(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3617554号
(P3617554)
(45)発行日　平成17年2月9日(2005.2.9)　(24)登録日　平成16年11月19日(2004.11.19)

(51)Int.Cl.[7]　F I
G06T 1/20　G06T 1/20 A

請求項の数 4　(全 11 頁)

(21)出願番号　特願平7-142938
(22)出願日　平成7年6月9日(1995.6.9)
(65)公開番号　特開平8-339433
(43)公開日　平成8年12月24日(1996.12.24)
審査請求日　平成14年6月10日(2002.6.10)

(73)特許権者　000132471
株式会社セガ
東京都大田区羽田１丁目２番１２号
(74)代理人　100079108
弁理士　稲葉　良幸
(74)代理人　100080953
弁理士　田中　克郎
(74)代理人　100093861
弁理士　大賀　眞司
(72)発明者　竹下　英敏
東京都大田区羽田１丁目２番１２号　株式会社セガ・エンタープライゼス内

審査官　脇岡　剛

最終頁に続く

(54)【発明の名称】データ処理方法及び装置、並びに画像処理装置

(57)【特許請求の範囲】

【請求項１】

プログラム記憶手段に格納されたプログラムに基づき、前記プログラム記憶手段に格納された圧縮データをデータ記憶手段に展開し、一定の割込信号のタイミングで、前記データ記憶手段内のデータを出力するデータ処理方法において、

前記データ記憶手段内のデータを出力する間に、前記データ記憶手段に展開する圧縮データが格納されている前記プログラム記憶手段上のアドレスと、その圧縮データが展開される前記データ記憶手段上の展開先のアドレスとを登録バッファに登録し、

前記データ記憶手段内のデータを出力し終わってから前記割込信号が発生するまでの間に、前記登録バッファに登録された各アドレスに基づいて、前記プログラム記憶手段上の圧縮データを前記データ記憶手段に展開することを特徴とするデータ処理方法。

【請求項２】

プログラム及び圧縮データが格納されるプログラム記憶手段と、前記プログラム記憶手段に格納された圧縮データが展開されるデータ記憶手段と、一定のタイミングで割込信号を発生する割込信号発生手段と、前記プログラム記憶手段内の圧縮データを前記データ記憶手段に展開する展開手段と、前記プログラム記憶手段内のプログラムに基づいて各部の制御を行うと共に、前記割込信号のタイミングで、前記データ記憶手段に展開されたデータを出力するＣＰＵとからなるデータ処理装置において、

前記データ記憶手段に展開する圧縮データが格納されている前記プログラム記憶手段上のアドレスと、その圧縮データが展開される前記データ記憶手段上の展開先のアドレスとが

登録される登録バッファと、前記ＣＰＵにより前記データ記憶手段に展開されたデータが出力される間に、前記圧縮データについての各アドレスを前記登録バッファに登録する登録手段とを具備し、
前記展開手段は、
前記ＣＰＵにより前記データ記憶手段に展開されたデータが出力され終わってから前記割込信号が発生するまでの間に、前記登録バッファに登録された各アドレスに基づいて前記プログラム記憶手段上の圧縮データを前記データ記憶手段に展開することを特徴とするデータ処理装置。
【請求項３】
プログラム及び画像の圧縮データが格納されるプログラム記憶手段と、ディスプレイに表示される画像に対応した画像データを記憶する画像データ記憶手段と、一定のタイミングで割込信号を発生する割込信号発生手段と、前記プログラム記憶手段内の圧縮データを前記画像データ記憶手段に展開する展開手段と、前記プログラム記憶手段内のプログラムに基づいて各部の制御を行うと共に、前記割込信号のタイミングで、１フレーム毎に前記画像データ記憶手段に展開された画像データをディスプレイに表示させるＣＰＵとからなる画像処理装置において、
前記ディスプレイに表示される画像について、その圧縮データが格納されている前記プログラム記憶手段上のアドレスと、その圧縮データが展開される前記画像データ記憶手段上の展開先のアドレスとが登録される登録バッファと、
前記ディスプレイに１フレーム分の画像が表示される間に、前記圧縮データについての各アドレスを前記登録バッファに登録する登録手段とを具備し、
前記展開手段は、
前記ディスプレイに１フレーム分の画像を表示し終わってから前記割込信号が発生するまでの間に、前記登録バッファに登録された各アドレスに基づいて前記プログラム記憶手段上の圧縮データを前記画像データ記憶手段に展開することを特徴とする画像処理装置。
【請求項４】
プログラム記憶手段に格納されたプログラムに基づき、一定の割込信号のタイミングで画面が表示されるようにＣＰＵの処理が行われ、前記プログラム記憶手段に格納された圧縮データがデータ記憶手段に展開されるデータ処理方法であって、
前記プログラムのゲームの処理中に、前記データ記憶手段に展開する圧縮データが格納されている前記プログラム記憶手段上のアドレスと、その圧縮データが展開される前記データ記憶手段上の展開先のアドレスとを登録バッファに登録し、前記プログラムのゲームの処理終了後、前記割込信号が発生するまでの間に、前記登録バッファに登録された各アドレスに基づいて、前記プログラム記憶手段上の圧縮データを前記データ記憶手段に展開することを特徴とするデータ処理方法。
【発明の詳細な説明】
【０００１】
【産業上の利用分野】
本発明は、圧縮して記憶された画像等のデータを伸長して出力するデータ処理方法及び装置、並びに画像処理装置に係り、特に、ＣＰＵを効率よく使用することを可能とする技術に関する。
【０００２】
【従来の技術】
従来より、テレビゲーム機等に使用される画像処理装置としては、ＣＰＵの制御により、ＣＤ－ＲＯＭ等の記憶媒体から背景画とキャラクタ等の動画とについての画像データをビデオＲＡＭ等の画像メモリに転送し、画像メモリから適宜画像データを呼び出してビデオ信号として出力する方式の装置が知られている。このような装置においては、１画面毎に、ＲＯＭに格納されたゲームプログラム及び操作者が外部から入力する信号に基づき、キャラクタの位置を演算して画像データとすると同時に、画像メモリに展開された背景についての画像データを読み出す。そして、このような画面を連続して切り換えることにより

