数レベル2ディスクリプタポインタに、レベル2ディスクリプタテーブル204の先頭アドレスが代入される。

[0097] ステップ137において、変数レベル2ディスクリプタポインタが指すア

ドレスからレベル2ディスクリプタ206が取得される。ステップS138において、ステップS137の処理で取得されたレベル2ディスクリプタ206内に物理ページ207が存在するか否かが判断される。ステップS138において、取得されたレベル2ディスクリプタ206内に物理ページ207が存在すると判断された場合、ステップS139に処理が進められ、取得されたレベル2ディスクリプタ206内に物理ページ207は存在しないと判断された場合、ステップS143に処理が進められる。

[0098] ステップS139において、ステップS137の処理で取得されたレベル2ディスクリプタ206内の物理ページ207が、ステップS115(図7)にて保存の対象となるRAM102内のアドレスの範囲内であるか否かが判断される。ステップS139において、レベル2ディスクリプタ206内の物理ページ207が、保存の対象となるRAM102内のアドレスの範囲内であると判断された場合、ステップS140に処理が進められ、保存の対象となるRAM102内のアドレスの範囲内ではないと判断された場合、ステップS143に処理が進められる。

[0099] ステップS140において、ステップS137で取得されたレベル2ディスクリプタ206のアクセス許可bit(図3のアクセス許可bit305)がチェックされ、その物理ページ207がアクセス許可にされているか否かが判断される。ステップS140において、物理ページ207へのアクセスが許可されていると判断された場合、ステップS141に処理が進められ、物理ページ207へのアクセスは許可されていないと判断された場合、ステップS143に処理が進められる。

[0100] ステップS141において、ステップS137の処理で取得されたレベル2ディスクリプタ206のアクセス許可bit305がアクセス禁止を表すbitに書き換えられる。そして、ステップS142において、書き換えられたレベル2ディスクリプタ206に対してマーキングが実行される。この処理は、ステップS137の処理で取得されたレベル2ディスクリプタ206のアクセス許可bit305が、本発明が適用されたソフトウェアにより書き換えら

れたのか、他のソフトウェア、例えば、本来のOS等の動作によって書き換えられたのかを識別するための情報を保存する（マーキングする）ために行われる。

- [0101] ステップS 1 4 2におけるマーキングの仕方については、アーキテクチャに依存し、本発明を適用するうえでの制限はない。例えば、レベル2 ディスクリプタ 2 0 6に使われていない空きbitが存在するのであれば、その空きbitをマーキングの情報を埋め込むbitとして使用することができる。また、別途テーブルを持たせてマーキングされたところとされていないところとが管理されるようにしても良い。いずれにせよ、仮にOS等が搭載されたシステムで、かつ、OSがこれらのbitを使用していた場合は、共存するような仕組みにすることで、本発明を実施することが可能である。
- [0102] ステップS 1 4 3において、変数レベル2 ディスクリプタポインタが、次のレベル2 ディスクリプタ 2 0 6のポインタのアドレスに移動される。このステップS 1 4 3への処理には、ステップS 1 3 8において、レベル2 ディスクリプタ 2 0 6内に物理ページ 2 0 7が存在しないと判断された場合、ステップS 1 3 9において、レベル2 ディスクリプタ 2 0 6がRAM 1 0 2を指していないと判断された場合、または、ステップS 1 4 0において、物理ページ 2 0 7へのアクセスは許可されていないと判断された場合にも来る。
- [0103] ステップS 1 4 4において、レベル2 ディスクリプタポインタが最終に到達したか否かが判断される。ステップS 1 4 4において、レベル2 ディスクリプタポインタが最終に到達したと判断されるまで、ステップS 1 3 7に処理が戻され、それ以降の処理が繰り返される。一方、ステップS 1 4 4において、レベル2 ディスクリプタポインタが最終に到達したと判断された場合、ステップS 1 3 4に処理が進められる。ステップS 1 3 4以降の処理については既に説明したので、その説明は省略する。
- [0104] このようにして、MMU 1 3 1のMMUテーブル 4 0 3が書き換えられる。
- [0105] 次に、高速起動時の処理について説明する。高速起動は、ステップS 1 0

3において、通常起動スイッチがONになっていないと判断されたとき、すなわち、高速起動にスイッチが切り替えられていると判断されたときに実行される。図9のフローチャートは、ステップS103において、通常起動スイッチがONにはなっていないと判断されたときに処理が進められるフローチャートであり、高速起動時の処理について説明するためのフローチャートである。

- [0106] ステップS161において、高速起動モードで起動するため、通常起動フラグがOFF（高速起動フラグがON）に設定される。ステップS162において、必要に応じて、割込ベクタ405、ページフォルトハンドラ404、MMUテーブル403が、イメージが保存された時と同じRAM102のアドレスに読み込まれる。
- [0107] ステップS163において、MMU131のMMUテーブルが読み込まれる。このステップS163において実行されるMMUテーブル読み込み処理については、図10のフローチャートを参照し、後述する。
- [0108] MMUテーブルの読み込みが終わると、ステップS164に処理が進められる。ステップS164において、CPU101のアドレス空間が、物理アドレスモードから仮想アドレスモードに遷移される。物理アドレスモードから仮想アドレスモードに遷移させる場合、アドレス空間が変化するため、ステップS164における処理では、論理アドレスと物理アドレスが同一アドレス空間にマッピングされる。
- [0109] ステップS165において、ステップS110（図7）で保存されたCPU101のレジスタの値が不揮発メモリ104から読み出され、CPU101に対して復帰される。CPU101やレジスタの種類は任意であり、本発明を適用する上での制限はない。
- [0110] ステップS166において、ステップS109（図7）で保存されたI/Oのレジスタの値が、不揮発メモリ104から読み出され、I/O部109に対して復帰される。I/Oの種類や仕様は任意であり、本発明を適用する上での制限はない。

- [0111] このように、レジスタなどが復帰されると、ステップS 107 (図7)に処理が進められる。ステップS 107において、ソフトウェアが動作する。この場合、ステップS 104乃至S 106の処理が実行されずに、ステップS 107においてソフトウェアが動作開始となる。よって、ステップS 104乃至S 106の処理が実行される分だけ、少なくともソフトウェアが動作開始できるまでにかかる時間が短縮できることになる。特に、ステップS 105におけるOSの起動やMMUの初期化にかかる時間、およびステップS 106におけるソフトウェアの起動にかかる時間をなくすることができることで、大幅な時間の短縮を期待することができる。
- [0112] 図9のフローチャートの説明に戻り、ステップS 163で実行されるMMUテーブル読み込み処理の詳細について、図10のフローチャートを参照して説明する。
- [0113] ステップS 181において、MMUテーブル403の読み出しが開始されると、まずレベル1ディスクリプタテーブル201が読み出される。このレベル1ディスクリプタテーブル201は、ステップS 115 (図7)の処理で、不揮発メモリ104にRAM 102の内容が保存されたが、その不揮発メモリ104に保存されている内容から、レベル1ディスクリプタテーブル201のみが読み出される。
- [0114] ステップS 182において、変数レベル1ディスクリプタポインタに、レベル1ディスクリプタテーブル201の先頭アドレスが代入される。ステップS 183において、変数レベル1ディスクリプタポインタが指すアドレスからレベル1ディスクリプタ203が取得される。ステップS 184において、ステップS 183の処理で取得されたレベル1ディスクリプタ203内にレベル2ディスクリプタテーブル204へのポインタが存在するか否かが判断される。
- [0115] ステップ184において、取得されたレベル1ディスクリプタ203内にレベル2ディスクリプタテーブル204へのポインタが存在すると判断された場合、ステップS 187に処理が進められ、取得されたレベル1ディスクリプ

