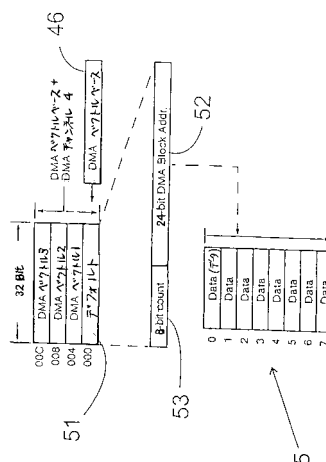




【特許請求の範囲】

【請求項 1】

DMA モジュールにおいて、すなわち

- a) アドレス可能なメモリ (5) の記憶スペースへの書き込みおよび／または読み出しアクセスを実行する DMA 制御ロジック (42) と、
- b) 記憶されたアドレスを隣接する記憶スペースに合わせて前方向調整するアドレスカウンタとを備える上記の DMA モジュールにおいて、

このアドレスカウンタが、メモリ (5) からのアドレス値の読み出しと、そのアドレス値のメモリへの書き込みとの間で、このアドレス値を正確に 1 回前方向調整するものとして設けられていることを特徴とする、上記の DMA モジュール。

10

【請求項 2】

DMA モジュールを操作する方法であって、下記の諸段階を含むもの：

- a) アドレス可能なメモリ (5) において第 1 のアドレスが表す記憶スペースに読み出しアクセスを実行して、そこで少なくとも 1 つの第 2 のアドレス (52) を読み出すこと、
- b) 第 2 のアドレス (52) を隣接する記憶スペースに合わせて前方向調整することと、第 2 のアドレス (52) を第 1 のアドレスが表す記憶スペースに記憶させることとを、任意の順序で行うこと、
- c) 第 2 のアドレス (52) が表す記憶スペースで書き込みまたは読み出しアクセスを実行すること。

【請求項 3】

段階 c) を実行する前にかならずカウント値 (53) を読み出すことと、そのカウント値 (53) を減分または増分して段階 c) の後でふたたび記憶することと、そのカウント値が事前指定された最終値に達するとその手順を終了することとを特徴とする、請求項 2 に記載の方法。

20

【請求項 4】

カウント値 (53) が、段階 a) の読み出しアクセスのときはそれとともに読み出され、段階 b) の書き込みアクセスのときはそれとともに書き込まれることを特徴とする、請求項 3 に記載の方法。

【請求項 5】

いずれの書き込みないし読み出しアクセスも、DMA モジュール (4) をメモリ (5) と結合するバス (8) の 1 サイクルの範囲内で行われることを特徴とする、請求項 2～4 のいずれかに記載の方法。

30

【請求項 6】

DMA モジュール (4) は、それがメモリ (5) へのアクセスを行う 1 つのバスサイクルで、同様にメモリ (5) と結合されているマイクロコントローラ (1) を停止させることを特徴とする、請求項 2～5 のいずれかに記載の方法。

【請求項 7】

周辺モジュールが DMA アクセスをリクエストし、周辺モジュールの識別により第 1 のアドレスを選択するという段階を含む、請求項 2～6 のいずれかに記載の方法。

【請求項 8】

周辺モジュールがその割り込みリクエスト回線を経由して DMA アクセスをリクエストすることを特徴とする、請求項 7 に記載の方法。

40

【請求項 9】

割り込みリクエスト回線を経由して DMA アクセスを作動するとき、割り込みリクエストは抑制されることを特徴とする、請求項 7 または 8 に記載の方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、DMA モジュールとその操作方法に関する。DMA モジュールは、定期的に反復されるデータ転送タスクを CPU から負担解除するため、マイクロコンピュータシス

50

テムやマイクロコントローラシステムで使用される。DMAモジュールとは、次のように特化されたプロセッサの一種と理解される。すなわち、アクセスしたい記憶領域の指定、たとえばスタートおよびストップアドレス、あるいはスタートアドレスおよびそれにつながるいくつかの記憶スペースの指定を受け取り、指定された領域のアドレスを順次迅速にアドレスバスにアウトプットして、記憶領域の迅速な読み出しと書き込みを可能にするプロセッサである。DMAモジュールは、2つの記憶領域の間、または1つの記憶領域と周辺装置との間のそのようなデータ転送を可能とし、そのためのCPUの処理タスクを必要としない。従ってDMAモジュールは、CPUに単純なデータ転送タスクの負担を解除し、平均的に達成可能なCPUパフォーマンスをこれにより向上する。