、キャラクタが移動するように見せている。
【０００３】
ところで、このような画像処理装置においては、画面上に表示されているキャラクタに対して移動などの処理を行う場合に、そのゲームのシチュエーション毎にＣＰＵに加わる負担が異なっている。すなわち、キャラクタとして簡単な構造のものや１画面で移動させるキャラクタの数が少ない場合には、ＣＰＵが次の画面を表示させる処理を短時間で行うことができる。また、移動時に回転などの複雑な計算を必要とする場合に比較して、移動の方向が直線的な場合には、ＣＰＵの処理時間も短くなる。そのため、仮にＣＰＵの処理速度のまま画面の切換を行うと、複雑な処理を行う場合と簡単な処理を行う場合とで１画面毎に表示するスピードが異なってしまう。
【０００４】
しかし、ゲームの場合、ある特定のキャラクタはそのキャラクタに定められた速度で移動することが求められており、ＣＰＵの処理に時間がかかる画面と時間がかからない画面とで表示スピードが違うと、キャラクタの移動スピードが異なってしまい、ゲームとして不自然になったり、はなはだしい場合にはゲームが成立しなくなる。
【０００５】
これを防止するため、ラスタースキャンタイプの画像表示手段（ＣＲＴ）では、垂直同期信号に同期して発生する割込信号（以下、Ｖブランクインタラプトと呼ぶ）毎に１画面づつ画像表示を切り換えることで、ＣＰＵの負担量に関わりなく一定のタイミングで次の画面が表示されるようにしている。すなわち、負担が軽くＣＰＵの処理が済んでしまった場合でも、Ｖブランクインタラプトの割込があるまでは、次の画面の表示を待つことで、キャラクタの移動速度を一定にしている。
【０００６】
なお、このことは、ＣＰＵの処理量と画像の表示タイミングとの関係をキャラクタの移動を例にとって説明したが、他の処理の場合でも同様である。
【０００７】
このような処理の具体的なフローを、図７（ａ）及び（ｂ）に示す。また、図４に（ａ）ＣＰＵの処理のタイムチャート及び（ｂ）Ｖブランクインタラプトの発生タイミングを示す。まず、図７（ａ）のステップＳＡ１において、ＣＰＵは、Ｖブランクインタラプトが出力されたことを表すＶブランクフラグＶ＿ＦＬＧに「０」を設定する。次に、ステップＳＡ２において、ＣＰＵは、Ｖブランクフラグ Ｖ＿ＦＬＧに「１」が設定されているか否かを判断する。この判断結果が「ＮＯ」である間、すなわち、Ｖブランクインタラプトが出力されるまでの間（図４における時刻ｔ１から時刻ｔ２まで）は、このステップＳＡ２の判断を繰り返す。
【０００８】
一方、図４に示す時刻ｔ２において、垂直同期信号に応じてＶブランクインタラプトが発生すると、図７（ｂ）のステップＳＢ１において、ＣＰＵは、Ｖブランクフラグ Ｖ＿ＦＬＧに「１」を設定する。これにより、図７（ａ）のステップＳＡ２における判断結果が「ＹＥＳ」となり、ステップＳＡ３に進む。そして、ステップＳＡ３において、画像メモリから背景及び動画の画像データをそのまま読み出すか、もしくは動画の画像データに回転その他の演算処理を施して、ディスプレイへ転送し、ゲームの処理を実行する。この処理が終了すると、再びステップＳＡ１に戻り、Ｖブランクフラグ Ｖ＿ＦＬＧに「０」を設定する（図４における時刻ｔ３）。以降、Ｖブランクインタラプトが出力されると（時刻ｔ４）、再びゲームの処理を実行する。
【０００９】
【発明が解決しようとする課題】
上述した従来の画像処理装置においては、図７（ａ）に示すように、Ｖブランクフラグ Ｖ＿ＦＬＧが立つまで空ループを回り続け、無駄な処理を行うこととなる（ステップＳＡ２）。そして、Ｖブランクインタラプトが出力されるまで（図４における時刻ｔ３から時刻ｔ４まで）、ＣＰＵは待機して次の処理を行うことができず、能力を無駄に費やすことと