タ 203 内にレベル 2 ディスクリプタテーブル 204 へのポインタは存在しないと判断された場合、ステップ S 185 に処理が進められる。

[0116] ステップ S 185 において、変数レベル 1 ディスクリプタポインタが、次のレベル 1 ディスクリプタポインタのアドレスに移動される。そして、ステップ S 186 において、レベル 1 ディスクリプタポインタが、最終に到達したか否かが判断される。ステップ S 186 において、レベル 1 ディスクリプタポインタが、最終に到達したと判断された場合、ステップ S 164 (図 9) に処理が進められる。すなわちこの場合、MMU テーブル 403 の読み込みが完了されたため、次の処理へ処理が進められる。

[0117] 一方、ステップ S 186 において、レベル 1 ディスクリプタポインタ 203 は、最終に到達していないと判断された場合、ステップ S 183 に処理が戻され、それ以降の処理が繰り返される。ステップ S 183 乃至 S 186 が繰り返され、ステップ S 184 において、取得されたレベル 1 ディスクリプタ 203 内にレベル 2 ディスクリプタテーブル 204 へのポインタが存在すると判断されると、ステップ S 187 に処理が進められる。

[0118] ステップ S 187 において、ステップ S 183 で取得されたレベル 1 ディスクリプタ 203 内にある、レベル 2 ディスクリプタテーブル 204 へのポインタが、RAM 102 を指しているか否かが判断される。ステップ S 187 において、レベル 2 ディスクリプタテーブル 204 へのポインタが、RAM 102 を指していると判断された場合、ステップ S 188 に処理が進められ、レベル 2 ディスクリプタテーブル 204 へのポインタは、RAM 102 を指していないと判断された場合、ステップ S 185 に処理が進められ、それ以降の処理が繰り返される。

[0119] ステップ S 188 において、レベル 2 ディスクリプタテーブルが読み出される。このレベル 2 ディスクリプタテーブル 204 は、ステップ S 115 (図 7) の処理で、不揮発メモリ 104 に RAM 102 の内容として保存され、その不揮発メモリ 104 に保存されている内容から、レベル 2 ディスクリプタテーブル 204 のみが読み出される。その後、処理は、ステップ S 185 に

進められ、それ以降の処理が繰り返される。

[0120] このようにして、MMUテーブルの読み出しが行われる。

[0121] 次に、図 11 のフローチャートを参照し、ページフォルトが発生したときに実行される処理について説明を加える。ステップ S 201 において、ページフォルトが発生すると、割込ベクタ 405 にジャンプされる。すなわち、割込ベクタ 405 から、実際のページフォルト処理を行う割込ハンドラへのジャンプが実行される。

[0122] 一般的な CPU 101 は、ページフォルトが発生した場合、割込処理として特定の割込ベクタにジャンプし、その割込ハンドラとして処理する。図 11 に示したページフォルト発生時の処理に係わるフローチャートは、ページフォルトの割込を処理する割込ハンドラを想定しているが、これらは CPU 101 のアーキテクチャに依存する。なお、本発明の適用は、CPU 101 のメーカーや型番により、制限が加えられることはない。

[0123] ステップ S 202 において、通常起動フラグが ON であるか否かが判断される。通常起動フラグは、例えば、ステップ S 104 (図 7) の処理で ON に設定される。ステップ S 202 において、通常起動フラグが ON であると判断された場合、換言すれば、通常起動であると判断された場合、ステップ S 207 に処理が進められる。一方、ステップ S 202 において、通常起動フラグが ON ではないと判断された場合、換言すれば、高速起動であると判断された場合、ステップ S 203 に処理が進められる。

[0124] ステップ S 203 において、対象となる物理ページ 207、即ちページフォルトが発生したアドレスに対応する物理ページ 207 が、マーキングされた物理ページ 207 であるか否かが判断される。マーキングは、ステップ S 142 (図 8) の処理で実行されたマーキングである。すなわち、マーキングされている物理ページ 207 は、本発明を適用したソフトウェアにより、アクセス禁止に書き換えられた物理ページ 207 である。

[0125] なお、一般的に物理ページ 207 とアドレスには、以下の計算式が満たされるが、これらは CPU 101 のアーキテクチャに依存するので、特にこの

計算式に本発明の適用範囲が限定されるものではない。

物理ページ = アドレス／ページサイズ（例えば、上記した例では4KB）

この式が示すように物理ページは、アドレスをページサイズで除算したものとなる。

[0126] ステップS203において、ページフォルトが発生したアドレスに対応する物理ページ207は、マーキングされた物理ページ207であると判断された場合、ステップS204に処理が進められ、マーキングされていない物理ページ207であると判断された場合、ステップS207に処理が進められる。

[0127] ステップS204において、対象となる物理ページ207、即ちページフォルトが発生したアドレスに対応する物理ページ207が、ステップS115（図7）の処理で不揮発メモリ104に保存されたイメージから4KB分のみ読み出される。

[0128] ステップS205において、対象となる物理ページ207、即ちページフォルトが発生したアドレスに対応する物理ページ207の、レベル2ディスクリプタ206のアクセス許可bit305がアクセス許可に書き換えられる。ステップS206において、ステップS142（図8）の処理でマーキングした識別情報が解除される。

[0129] このようにして、ページフォルトが発生されたときの処理が実行されることで、高速起動が実現される。

[0130] 一方、ページフォルトが発生したが、ステップS202において、通常起動フラグがONであると判断された場合、または、ステップS203において、対象ページはマーキングされていないと判断された場合、ステップS207に処理が進められる。ステップS207において、標準のページフォルトの処理が実行される。すなわち、通常起動のときや、本発明が適用されたソフトウェア以外のソフトウェア（OSなど）が、アクセスを不許可に設定していたような場合、通常起動や、アクセス不許可時の処理が実行される。

[0131] 本発明を適用したシステムに、OS等が搭載されていた場合、標準的なペ

ージフォルトハンドラが通常実装されている。それに対して、上記したようなページフォルト機能がシステムに実装された場合、OS本来が行うページフォルト処理が、改めて実行される必要があるときもある。このステップS207の処理は、OS等のシステムに依存するものであり必須の処理ではないため、本実施の形態として省略することも可能である。