【背景技術】

10

【0002】

CPUとDMAモジュールがメモリにアクセスするための共通なバスを共有するならば、DMAモジュールがメモリにアクセスして書き込みまたは読み出しするバスサイクルごとに、CPUが同時にアクセスを試みるのを防止するため、CPUを停止しなければならない。この「サイクル スティール (cycle stealing)」と呼ばれる方法の場合、CPUはたしかにDMAの操作によってわずかにブレーキをかけられる。しかしこのようにして失われるCPUサイクルの個数は、CPU自体がデータ伝送を制御しなければならない場合よりも明らかに少ない。従ってDMAモジュールの使用によってCPUパフォーマンスの向上が得られる。

【0003】

20

DMA伝送は、常にスタートアドレスからストップアドレスに行われる。従来の方法によれば、両方のアドレスともDMAモジュールの内部レジスタに記憶される。これにより、ただ1つのバスサイクルの中で1つのアドレスをアウトプットし、そのアドレスが表す記憶場所を読み出しあるいは書き込んで、それに続く読み出し動作または書き込み動作のためアドレスカウンタを前方向調整することができる。

【0004】

このようなレジスタを2つ持つDMAモジュールは、その都度1つだけDMAプロセスを同時に実行できる。優先度の高い1つのDMAプロセスを実行できるようにするため、動作中のDMAプロセスを中断しなければならないとき、動作中の優先度の低いプロセスはその時点の数値とともに、DMAモジュールから送出されて中間記憶されなければならない。優先度の高いプロセスの対応するパラメータをロードして、そのプロセスを実行するためである。アドレスのこのようなロードアウトとロードインは、DMAモジュールとマイクロコントローラにブレーキをかけるので、望ましくないことである。

30

【0005】

考えられるのは、このようなDMAモジュールに多数のレジスタペアを装備し、これらのペアがさまざまなDMAプロセスのパラメータをそれぞれに含むことができるものとし、また動作中のDMAプロセスを優先度の高いプロセスのために中断するには、これらのペアを切り替えるだけでよいようにすることであろう。

しかしこの場合に生じる問題は、この種のレジスタは半導体基板上の必要面積が大きく、同じ大きさのメモリエlementがRAM中で必要とする面積の5倍に達することである。必要面積がこのように大きいため、多重処理可能なDMAモジュールの製造は高価なものとなる。

40

【発明の開示】

【発明が解決しようとする課題】

【0006】

本発明の課題は、面積をわずかしかなければならないDMAモジュールを得ること、ないしはこのような面積の小さいDMAモジュールを使用できるDMAモジュール操作方法を得ることである。

【課題を解決するための手段】

【0007】

50

この課題はまず、次のようなDMAモジュールの場合、すなわち従来の方法によって、アドレス可能なメモリの記憶スペースへの書き込みアクセスまたは読み出しアクセスを実行するDMA制御ロジックと、記憶されたアドレスを隣接する記憶スペースに合わせて前方向調整するアドレスカウンタとを備えるDMAモジュールの場合、次のようにして解決される。すなわち、このアドレスカウンタは、DMAモジュールの内部レジスタには働かず、メモリからのアドレス値の読み出しとメモリへのアドレス値の書き込みとの間に、このアドレス値を正確に1回前方向調整するものとして設けられることにより、解決される。言い換えれば、アドレスカウンタが、メモリからのアドレス値の読み出しとメモリへのアドレス値の書き込みとの間に、このアドレスカウンタを正確に1回、前に進めるように設けられることにより、解決される。アドレス値が読み出され、ないし書き込まれるメモリの記憶スペースは、従来の方法ではDMAモジュールに内蔵されていたレジスタの機能をこうして引き受ける。このようにしてDMAモジュールの必要スペースが減少し、重要な追加的利点として次のことが得られる。すなわち、多数のDMAプロセスの制御にこのDMAモジュールを使用することができ、これらのプロセスをたがい中断することができる。そのためにはメモリの多数の記憶スペースを、DMAブロックを指定するためのメモリとして用意すればそれでよい。