なっていた。
【００１０】
一方、ＣＰＵは、上述したように無駄な処理を行う反面、上記画像表示以外の処理も多々必要とされる。例えば、背景画などの画像データや次のステージのプログラムなどが記憶媒体に圧縮された状態で記録されている場合、上述したように、ゲームの進行に従って、これらの圧縮データを順次画像メモリに展開する作業がＣＰＵに要求される。これらの作業は、従来では、上記のようにＣＰＵが１画面ずつ行っている画像処理の間に行われていた。すなわち、ＣＰＵは、ある画面の表示が終了し、次のステージに移動してその画面を表示する前に、画像表示とは別処理であるデータの展開などの処理を行っていた。しかし、圧縮データの展開のようにＣＰＵに負担がかかり、処理時間を要する作業が次の画面表示の間に挿入されると、画像表示が円滑に行えなくなったり、ゲームの進行中に画面が一瞬止まるという事態が発生する場合があった。
【００１１】
本発明は、このような従来技術の問題点に鑑みて提案されたものであり、その目的は、ＣＰＵの能力を無駄に費やすことなく、効率よく圧縮データの展開を実行することができるデータ処理方法及び装置を提供することにある。
【００１２】
また、他の目的は、ＣＰＵの能力を無駄に費やすことなく、効率よく圧縮データの展開を実行することができる画像処理装置を提供することにある。
【００１３】
【課題を解決するための手段】
上記目的を達成するために、請求項１記載の発明は、プログラム記憶手段に格納されたプログラムに基づき、前記プログラム記憶手段に格納された圧縮データをデータ記憶手段に展開し、一定の割込信号のタイミングで、前記データ記憶手段内のデータを出力するデータ処理方法において、前記データ記憶手段内のデータを出力する間に、前記データ記憶手段に展開する圧縮データが格納されている前記プログラム記憶手段上のアドレスと、その圧縮データが展開される前記データ記憶手段上の展開先のアドレスとを登録バッファに登録し、前記データ記憶手段内のデータを出力し終わってから前記割込信号が発生するまでの間に、前記登録バッファに登録された各アドレスに基づいて、前記プログラム記憶手段上の圧縮データを前記データ記憶手段に展開することを特徴としている。
【００１４】
請求項２記載の発明は、プログラム及び圧縮データが格納されるプログラム記憶手段と、前記プログラム記憶手段に格納された圧縮データが展開されるデータ記憶手段と、一定のタイミングで割込信号を発生する割込信号発生手段と、前記プログラム記憶手段内の圧縮データを前記データ記憶手段に展開する展開手段と、前記プログラム記憶手段内のプログラムに基づいて各部の制御を行うと共に、前記割込信号のタイミングで、前記データ記憶手段に展開されたデータを出力するＣＰＵとからなるデータ処理装置において、前記データ記憶手段に展開する圧縮データが格納されている前記プログラム記憶手段上のアドレスと、その圧縮データが展開される前記データ記憶手段上の展開先のアドレスとが登録される登録バッファと、前記ＣＰＵにより前記データ記憶手段に展開されたデータが出力される間に、前記圧縮データについての各アドレスを前記登録バッファに登録する登録手段とを具備し、前記展開手段は、前記ＣＰＵにより前記データ記憶手段に展開されたデータが出力され終わってから前記割込信号が発生するまでの間に、前記登録バッファに登録された各アドレスに基づいて前記プログラム記憶手段上の圧縮データを前記データ記憶手段に展開することを特徴としている。
【００１５】
請求項３記載の発明は、図１に示すように、プログラム及び画像の圧縮データが格納されるプログラム記憶手段２と、ディスプレイに表示される画像に対応した画像データを記憶する画像データ記憶手段３と、一定のタイミングで割込信号を発生する割込信号発生手段１と、プログラム記憶手段２内の圧縮データを画像データ記憶手段３に展開する展開手段