[0132] このようにして、ページフォルトが発生したときの処理が実行されることで、高速起動が実現される。

[0133] [効果]

上記したように、Memory Management Unit(MMU)、もしくは、MMU相当のメモリ管理機能を搭載したコンピュータシステム上で、MMUのテーブルに対し、ソフトウェアの動作に必要なRAMの最小単位、いわゆるページに対して、全てのページにてページフォルトが発生するようにページテーブルエントリを書き換え、起動時は、アクセスするRAMに対して発生したページフォルトに対し、本来OS等が有する既知の技術であるページイン・ページアウトとしての機能だけでなく、発生したページフォルトの機能を保存したメモリーイメージのページ単位の読み込みに用いることで、以下のような効果がある。

[0134] まず、必要最小限のメモリーイメージの読み込み容量を実現することが可能となる。このことにより、例えば、パーソナルコンピュータの起動時間を短縮することが可能となる。具体的には、従来、数十秒から数分を要していた起動時間を、数秒以内で起動させることができるようになる。

[0135] また、デジタル家電の起動時間を短縮することが可能となる。テレビジョン受像器やハードディスクレコーダなどのデジタル家電（電子機器）には、OS（所定のソフトウェア）を搭載した機種がある。所定のソフトウェアを搭載した機器の場合、起動時間が長くなることがあるが、本発明を適用することで、これらのデジタル家電における起動時間を短縮できるようになる。

[0136] また、バッテリーの寿命を延命化することが可能となる。従来の高速起動を実現させる方法として、CPUやメモリを省電力モードに移行させる方法が

あった。この方法では、省電力モードとは言え、電力は必要であり、バッテリーで動く機器にとってはその消費電力は無視できない。本発明を適用することで、不揮発メモリに起動イメージを保存することが可能となり、電源をRAMに供給したまま停止させる、いわゆるサスペンド（従来の省電力モードに対応）を使う必要がなくなる。よって、結果的に、バッテリー寿命を著しく伸ばすことができる。

[0137] さらに、家電製品の省エネルギー化を実現することが可能となる。テレビジョン受像器やハードディスクレコーダなどは一般的に起動が遅く、そのために「高速起動モード」なるモードを持つ機種も存在している。しかしながら、この「高速起動モード」は、家電製品を高速に起動させるため、常に電源を投入していることで、高速起動を実現している。そのため電力は電源投入時と同等に消費されている。

[0138] しかしながら、本発明を適用することで、起動時間を短くすることができるため、「高速起動モード」のときにかかる起動時間と同等、またはそれ以上に短い起動時間で起動させることができるようになり、「高速起動モード」を設ける必要がなくなる。したがって、「高速起動モード」のときに、常に電源を投入してなくてはならないといった状態をなくすることが可能となるため、結果として省エネルギー化を実現することが可能となる。

[0139] 上記したコンピュータが実行するプログラムは、本明細書で説明する順序に沿って時系列に処理が行われるプログラムであっても良いし、並列に、あるいは呼び出しが行われたとき等の必要なタイミングで処理が行われるプログラムであっても良い。また、専用のハードウェアで構成することもできる。また、本明細書において、システムとは、複数の装置により構成される装置全体を表すものである。

[0140] なお、本発明の実施の形態は、上述した実施の形態に限定されるものではなく、本発明の要旨を逸脱しない範囲において種々の変更が可能である。

符号の説明

[0141] 100 ハードディスクレコーダ, 101 CPU, 102 RAM

, 103 ROM, 104 不揮発メモリ, 105 MPEGエンコード・デコード部, 106 チューナ, 107 HDDインターフェース, 108 HDD, 109 I/O部, 110 起動モード切替部

請求の範囲

- [請求項1] メモリを管理する機能を有する制御装置に、
- 所定のソフトウェアの動作に必要なページに対して、全てのページでページフォルトが発生するようにページテーブルエントリを書き換え、
- 前記ソフトウェアが起動時に、前記ページテーブルエントリで、ページフォルトが発生し、そのページフォルトが発生したページを順次読み出す
- ステップを含むコンピュータが読み取り可能なプログラム。
- [請求項2] 前記所定のソフトウェアが起動された後、前記ページテーブルエントリを書き換え、その起動時のデータ、プログラムコード、テーブル、ページフォルトハンドラ、割り込みベクタ、およびレジスタを、前記メモリに記憶させる
- 請求項1に記載のプログラム。
- [請求項3] 前記メモリのうち、書き換えの対象となる前記ページテーブルエントリを記憶しているのはRAMであり、順次読み出される前記ページを記憶しているのは不揮発メモリである 請求項1に記載のプログラム。
- [請求項4] 組み込み型のコンピュータが読み込む
- 請求項1に記載のプログラム。
- [請求項5] メモリを管理する機能を有する制御装置の制御方法において、
- 所定のソフトウェアの動作に必要なページに対して、全てのページでページフォルトが発生するようにページテーブルエントリを書き換え、
- 前記ソフトウェアが起動時に、前記ページテーブルエントリで、ページフォルトが発生し、そのページフォルトが発生したページを順次読み出す
- ステップを含む制御方法。

[請求項6]