10

【0008】

このようなDMAモジュールの操作は、次の諸段階からなる。

a) アドレス可能なメモリにおいて第1のアドレスが表す記憶スペースに対する読み出しアクセスの実行。これはそこで少なくとも1つの第2のアドレスを読み出すためであって、そのアドレスとは、動作中のDMAプロセスでアクセスしなければならない次の記憶スペースを示すものである。

20

b) 第2のアドレスを隣接する記憶スペースに合わせて前方向調整する（または、第2のアドレスを隣接する記憶スペースに合わせて前進させる）。この第2のアドレスを、第1のアドレスが表す記憶スペースに記憶させる。

c) 第2のアドレスが表す記憶スペースに、書き込みアクセスまたは読み出しアクセスを実行する（この場合、最初に第2のアドレスを前方向調整して、次にその表す記憶スペースにアクセスするのか、あるいはその逆を行うのかはいつでもよいことである）。

【0009】

これまでDMAプロセスから次のプロセスに交代するには、まずレジスタ値を送出して交換する必要があったが、DMAプロセスに必要なパラメータはアドレス可能なメモリにつねに存在するので、その必要はなくなる。従って、さまざまなDMAプロセス間の迅速な切り替えが可能である。

30

1つのDMAプロセスがいつ完了したかを、DMAモジュールが検出できるためには、段階c)を実行する前にかならず、実行されるべきメモリアクセスの回数を表すカウント値を読み出すこと、そしてそのカウント値を減分し、段階c)を実行した後そのカウント値をふたたび記憶させることが合目的である。DMAモジュールはそのカウント値により、DMAプロセスがいつ完了したかを検出し、プロセス動作を正しい時点で終了することができる。

【0010】

40

メモリアクセスの回数を最小限にするためには、このカウント値を段階a)の読み出しアクセスごとにとともに読み出し、段階b)の書き込みアクセスごとにとともに書き込むのが合目的である。これが問題なく可能なのは、第2のアドレスとカウント値の総ビット数が、個々の読み出しアクセスで読み出されたデータワードの幅、すなわちデータバスの幅を超えない場合である。従って1つのDMAプロセスに必要なパラメータを読み出し、ないし書き込むには、それぞれ1つのバスサイクルで十分である。

【0011】

本発明のもう1つの視点は、周辺モジュールがDMA転送を「注文する」のに用いるDMAリクエスト信号を伝送するため、周辺モジュールにしばしばいずれにせよ設けられている割り込みリクエスト回線を利用することである。したがって有利な方法として、DM

50

A モジュールを割り込みコントローラへのリクエスト回線に接続し、いわば割り込みコントローラに前置する。

【0012】

このようにして、高価な割り込みを導入する以前に、(CPU パフォーマンスと比較して)「より安価な」いくつかの DMA 転送を導入する構造が生まれる。このような構造は、いちじるしい CPU 負荷なしで、長いインターバルにわたってデータを収集し、最後に CPU 関与の下に集中して処理を行うのに適している。

【0013】

本発明によれば、最初は DMA 制御の下にゆっくりと、周辺モジュールと RAM との間でデータ表を伝送するが、最後に CPU 割り込みルーチンを導入する。この CPU 割り込みルーチンにおいて、CPU が時間的に集中してたとえば既存のデータ表を処理し、次に DMA モジュールを次のデータ表伝送のためにコンフィギュレーションし、この DMA シーケンスをスタートすることができる。

【発明を実施するための最良の形態】

【0014】

本発明のそのほかの諸特徴と利点を、添付の図面の引用による下記の実施例をもとに説明する。

図 1 に図式的に示したマイクロコントローラシステムは、CPU 1、バスコントローラ 2、割り込み管理ユニット(割り込みコントローラ ICU) 3、DMA コントローラ 4、スタティック・ランダムアクセスメモリ(SRAM) 5、読み出し専用メモリ(ROM) 6、入/出力モジュール 7 を備える。これらのコンポーネント 1~7 は、アドレス、データ、制御一回線を持つ内部のバス 8 によって相互に結合されている。I/O モジュール 7 はさらに外部の 1 つまたは複数のデータ端末と結合され、データ端末からデータを受け取って内部のバス 8 に供給するか、あるいはデータ端末に内部のバス 8 からデータを出力する。I/O モジュール 7、DMA コントローラ 4、ICU 3 は、バス 8 から分離された多数の制御回線 9~12 によって相互に直接結合されているが、図 2 も併せて用いこれらの構造と機能をさらに詳しく説明する。