４と、プログラム記憶手段２内のプログラムに基づいて各部の制御を行うと共に、割込信号のタイミングで、１フレーム毎に画像データ記憶手段２に展開された画像データをディスプレイに表示させるＣＰＵ５とからなる画像処理装置において、ディスプレイに表示される画像について、その圧縮データが格納されているプログラム記憶手段２上のアドレスと、その圧縮データが展開される画像データ記憶手段３上の展開先のアドレスとが登録される登録バッファ６と、ディスプレイに１フレーム分の画像が表示される間に、圧縮データについての各アドレスを登録バッファ６に登録する登録手段７とを具備し、展開手段４は、ディスプレイに１フレーム分の画像を表示し終わってから割込信号が発生するまでの間に、登録バッファ６に登録された各アドレスに基づいてプログラム記憶手段２上の圧縮データを画像データ記憶手段３に展開することを特徴としている。

【００１６】

【作用】

請求項１及び２記載の発明において、データ記憶手段に展開されたデータは、一定の割込信号のタイミングで出力される。この間、後に出力されるデータについて、その圧縮データが格納されているプログラム記憶手段上のアドレスと、その圧縮データが展開されるデータ記憶手段上の展開先のアドレスとが、登録バッファに登録される。そして、データの出力が終了して次の割込信号が発生するまでの間に、上記登録バッファに登録された各アドレスに基づいて、プログラム記憶手段上の圧縮データがデータ記憶手段に展開される。このため、従来は、割込信号が発生するまでの間にタイミングを合わせるために無駄な処理を行っていたが、本発明では、この時間に圧縮データの展開を行うため、後にこのデータを出力する場合の時間が短縮される。また、ＣＰＵの能力を無駄に費やすことなく、有効に利用することができる。

【００１７】

請求項３記載の発明において、ＣＰＵ５により、一定の割込信号のタイミングで、１フレーム分の画像データがディスプレイに表示される。この表示が行われている間、例えば、まだ表示されていない画像、もしくは、ゲーム機等の場合では次に表示されるべきステージの背景画等について、その圧縮データが格納されているプログラム記憶手段２上のアドレスと、その圧縮データが展開される画像データ記憶手段３上の展開先のアドレスとが、登録手段７によって登録バッファ６に登録される。そして、展開手段４により、当該フレームの表示が終了して次の割込信号が発生するまでの間に、上記登録バッファ６に登録された各アドレスに基づいて、プログラム記憶手段２上の圧縮データが画像データ記憶手段３に展開される。そして、例えばゲーム等の場合では、次のステージに移る際に、ＣＰＵ５によって、この画像データ記憶手段３に展開された画像データがそのまま読み出されて表示される。このため、処理時間の短縮化、及びＣＰＵの有効な使用を実現することができる。

【００１８】

【実施例】

以下、本発明による画像処理装置の一実施例について、図面を参照して説明する。

【００１９】

（１）実施例の構成

図２は、本発明の一実施例による画像処理装置の構成を示すブロック図である。この図において、１０はバスＢを介して装置全体を制御するＣＰＵであり、２０はゲームプログラム及び圧縮された画像データ等が格納されているＲＯＭである。３０は、垂直同期信号に応じてＶブランクインタラプトを出力する割込信号発生手段である。また、４０はディスプレイ５０に表示される背景画及び動画の画像データが展開されるＶＲＡＭであり、ＶＤＰ（Ｖｉｄｅｏ　Ｄｉｓｐｌａｙ　Ｐｒｏｃｅｓｓｏｒ　）６０によって書込み及び読出しが行われる。

【００２０】

ＶＤＰ６０は、ＲＯＭ２０に格納されている画像データをＶＲＡＭ４０に展開すると共に、水平同期信号に同期して、ＶＲＡＭ４０から画像データを読み出してディスプレイ５０

に供給する。７０は、展開される圧縮データのＲＯＭ２０上のアドレスとＶＲＡＭ４０上の展開先のアドレスとが登録される登録バッファである。ＣＰＵ１０は、上記圧縮データのＲＯＭ２０上のアドレスと展開されるべきＶＲＡＭ４０上のアドレスとを１組とし、それらの組の複数組（ｎ組とする）を一単位として登録バッファ７０に登録する。

【００２１】

（２）実施例の動作

次に、本実施例による画像処理装置の動作について説明する。図３（ａ）は、ＣＰＵ１０が実行する処理のメインルーチンを示すフローチャートであり、（ｂ）は、登録バッファ７０にデータを登録する登録ルーチンを示すフローチャートである。この図３（ｂ）に示す登録ルーチンは、ゲームプログラムに予めその呼び出しの位置が設定されており、ゲーム処理中の任意の箇所で実行されるようになっている。