メモリを管理する機能を有する制御装置において、

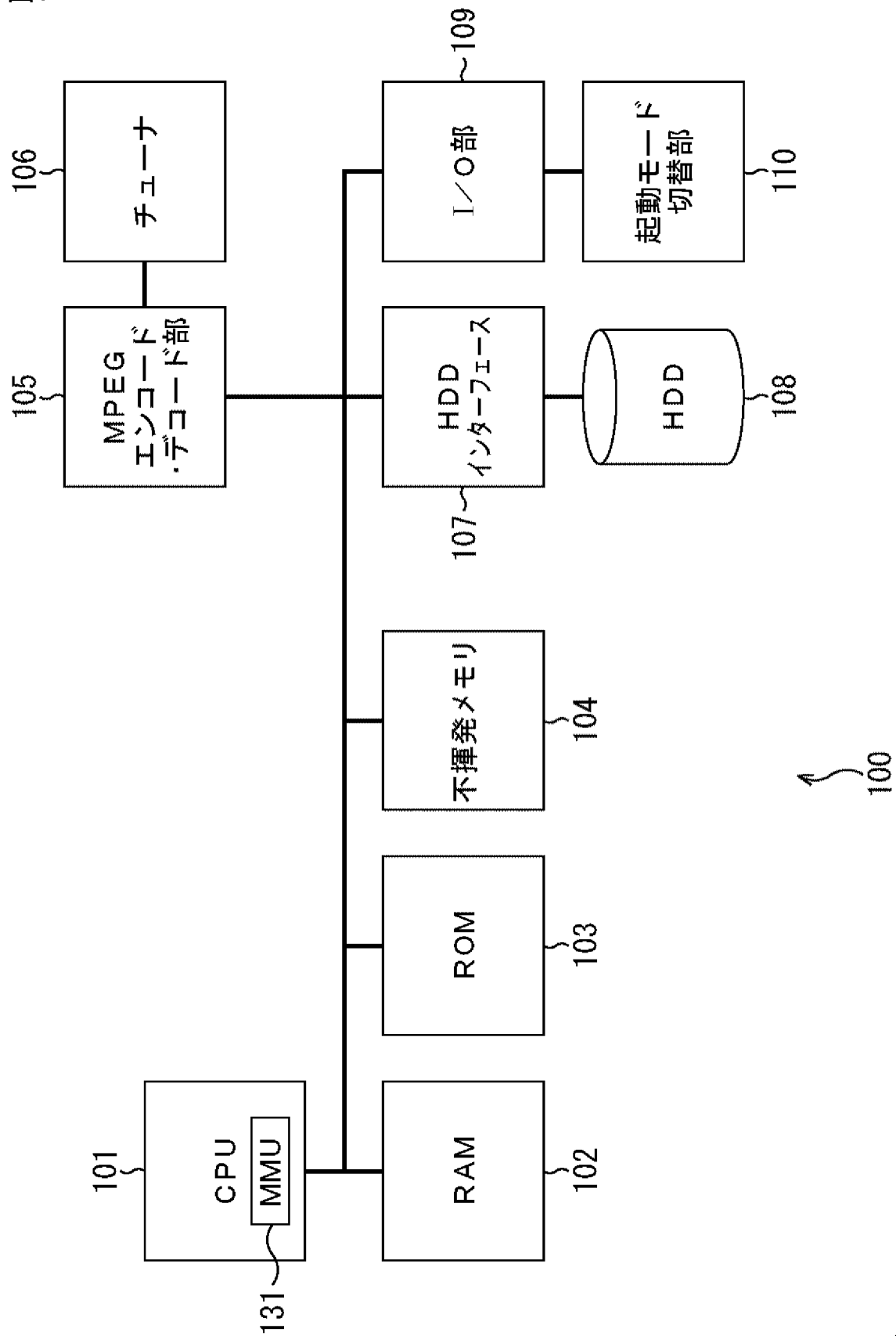
所定のソフトウェアの動作に必要なページに対して、全てのページでページフォルトが発生するようにページテーブルエントリを書き換える書き換え手段と、

前記ソフトウェアが起動時に、前記ページテーブルエントリで、ページフォルトが発生し、そのページフォルトが発生したページを順次読み出す読み出し手段と

を備える制御装置。

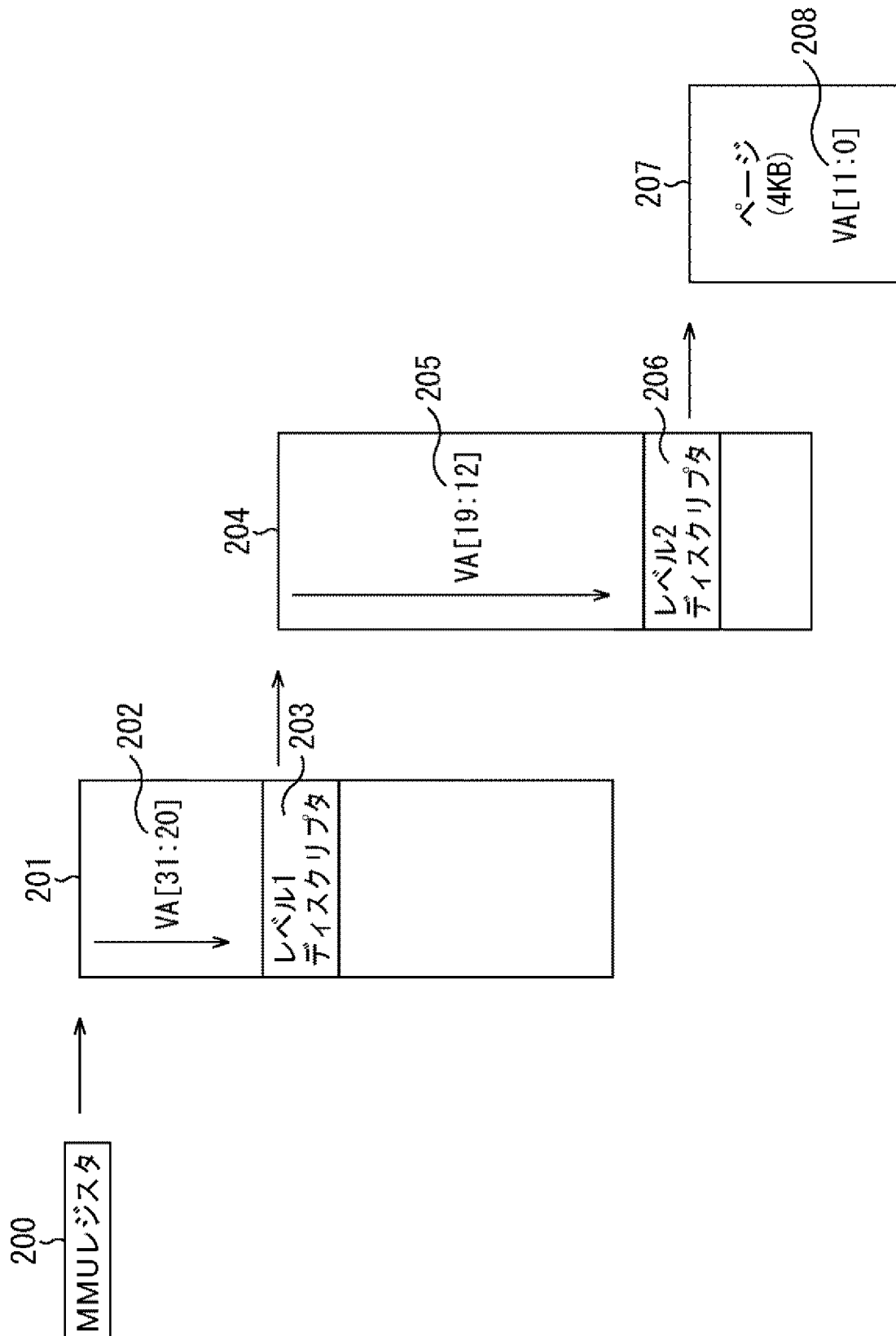
[図1]

図1



[図2]

図2



[図3]

