

TMS320VC5501/5502 DSP

ダイレクト・メモリ・アクセス (DMA) コントローラ

リファレンス・ガイド

TMS320VC5501/5502 DSP

ダイレクト・メモリ・アクセス (DMA) コントローラ

リファレンス・ガイド

この資料は、Texas Instruments Incorporated (TI) が英文で記述した資料を、皆様のご理解の一助として頂くために日本テキサス・インスツルメンツ (日本 TI) が英文から和文へ翻訳して作成したものです。資料によっては正規英語版資料の更新に対応していないものがあります。日本 TI による和文資料は、あくまでも TI 正規英語版をご理解頂くための補助的参考資料としてご使用下さい。製品のご検討およびご採用にあたりましては必ず正規英語版の最新資料をご確認下さい。

TI および日本 TI は、正規英語版にて更新の情報を提供しているにもかかわらず、更新以前の情報に基づいて発生した問題や障害等につきましては如何なる責任も負いません。



ご注意

日本テキサス・インスツルメンツ株式会社（以下TIJといたします）及びTexas Instruments Incorporated (TIJの親会社、以下TIJ及びTexas Instruments Incorporatedを総称してTIといたします)は、その製品及びサービスを任意に修正し、改善、改良、その他の変更をし、もしくは製品の製造中止またはサービスの提供を中止する権利を留保します。従いまして、お客様は、発注される前に、関連する最新の情報を取得して頂き、その情報が現在有効かつ完全なものであるかどうかをご確認下さい。全ての製品は、お客様とTIとの間に取引契約が締結されている場合は、当該契約条件に基づき、また当該取引契約が締結されていない場合は、ご注文の受諾の際に提示されるTIの標準契約約款に従って販売されます。

TIは、そのハードウェア製品が、TIの標準保証条件に従い販売時の仕様に対応した性能を有していること、またはお客様とTIとの間で合意された保証条件に従い合意された仕様に対応した性能を有していることを保証します。検査およびその他の品質管理技法は、TIが当該保証を支援するのに必要とみなす範囲で行なわれております。各デバイスの全てのパラメーターに関する固有の検査は、政府がそれ等の実行を義務づけている場合を除き、必ずしも行なわれておりません。

TIは、製品のアプリケーションに関する支援もしくはお客様の製品の設計について責任を負うことはありません。TI製部品を使用しているお客様の製品及びそのアプリケーションについての責任はお客様にあります。TI製部品を使用したお客様の製品及びアプリケーションについて想定される危険を最小のものとするため、適切な設計上および操作上の安全対策は、必ずお客様にてお取り下さい。

TIは、TIの製品もしくはサービスが使用されている組み合わせ、機械装置、もしくは方法に関連しているTIの特許権、著作権、回路配置利用権、その他のTIの知的財産権に基づいて何らかのライセンスを許諾するということは明示的にも黙示的にも保証も表明もしておりません。TIが第三者の製品もしくはサービスについて情報を提供することは、TIが当該製品もしくはサービスを使用することについてライセンスを与えよとか、保証もしくは是認するということの意味しません。そのような情報を使用するには第三者の特許その他の知的財産権に基づき当該第三者からライセンスを得なければならない場合もあり、またTIの特許その他の知的財産権に基づきTI からライセンスを得て頂かなければならない場合もあります。

TIのデータ・ブックもしくはデータ・シートの中にある情報を複製することは、その情報に一切の変更を加えること無く、且つその情報と結び付けられた全ての保証、条件、制限及び通知と共に複製がなされる限りにおいて許されるものとします。当該情報に変更を加えて複製することは不正で誤認を生じさせる行為です。TIは、そのような変更された情報や複製については何の義務も責任も負いません。

TIの製品もしくはサービスについてTIにより示された数値、特性、条件その他のパラメーターと異なる、あるいは、それを超えてなされた説明で当該TI製品もしくはサービスを再販売することは、当該TI製品もしくはサービスに対する全ての明示的保証、及び何らかの黙示的保証を無効にし、且つ不正で誤認を生じさせる行為です。TIは、そのような説明については何の義務も責任もありません。