【００２２】

なお、以下に示すＶブランクフラグＶ＿ＦＬＧは、ゲームの開始時、もしくは割込信号発生手段３０からＶブランクインタラプトが出力された時「１」となり、ゲームの処理が終了すると「０」となるフラグである。また、展開中フラグＦＬＧは、登録バッファ７０にデータが登録されると「１」となり、登録バッファ７０に登録されたデータに関する展開処理が終了すると「０」となるフラグである。

【００２３】

＜メイン処理＞

まず、ゲームの開始時、ＣＰＵ１０はＶＤＰ６０を制御して、最初のステージにおいてディスプレイ５０に表示すべき背景画等の画像データを、ＲＯＭ２０から読み出してＶＲＡＭ４０に転送させる。また、ＣＰＵ１０は、展開中フラグＦＬＧに「０」を設定し（ステップＳＣ１）、ＶブランクフラグＶ＿ＦＬＧに「０」を設定する（ステップＳＣ２）。

【００２４】

そして、ＶブランクフラグＶ＿ＦＬＧもしくは展開中フラグＦＬＧの値が「１」となるまで、処理待ち状態となる（ステップＳＣ３～ステップＳＣ４）。ここで、図４（ｂ）に示す時刻ｔ２において、割込信号発生手段３０からＶブランクインタラプトが出力されると、同図（ｃ）に示すように、ＣＰＵ１０はＶブランクフラグＶ＿ＦＬＧに「１」を設定する。これにより、ＣＰＵ１０は１フレーム分のゲームの処理を実行する（ステップＳＣ１０）。すなわち、ＶＤＰ６０はＣＰＵ１０の制御により、水平同期信号に同期して、ＶＲＡＭ４０から適宜画像データを呼び出してビデオ信号としてディスプレイ５０に出力する。

【００２５】

そして、１フレーム分の表示処理が終了すると、ＣＰＵ１０は、図４（ｃ）に示す時刻ｔ３においてＶブランクフラグＶ＿ＦＬＧに「０」を設定する（ステップＳＣ２）。その後、時刻ｔ４に割込信号発生手段３０からＶブランクインタラプトが出力されると、同様に次の１フレーム分のゲームの処理を実行する。

【００２６】

＜登録処理＞

一方、上記ゲームの処理中、ゲームプログラムにおいて登録ルーチンの呼び出しが入ると、ＣＰＵ１０は、図３（ｂ）の登録ルーチンを実行する。まず、ＣＰＵ１０は、登録バッファ７０に登録可能なエリアがあるか否か判断する（ステップＳＤ１）。そして、登録可能なエリアがある場合は、ゲームプログラムに基づき、ディスプレイ上にまだ表示されていない画像、もしくは、次のステージの表示に必要な画像等の圧縮データについて、そのＲＯＭ２０上のアドレスと展開すべきＶＲＡＭ４０上のアドレスとをｎ組（ｎは任意の数）、登録バッファ７０に登録する（ステップＳＤ２）。その後、ＣＰＵ１０は、展開中フラグＦＬＧに　「１」を設定して、登録ルーチンを終了する。

【００２７】

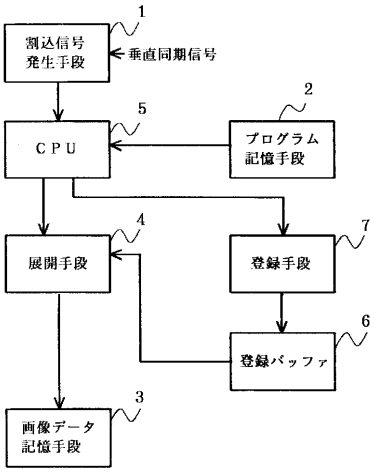
この登録ルーチンは、ゲームの処理中任意に実行されるが、登録バッファ７０に登録可能なエリアがなくなると、登録処理を実行せずに終了する。