【0015】

図 2 は DMA コントローラ 4 の構造の詳細を示す。DMA コントローラ 4 の機能は 2 つに分かれるがその 1 つが、DMA アクセスを作動できるさまざまなプロセスの管理、ここではモジュール 7 を介してマイクロコントローラシステムと結合しているさまざまなデータ送信装置または受信装置の管理、すなわちあたえられた時点で複数のプロセスのうちいずれのために DMA を実行するのか決定することである。DMA コントローラ 4 のもう 1 つの部分は、具体的なメモリアccessの制御を担当する。これら 2 つの領域の管理回路を、ここでは DMA チャンネルロジック 41 ないし DMA 制御ロジック 42 と呼ぶ。DMA チャンネルロジック 41 は、いずれの DMA チャンネルに対しても、すなわちそのために DMA を作動できるいずれのプロセスに対しても 1 回存在する。リクエストプロセス x を識別する信号 DREQ x (x = 1、2、3 等) を制御回線 9 から加えられるインプットマルチプレクサ 43 は、DMA プロセスがペンディングであることを表明するフラッグを、RS フリップフロップでセットする。対応する DMA サイクルが動作終了しているとき、このフラッグは DMA チャンネルロジック 41 により自動的にリセットされる。リリースフラッグ EN は、プライオリティエンコーダー 45 に対してこのペンディングフラッグをマスクする。プライオリティエンコーダー 45 はすべての DMA チャンネルロジック 41 のアウトプットと結合されており、みずからのアウトプットに順序数を供給する。この順序数とは、ペンディングである DMA プロセスのチャンネルロジック 41 のうち優先度レベルが最も高いものに対応するものである。個々のプロセスの優先度レベルは、プライオリティエンコーダー 45 に記録される。DMA チャンネルロジック 41 は、さらに制御回線 10 を経由して作動プロセスに、肯定応答信号 DACK x をフィードバックする。この信号は DMA アクセスの実行を表示する。また一アクセス方向に応じて一メモリ 5 から読み出されたデータ値が内部のバス 8 にあって受け入れ可能であること、あるいはプロセス

から供給されたデータ値が受け入れ済みであって、新たなデータ値を供給しなければならないことを、プロセスに通知する。

【0016】

スイッチ47がリクエスト信号回線DREQ_x (x = 1, 2, 3) に配置されている。このスイッチによって、回線DREQ_xに到着する信号を直接ICU3にループして、割り込みを作動できるようにするか、あるいはDMAチャンネルロジック41から生じた制御信号をICU3に供給するかのどちらかを、選択して行うことができる。回線12は、リクエスト信号DREQ_xがスイッチ47で回線11にループされるという条件のとき、ICU3とインタフェース7との間でデータを直接伝送するのに用いられる。

【0017】

DMA制御ロジック42は、1つのDMAサイクルの範囲内で、いずれの個々のアクセスの実行をも制御する。時間的にオーバーラップして実行できるさまざまなDMAプロセスを管理するため、DMA制御ロジック42は、SRAM5にファイルされているDMAベクトル表51にアクセスする。このDMAベクトル表51のスタートアドレスは、DMA制御ロジック42のDMAベクトルベース（表開始レジスタ）46に記録されている。DMAベクトル表の記入は32ビットワードであって、これらのワードはDMAプロセスの指定を形成し、図3に示すように、それぞれ24ビット幅のアドレス52と8ビット幅のカウント値とにより構成される。このアドレスは、SRAM5においてDMAアクセスを行いたいとされるブロックのスタートアドレスであり、カウント値はこのブロックの長さをバイトで表示する。