図3



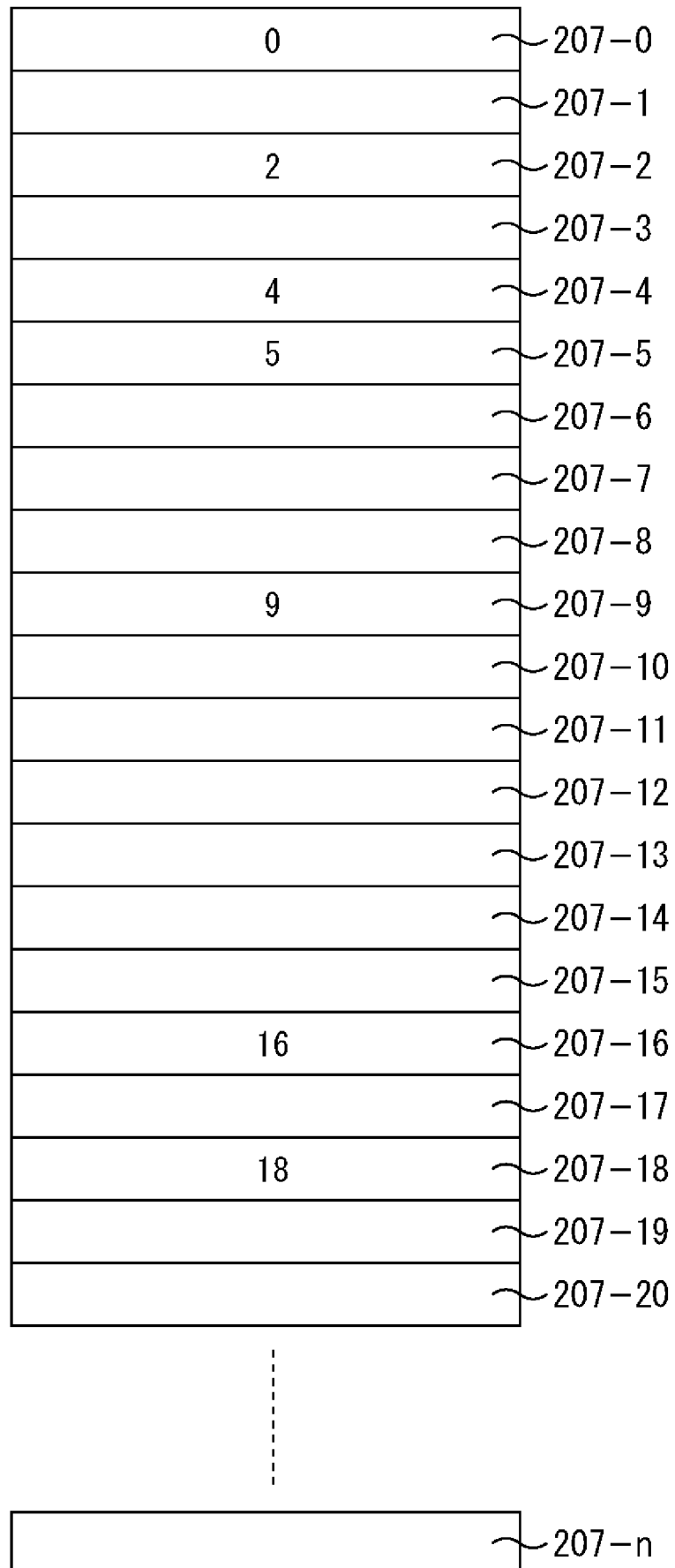
[図4]

図4

0	〜 207-0
1	〜 207-1
2	〜 207-2
3	〜 207-3
4	〜 207-4
5	〜 207-5
6	〜 207-6
7	〜 207-7
8	〜 207-8
9	〜 207-9
10	〜 207-10
11	〜 207-11
12	〜 207-12
13	〜 207-13
14	〜 207-14
15	〜 207-15
16	〜 207-16
17	〜 207-17
18	〜 207-18
19	〜 207-19
20	〜 207-20
⋮	
n	〜 207-n

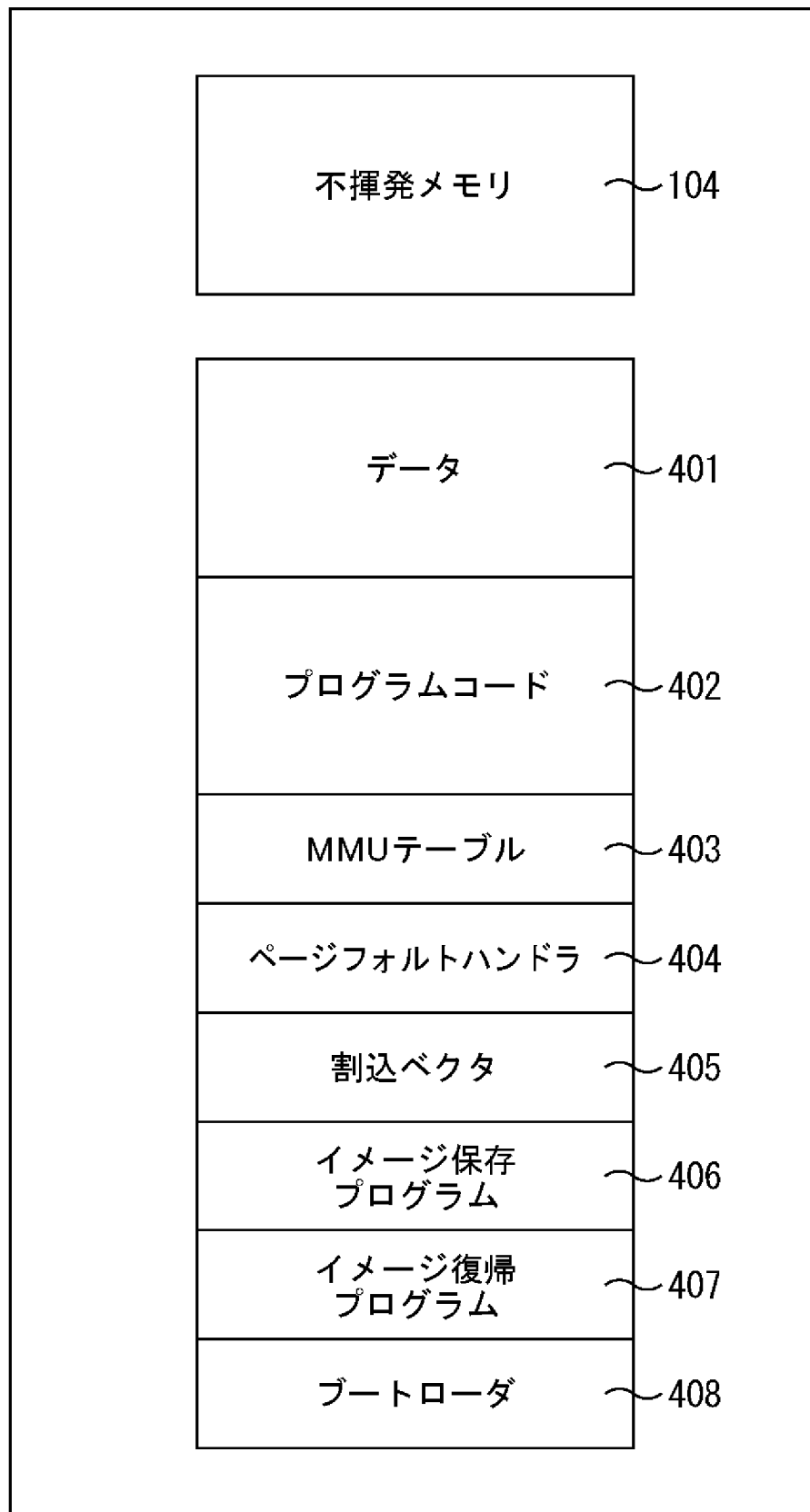
[図5]