【００２８】
＜ゲームの処理終了後（時刻ｔ３～ｔ４間）のメイン処理＞
次に、図４（ｃ）に示す時刻ｔ３から時刻ｔ４までの間、すなわち、ゲームの処理（ステップＳＣ１０）が終了してから次のＶブランクインタラプトが出力されるまでの間の処理について説明する。
【００２９】
まず、ゲームの処理中に、上述した登録ルーチンの実行により登録バッファ７０への登録がなされると、ＣＰＵ１０は、展開中フラグＦＬＧの設定値が「１」であると判断する（ステップＳＣ４）。また、ＶＤＰ６０は登録バッファ７０からまだデータを取得しておらず（ステップＳＣ５）、登録バッファ７０に登録されたデータについてまだ処理をおこなっていないので（ステップＳＣ６）、ＣＰＵ１０は、ＶＤＰ６０を制御して、登録バッファ７０から展開のためのデータを取得させる（ステップＳＣ７）。この場合、ＶＤＰ６０は、連続したアドレスに登録されたデータを所定の単位（例えばｎ組）で呼び出す。これにより、ＶＤＰ６０は、上記データ、すなわちＲＯＭ２０上のアドレスとＶＲＡＭ４０上のアドレスとに基づき、ＲＯＭ２０内の画像データをＶＲＡＭ４０に展開する（ステップＳＣ８）。
【００３０】
そして、ＶＤＰ６０が所定の単位のデータに基づく展開処理を終了すると（ステップＳＣ５）、ＣＰＵ１０は、登録バッファ７０上に未処理のデータが残っているか否かを判断する（ステップＳＣ６）。登録バッファ７０上にデータが残っている場合は、ＶＤＰ６０に、登録バッファ７０から展開のためのデータを取得させる（ステップＳＣ７）。そして、ＶＤＰ６０は、再びＶＲＡＭ４０への画像データの展開を行う（ステップＳＣ８）。
【００３１】
また、ＶＤＰ６０が登録バッファ７０に登録された全データについて展開処理を終了すると、ＣＰＵ１０は、展開中フラグＦＬＧに「１」を設定する（ステップＳＣ９）。
【００３２】
＜時刻ｔ４後のメイン処理＞
一方、割込信号発生手段３０からＶブランクインタラプトが出力されると、画像データの展開中と展開処理の終了後とに関わらず、Ｖブランクフラグ Ｖ＿ＦＬＧが「１」となり（ステップＳＣ３）、ゲームの処理が実行される（ステップＳＣ１０）。そして、このゲームの処理の間に上述した登録ルーチンが実行されると、ゲームの処理の終了後に、同様に画像データの展開処理が行われる（ステップＳＣ８）。また、前回の展開処理の際に展開の途中であった場合、もしくは、登録バッファ７０に登録された全データについて処理が終了していない場合は、続きのデータについて処理を行う。例えば、ＶＤＰ６０が取得したｎ組のデータについて処理がまだ終了していなかった場合は、展開が終了していないと判断して（ステップＳＣ５）、展開処理を続行する（ステップＳＣ８）。
【００３３】
＜具体例＞
次に、上述した本実施例について、具体的な画面を用いて説明する。例えば、あるゲームを構成するステージとして図５に示すステージ１があり、次のステージとして図６に示すステージ２があるものとする。また、ステージ１には背景画ＢＫ１が表示され、ステージ２には背景画ＢＫ２が表示され、これらのステージにおいて動画であるキャラクタＣＨＲが表示されるものとする。例えば、キャラクタＣＨＲは、ステージ１において、図５（ａ）に示す左端から同図（ｂ）に示す右端まで移動するものとする。
【００３４】
ステージ１において、ゲームの処理（ステップＳＣ１０）では、ＶＤＰ６０は、ＶＲＡＭ４０の対応する位置から背景画ＢＫ１の画像データとキャラクタＣＨＲの画像データとを読み出し、ディスプレイ５０に供給する。この時、キャラクタＣＨＲについては、フレーム毎にその移動位置に応じて画像データを読み出す。これにより、ディスプレイ５０には、キャラクタＣＨＲが図５（ａ）に示す位置から同図（ｂ）に示す位置まで移動するよう