【0018】

このアドレスとカウント値は、DMAプロセスの開始前にその都度、たとえばCPU1によってDMAベクトル表51に書き込まれる。このカウント値をここでたとえば8ビットと想定すると、DMAブロックは最大 $2^8 = 256$ バイトの長さを持つ。当然のことながら、このDMAベクトルのスタートアドレスとカウント値への区分を任意にこれと異なるものとし、異なるブロック長さとすることも考えられる。

【0019】

DMA制御ロジック42がプライオリティエンコーダー45から、そのためアクセスを実行したいDMAプロセスの番号を受け取ると、制御ロジック42は、プライオリティエンコーダー45から供給された番号の4倍をベクトルベース46のアドレスに加算することにより、アクセスアドレスを計算する。プライオリティエンコーダー45から供給された番号が、数値1、2または3を取ることができるとき、制御ロジック42は、図3のベクトル表51で「DMA - ベクトル1」、「DMA - ベクトル2」、「DMA - ベクトル3」などと記載されたフィールドにアクセスすることができる。

【0020】

容易に理解できることであるがDMAコントローラ4は、その回路構造にこれといった変更を加えずに異なる個数のDMAプロセスを管理するために、容易に適合可能である。プロセスの個数に対する制限的要因は、プライオリティエンコーダー45のアウトプット幅、すなわちプライオリティエンコーダーがDMA作動側プロセスの番号を制御ロジック42に伝達する回線のビット数と、DMAベクトル表51の大きさとの2つだけである。

【0021】

図4は、1つのDMAプロセスの過程におけるマイクロコントローラシステム内部のさまざまな信号の推移を示す。この場合 f_{sys} は、CPUとDMAコントローラ4とが動作するときに用いるシステムクロックパルスを表す。 f_{sys} の1つの周期は、1つのCPUクロックパルスに相当する。信号DREQ_x (x = 1, 2, 3等) のパルスは、プロセスxによるDMAリクエストを表示する。このリクエストに対応が行われるまで、複数のクロックサイクルが経過することがある。対応が行われると、DMAコントローラ4からメモリ制御に供給される信号DACCが、低いレベルから高いレベルに切り替わる。これは、DMAコントローラ4がバスを支配していることと、CPU1が書き込みまたは読み出しアクセスをしてはならないことを表示するためである。レベルの高いDACCをと

10

20

30

40

50

なうクロックサイクルの最初のサイクルで、DMAコントローラ4は、ベクトル「DMA - ベクトル1」、「DMA - ベクトル2」または「DMA - ベクトル3」のうち1つのアドレスを、バス8のアドレス回線Aに供給する。これらのベクトルのアドレスは、リクエスト側プロセスの識別と、ベクトルベース46に記録されたアドレスとにより計算される。これらベクトルの表51が存在するSRAM5はそれに対して、対応するアドレスに記録されたDMAブロックの指定（スタートアドレスと伝送されるべきバイト数からなる）を、バス8のデータ回線に合わせて設定する。これらの数値は、DMAコントローラ4からアドレスレジスタないしカウンタに受け入れられる。

【0022】

続くクロックサイクルでは、DMAベクトルのアドレスがあらためてバス8のアドレス回線に合わせて設定される。同時に1だけ増分されたスタートアドレスと1だけ減分されたカウント値とが、バス8のデータ回線に供給される。これはそれまでのDMAベクトルにこれらの数値を上書きするためである。

【0023】

次のクロックサイクルでは、増分されたアドレスがアドレス回線にアウトプットされる。これは一DMAプロセスのアクセス方向に応じて一対応する記憶場所へ書き込みまたは読み出しアクセスを実行するためである。

ただ1つのDMAプロセスの複数の書き込みまたは読み出しアクセスが直接に、すなわち優先度がより高いDMAプロセスによって中断されることなく順次続く場合、第2のおよびいずれのその後のメモリアccessに対しても、DMAベクトルの読み出しの段階を省略できる。そこに記憶された指定は、DMAモジュールのレジスタにすでに記憶されているからである。従って、DMAベクトルが表す記憶場所は、次のようなレジスタの機能を持つ。それは、SRAM5にあるメモリアドレス、かつ次の書き込み／読み出しアクセスを行いたいアドレスのすぐ前のメモリアドレスを示すレジスタである。すなわち、あるDMAプロセスを初期設定するとき、DMAベクトルに書き込まれるアドレス自体には何らアクセスが行われず、それに続くアドレスに初めてアクセスが行われる。カウント値が0（ゼロ）に達すると、DMAプロセスは終了される。