図5

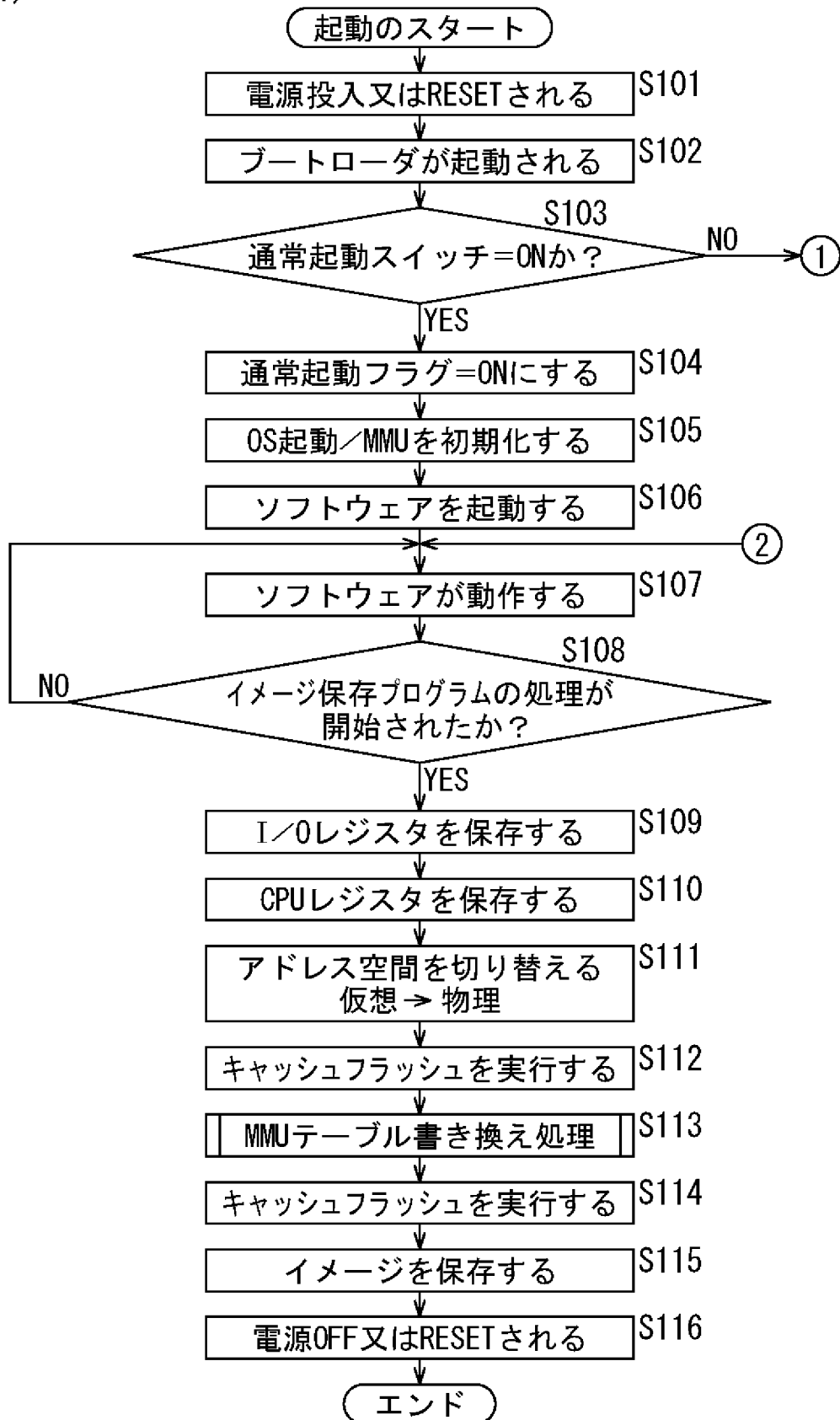


[図6]

図6



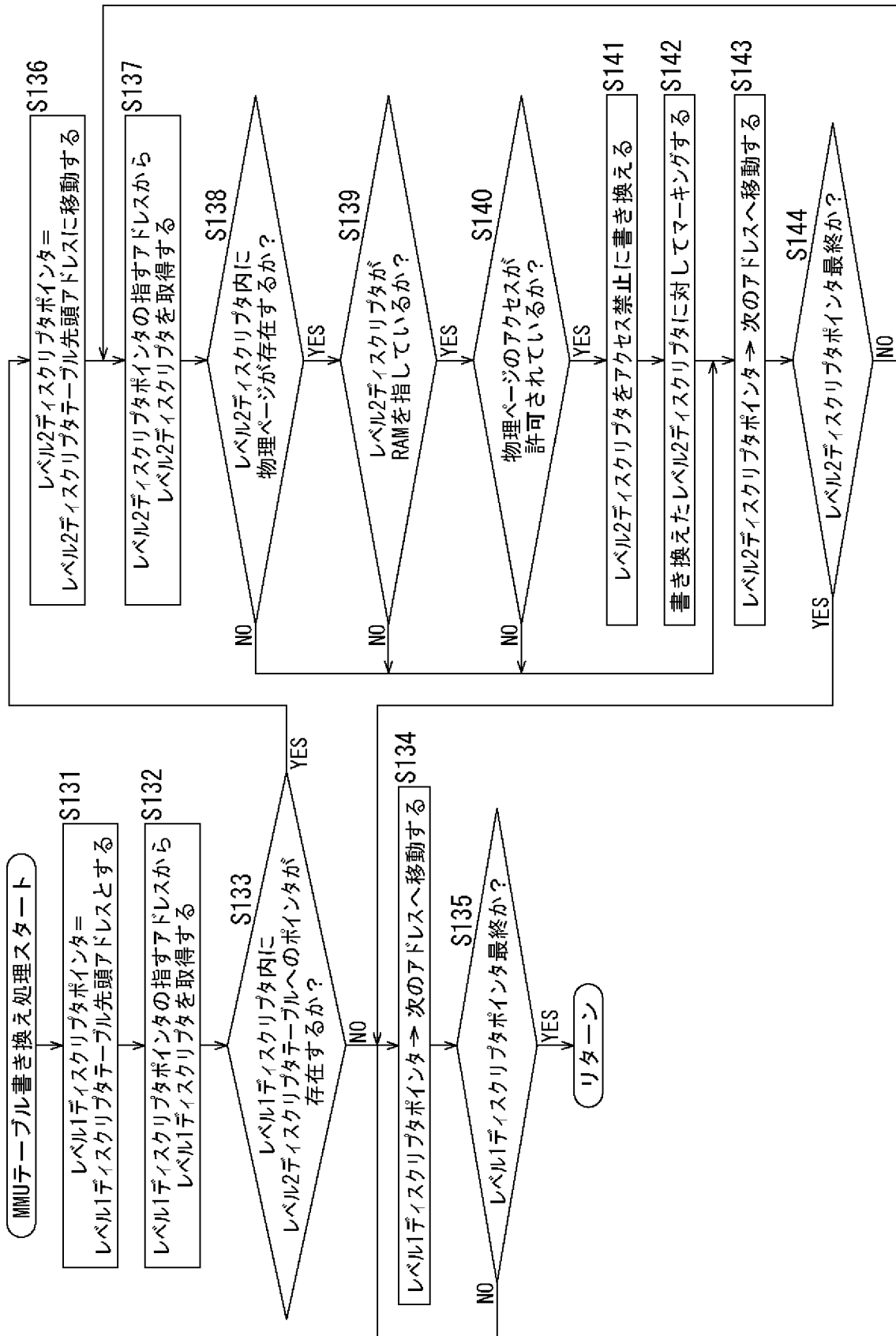
[図7]
図7
(7-1)