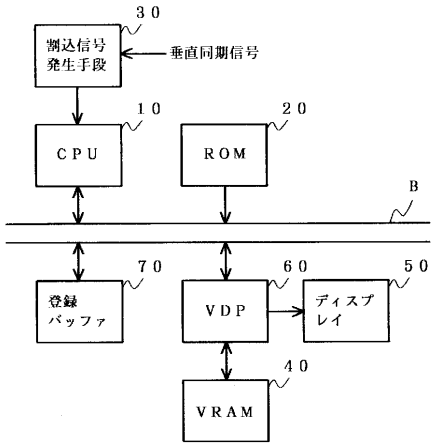
に表示される。
【００３５】
このステージ１におけるゲームの処理を実行している間、プログラム上の予め決められた位置で上述した登録ルーチンが実行される。すなわち、ＣＰＵ１０は、ゲームプログラムに基づき、次のステージ２において表示する背景画ＢＫ２の画像データについて、そのＲＯＭ２０上のアドレスとＶＲＡＭ４０上のアドレスとを登録バッファ７０に登録する。そして、ゲームの処理が終了し次のＶブランクインタラプトが発生するまでの間（上述した時刻ｔ３から時刻ｔ４までの時間）に、ＶＤＰ６０は、上記背景画ＢＫ２の画像データをＶＲＡＭ４０に展開する
（ステップＳＣ８）。
【００３６】
次に、ステージ１からステージ２に変わる時、ゲームの処理において、ＶＤＰ６０はＶＲＡＭ４０から、背景画ＢＫ２の画像データとキャラクタＣＨＲの画像データを読み出す。この時、ＶＲＡＭ４０に背景画ＢＫ２についての画像データが全て展開され終わっていない場合は、ＶＤＰ６０は残りの画像データの展開処理を行う。
【００３７】
（３）実施例の効果
以上のように、従来は、Ｖブランクインタラプトが発生するまでの時間（図４における時刻ｔ３から時刻ｔ４までの時間）に無駄な処理を行っていたが、本実施例では、可能な限りＲＯＭ１０に圧縮された画像データの展開を行うようにした。このように、ＣＰＵの空き時間を有効に利用して実行時間のかかる圧縮データの展開処理を実行するため、ＣＰＵを効率よく使用することができる。また、圧縮データの展開により画面が一時静止することがなく、ゲームの進行を滑らかに行うことが可能となる。
【００３８】
更に、ゲームの処理中に登録バッファ７０への登録ルーチンを実行するため、圧縮データの展開を行う際は、この登録バッファ７０を参照すればよく、展開の処理を効率よく行うことができる。
【００３９】
（４）その他の実施例
なお、本実施例では、画像データを展開する処理について示したが、これに限らず、ゲーム等の音楽もしくは効果音のデータをワークＲＡＭ等に展開する処理としてもよい。すなわち、１フレーム分のゲームの処理が終了してから次のＶブランクインタラプトが発生するまでの間に、次のステージで使用する音楽等のデータをワークＲＡＭに展開しておくようにする。または、ゲームのプログラム自体、もしくは各種テーブル等、ＣＰＵがデータとして扱うものについて、同様に登録してもよい。
【００４０】
また、本発明は上記実施例に限られるものではなく、テレビゲーム機以外、すなわち、パーソナルコンピュータ等のコンピュータの画像表示装置においても広く使用可能である。更に、画像表示装置に限らず、一定の割込信号のタイミングでデータを展開して出力するデータ処理装置であれば、同様な効果が得られる。
【００４１】
【発明の効果】
以上述べたように、本発明によれば、１フレーム分の画像を表示し終わってから割込信号が発生するまでの時間に、処理時間の長い圧縮データの展開を可能な限り行うようにしたので、ＣＰＵを効率よく使用することができると共に、処理時間の短縮化を実現することができる。従って、例えば画像が表示されるまでに画面が一瞬静止してしまうというような事態が極力発生しなくなり、ユーザに不快感を与えることがないという効果がある。
【図面の簡単な説明】
【図１】本発明による画像処理装置の基本構成を示すブロック図である。
【図２】本発明の一実施例による画像処理装置の構成を示すブロック図である。

【図３】同実施例におけるＣＰＵ１０の動作を説明するフローチャートであり、（ａ）はメインルーチン、（ｂ）は登録ルーチンを示す図である。
【図４】（ａ）ＣＰＵの動作を説明するためのタイムチャート、及び（ｂ）Ｖブランクインタラプトの出力タイミングを示す図である。
【図５】同実施例における具体的な画面の例を示す図である。
【図６】同実施例における具体的な画面の例を示す図である。
【図７】従来の画像処理装置のＣＰＵの動作を説明するフローチャートである。
【符号の説明】
１０…ＣＰＵ（ＣＰＵ５、展開手段４、登録手段７）
２０…ＲＯＭ（プログラム記憶手段２）
３０…割込信号発生手段（割込信号発生手段１）
４０…ＶＲＡＭ（データ記憶手段、画像データ記憶手段３）
５０…ディスプレイ
６０…登録バッファ（登録バッファ６）