【0024】

上記と異なる方法として、割り当てられたDMAベクトルが示す記憶場所でDMAプロセスを初期設定するとき、処理されるべきメモリブロックの大きさと、そこで書き込みまたは読み出し動作を実行したい最初のアドレスとが記憶されることも、もちろん意図できるであろう。その場合、上記図4に関連して説明した手順はわずかに変更されるだけであろう。DMAベクトルが表す記憶場所の読み出しには、そこに挙げられたアドレスへの書き込み／読み出しアクセスを直接続けることができよう。そして増分されたアドレスと減分されたカウント値は、それに続くクロックサイクルで、DMAベクトルが表す場所に記憶されることとなろう。

【0025】

一連の場合においてDMAプロセスは、ある1つの記憶スペースを読み出しまたはそれへ書き込むために3クロックサイクルを、そして続く記憶スペースいずれのためにも、CPUが停止しているクロックサイクルを少なくとも2サイクル必要とする。これは、次のようなDMAの場合よりも、すなわち、処理されるべきメモリブロックのアドレスがDMAコントローラのレジスタに直接記録されており、送出行われたメモリから得る必要がないDMAの場合よりもたしかに遅い。しかしこれは依然として、CPU1自体がこのタスクを実行しなければならない場合よりも明らかに速い。DMAコントローラ内に必要なレジスタの個数は、DMAコントローラ4が同時に処理できるDMAプロセスの回数に依存しないので、任意の回数のDMAプロセスを同時に処理できるDMAコントローラを構成するのは、本発明によって非常に簡単になる。

【産業上の利用可能性】

【0026】

本発明のDMAモジュールは、マイクロコンピュータシステムやマイクロコントローラ

10

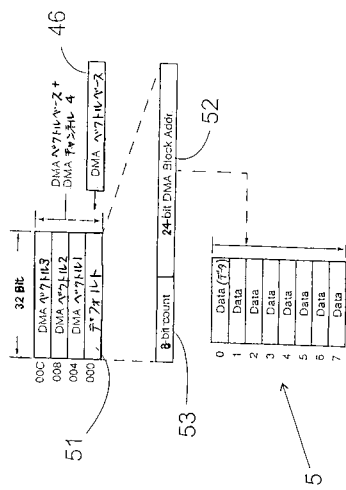
20

30

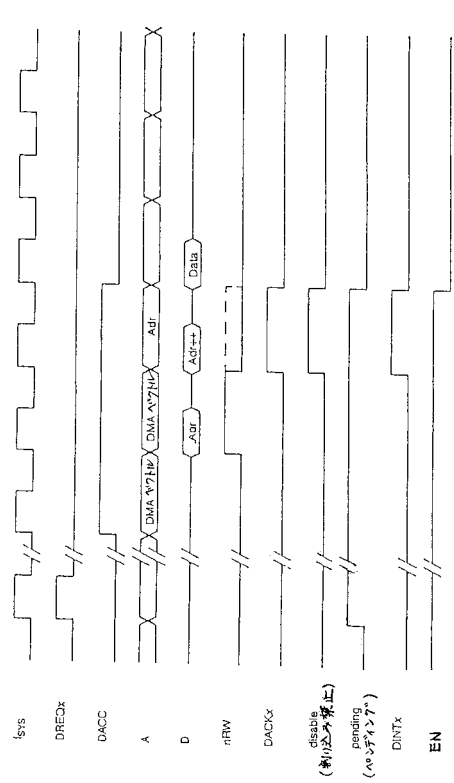
40

50

【図 3】



【図 4】



フロントページの続き

(74)代理人 100070563

弁理士 大村 昇

(74)代理人 100087620

弁理士 高梨 範夫

(72)発明者 ブルクハルト ギーベル

ドイツ、デー 7 9 2 1 1 デンツリンゲン、アムゼルヴェーク 2

Fターム(参考) 5B061 BC01 CC00 DD01 DD11 DD12 RR02 RR05