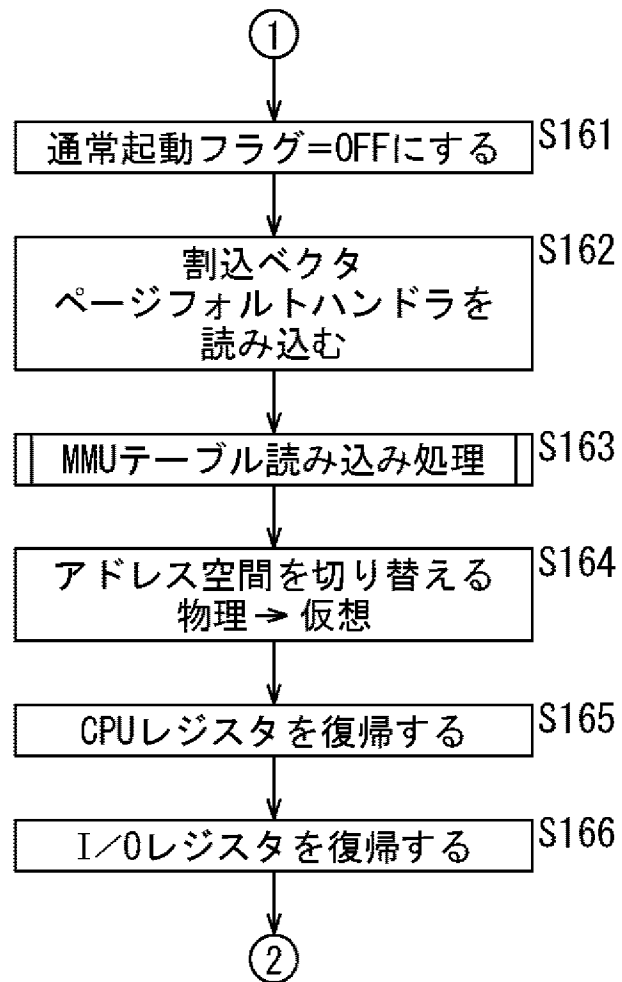
[図8]

図8

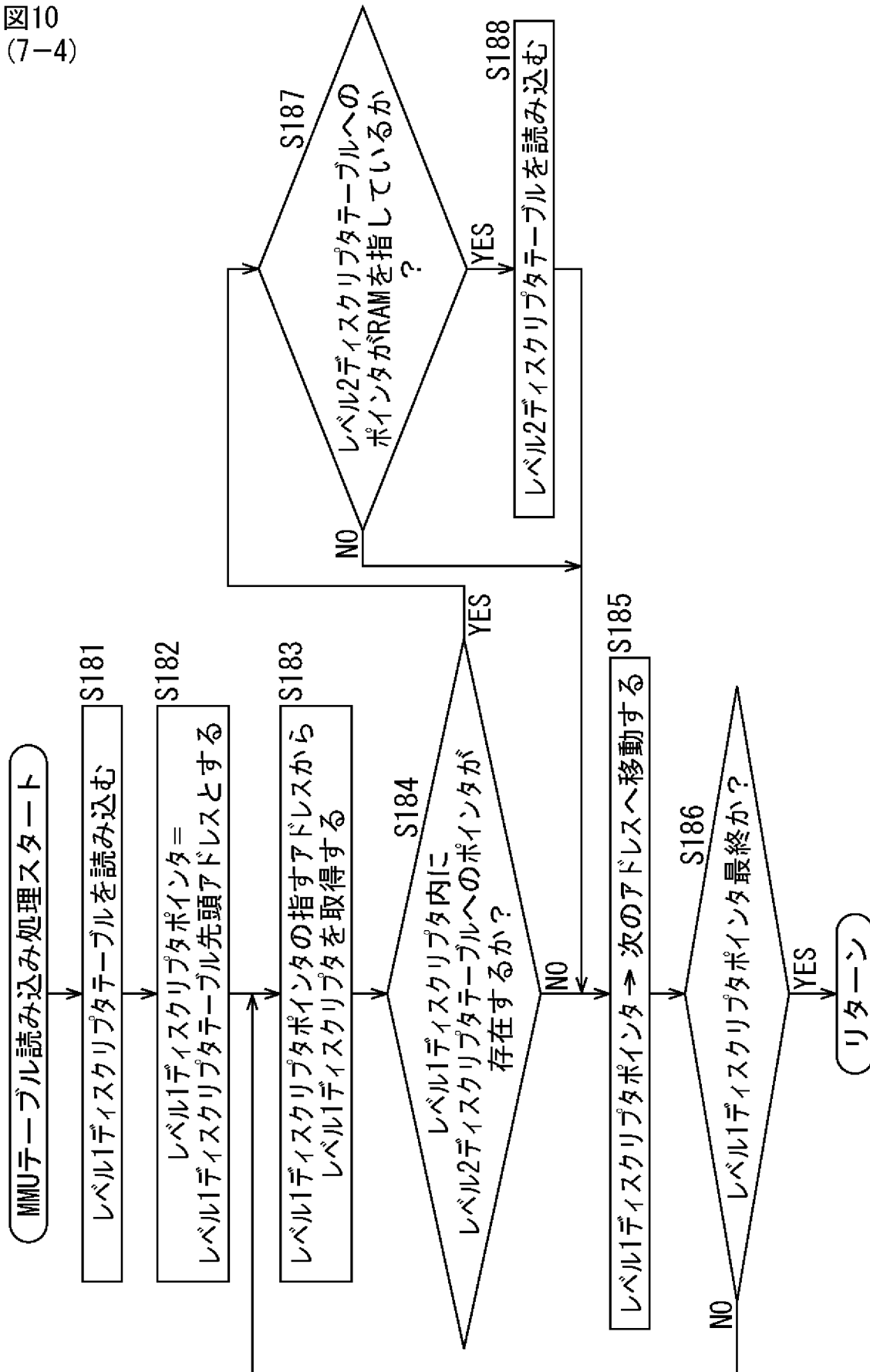
(7-2)



[図9]
図9
(7-3)