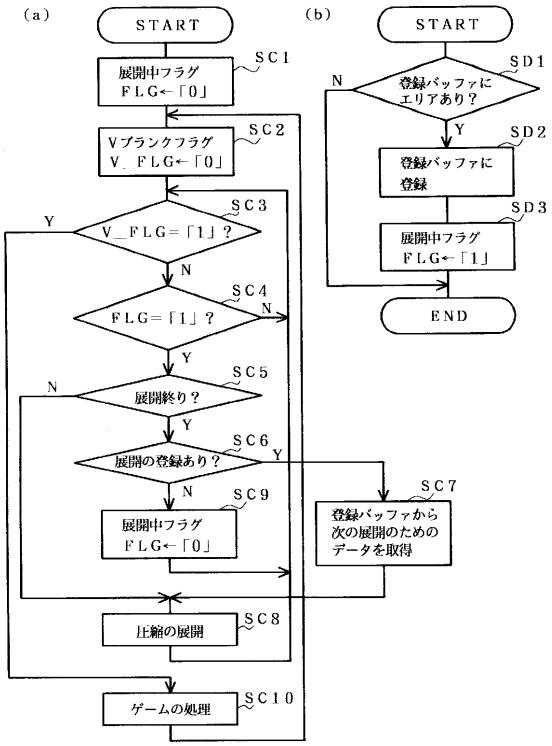
【図１】



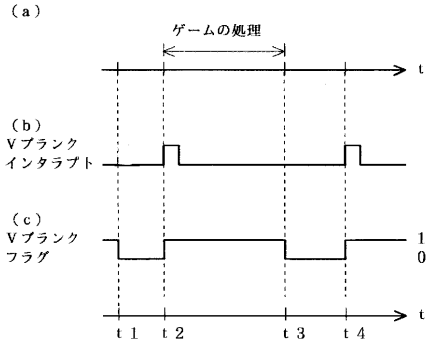
【図２】



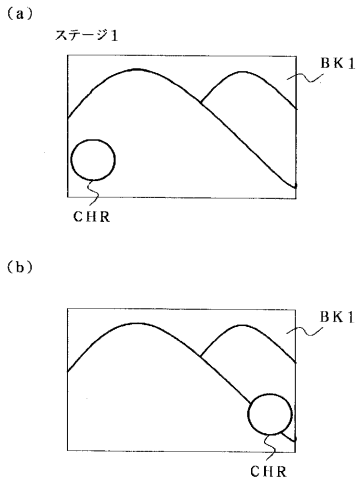
【図３】


(a)
START
展開中フラグ
ＦＬＧ←「０」
SC1
Ｖブランクフラグ
Ｖ_ＦＬＧ←「０」
SC2
SC3
Ｖ_ＦＬＧ＝「１」？
Y
N
SC4
ＦＬＧ＝「１」？
N
Y
SC5
展開終り？
N
Y
SC6
展開の登録あり？
Y
N
SC9
展開中フラグ
ＦＬＧ←「０」
SC7
登録バッファから
次の展開のための
データを取得
圧縮の展開
SC8
ゲームの処理
SC10
(b)
START
SD1
登録バッファに
エリアあり？
N
Y
SD2
登録バッファに
登録
SD3
展開中フラグ
ＦＬＧ←「１」
END

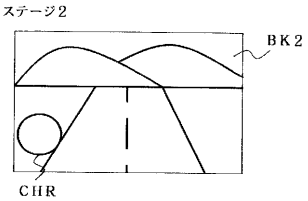
【図４】


(a)
ゲームの処理
t
(b)
Ｖブランク
インタラプト
(c)
Ｖブランク
フラグ
1
0
t
t1
t2
t3
t4

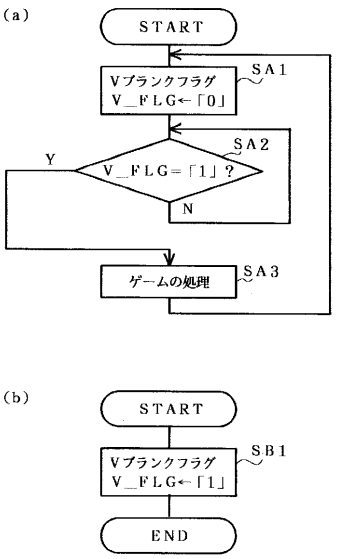
【図５】


(a)
ステージ1
BK1
CHR
(b)
BK1
CHR

【図６】


ステージ2
BK2
CHR

【図７】


(a)
START
Ｖブランクフラグ
Ｖ_ＦＬＧ←「０」
SA1
SA2
Ｖ_ＦＬＧ＝「１」？
Y
N
ゲームの処理
SA3
(b)
START
Ｖブランクフラグ
Ｖ_ＦＬＧ←「１」
SB1
END

フロントページの続き

(56)参考文献 特開平０７－１０４７３３（ＪＰ，Ａ）
特開平０７－０７３３３３（ＪＰ，Ａ）
特開平０７－０３６８１４（ＪＰ，Ａ）
特開平０６－１３９３３１（ＪＰ，Ａ）
特開平０６－１３９３３０（ＪＰ，Ａ）
特開平０６－１３８８５９（ＪＰ，Ａ）
特開平０６－０７５６５９（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.[7]，ＤＢ名)
G06T 1/20
A63F 13/00
G09G 5/00
G06T 13/00