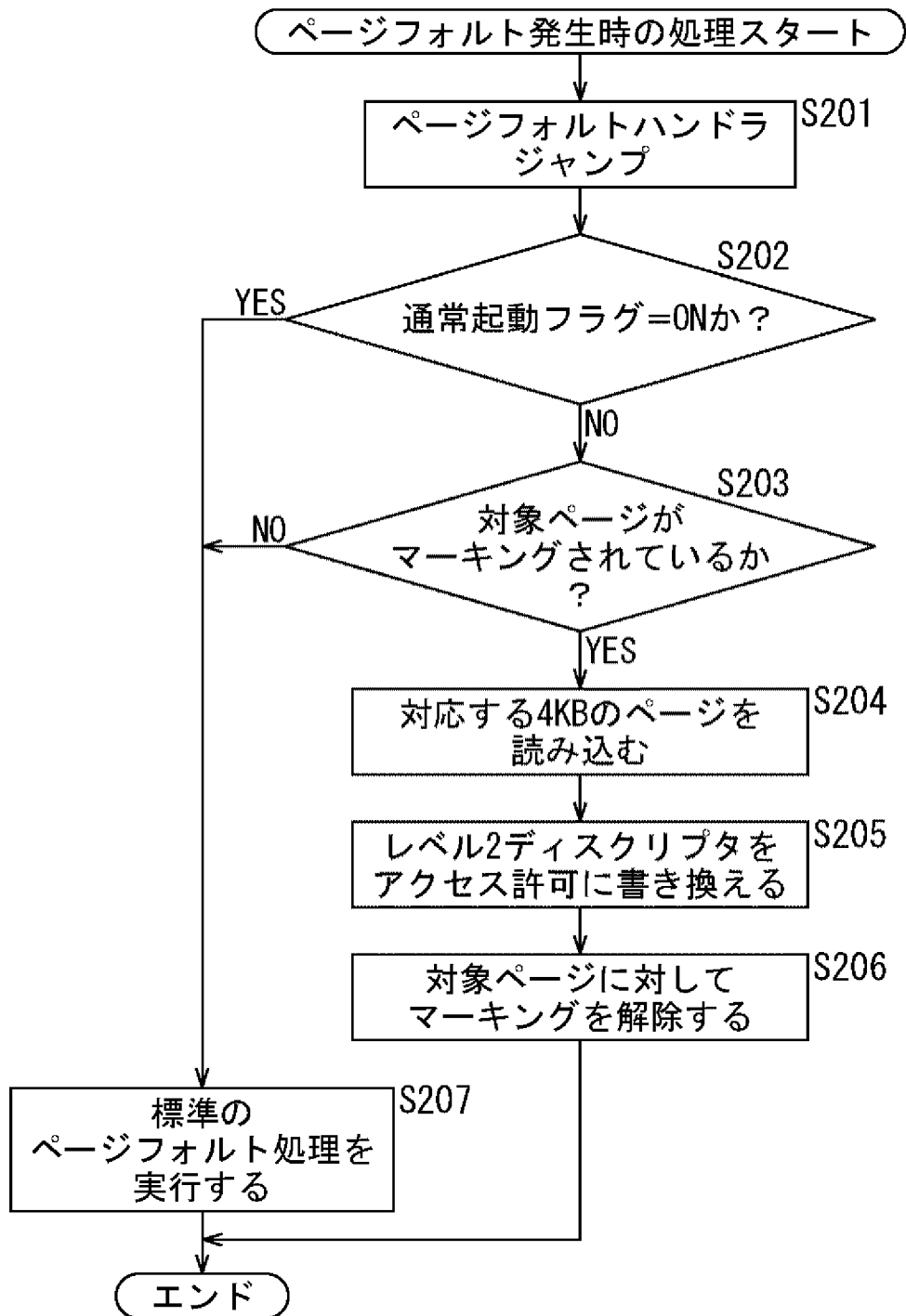


[図10]
図10
(7-4)



[図11]

図11



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2010/053627

A. CLASSIFICATION OF SUBJECT MATTER

G06F9/445(2006.01) i, G06F9/54(2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F9/445, G06F9/54, G06F12/06, G06F12/10

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho 1922-1996 Jitsuyo Shinan Toroku Koho 1996-2010

Kokai Jitsuyo Shinan Koho 1971-2010 Toroku Jitsuyo Shinan Koho 1994-2010

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JP 2007-334383 A (Sony Corp.), 27 December 2007 (27.12.2007), paragraphs [0016] to [0018], [0089] to [0101], [0119] to [0121], [0131] to [0133] & US 2008/0005541 A1	1-6
A	JP 2006-202252 A (Canon Inc.), 03 August 2006 (03.08.2006), paragraphs [0040] to [0042], [0049] to [0054] & US 2006/0173925 A1	1-6
A	JP 2005-149225 A (Sony Corp.), 09 June 2005 (09.06.2005), paragraphs [0041] to [0052] (Family: none)	1-6

☐ Further documents are listed in the continuation of Box C.☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search
10 May, 2010 (10.05.10)Date of mailing of the international search report
18 May, 2010 (18.05.10)Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

A. 発明の属する分野の分類 (国際特許分類 (I P C))

Int.Cl. G06F9/445(2006.01)i, G06F9/54(2006.01)i

B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (I P C))

Int.Cl. G06F9/445, G06F9/54, G06F12/06, G06F12/10

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	1 9 2 2 - 1 9 9 6 年
日本国公開実用新案公報	1 9 7 1 - 2 0 1 0 年
日本国実用新案登録公報	1 9 9 6 - 2 0 1 0 年
日本国登録実用新案公報	1 9 9 4 - 2 0 1 0 年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
X	JP 2007-334383 A (ソニー株式会社) 2007. 12. 27, 段落 [0 0 1 6] - [0 0 1 8], [0 0 8 9] - [0 1 0 1], [0 1 1 9] - [0 1 2 1], [0 1 3 1] - [0 1 3 3] & US 2008/0005541 A1	1-6
A	JP 2006-202252 A (キヤノン株式会社) 2006. 08. 03, 段落 [0 0 4 0] - [0 0 4 2], [0 0 4 9] - [0 0 5 4] & US 2006/0173925 A1	1-6

☒ C 欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

「A」特に関連のある文献ではなく、一般的技術水準を示すもの
「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの
「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)
「O」口頭による開示、使用、展示等に言及する文献
「P」国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&」同一パテントファミリー文献

国際調査を完了した日

1 0 . 0 5 . 2 0 1 0

国際調査報告の発送日

1 8 . 0 5 . 2 0 1 0

国際調査機関の名称及びあて先

日本国特許庁 (I S A / J P)

郵便番号 1 0 0 - 8 9 1 5

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

稲垣 良一

5 B

4 0 5 5

電話番号 0 3 - 3 5 8 1 - 1 1 0 1 内線 3 5 4 4

C (続き) . 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
A	JP 2005-149225 A (ソニー株式会社) 2005.06.09, 段落 [0041] - [0052] (ファミリーなし)	1-6