なお、日本テキサス・インスツルメンツ株式会社半導体集積回路製品販売用標準契約約款もご覧下さい。

<http://www.tij.co.jp/jsc/docs/stdterms.htm>

Copyright © 2005, Texas Instruments Incorporated
日本語版 日本テキサス・インスツルメンツ株式会社

弊社半導体製品の取り扱い・保管について

半導体製品は、取り扱い、保管・輸送環境、基板実装条件によっては、お客様での実装前後に破壊/劣化、または故障を起こすことがあります。

弊社半導体製品のお取り扱い、ご使用にあたっては下記の点を遵守して下さい。

1. 静電気

- 素手で半導体製品単体を触らないこと。どうしても触る必要がある場合は、リストストラップ等で人体からアースをとり、導電性手袋等をして取り扱うこと。
- 弊社出荷梱包単位（外装から取り出された内装及び個装）又は製品単品で取り扱いを行う場合は、接地された導電性のテーブル上で（導電性マットにアースをとったもの等）、アースをした作業者が行うこと。また、コンテナ等も、導電性のものを使うこと。
- マウンタやんだ付け設備等、半導体の実装に関わる全ての装置類は、静電気の帯電を防止する措置を施すこと。
- 前記のリストストラップ・導電性手袋・テーブル表面及び実装装置類の接地等の静電気帯電防止措置は、常に管理されその機能が確認されていること。

2. 温・湿度環境

- 温度：0～40℃、相対湿度：40～85%で保管・輸送及び取り扱いを行うこと。（但し、結露しないこと。）

- 直射日光があたる状態で保管・輸送しないこと。
3. 防湿梱包
 - 防湿梱包品は、開封後は個別推奨保管環境及び期間に従い基板実装すること。
 4. 機械的衝撃
 - 梱包品（外装、内装、個装）及び製品単品を落下させたり、衝撃を与えないこと。
 5. 熱衝撃
 - んだ付け時は、最低限260℃以上の高温状態に、10秒以上さらさないこと。（個別推奨条件がある時はそれに従うこと。）
 6. 汚染
 - んだ付け性を損なう、又はアルミ配線腐食の原因となるような汚染物質（硫黄、塩素等ハロゲン）のある環境で保管・輸送しないこと。
 - んだ付け後は十分にフラックスの洗浄を行うこと。（不純物含有率が一定以下に保証された無洗浄タイプのフラックスは除く。）

以上

以下余白

最初にお読みください

本書について

本書では、TMS320C55x™ (C55x™) DSP 生成の TMS320VC5501 および TMS320VC5502 デジタル・シグナル・プロセッサ (DSP) 上で利用できるダイレクト・メモリ・アクセス (DMA) コントローラの機能および動作について説明します。この DMA コントローラを使用すると、内部メモリ、外部メモリ、およびペリフェラルの間で、CPU からの干渉を受けずに、CPU の動作のバックグラウンドでデータを移動することができます。

『TMS320VC5501/5502 DSP Direct Memory Access (DMA) Controller Reference Guide』(資料番号 SPRU613G) を翻訳しています。

表記規則

本書では、次の表記規則を使用しています。

- 多くの場合、16 進数は末尾に h が付いて表されています。たとえば、次の数字は 16 進数の 40 (10 進数の 64) です。

40h

- 同様に、2 進数は通常、末尾に b が付いて表されています。たとえば、次の数字は 10 進数の 4 を 2 進数で示したものです。

0100b

関連資料

C55x デバイスおよびそのサポート・ツールなどを解説した関連資料は、次のとおりです。関連資料は、www.ti.com から入手可能です。www.ti.com にアクセスして、検索ボックスに文献番号を入力してください。

『**TMS320C55x Technical Overview**』(文献番号 SPRU393) では、TMS320C5000™ DSP プラットフォームにおける最新版固定小数点 DSP である TMS320C55x DSP について説明しています。以前の製品と同様に、このプロセッサは、高性能で低消費電力での動作に最適です。この資料では、CPU のアーキテクチャ、拡張された低消費電力機能、および組み込みエミュレーション機能について説明しています。

『**TMS320C55x DSP CPU Reference Guide**』(文献番号 SPRU371) では、TMS320C55x DSP のアーキテクチャ、レジスタ、CPU の動作について説明しています。

『**TMS320C55x DSP Peripherals Reference Guide**』(文献番号 SPRU317)では、TMS320C55x DSP で使用可能なペリフェラル、インターフェイス、および関連するハードウェアについて説明しています。

『**TMS320C55x DSP Algebraic Instruction Set Reference Guide**』(文献番号 SPRU375)では、TMS320C55x DSP の各代数表記命令について説明しています。また、命令セットの要約、命令オペコードの一覧、およびニーモニック命令セットへの相互参照も記述しています。

『**TMS320C55x DSP Mnemonic Instruction Set Reference Guide**』(文献番号 SPRU374)では、TMS320C55x DSP の各ニーモニック命令について説明しています。また、命令セットの要約、命令オペコードの一覧、および代数表記命令セットへの相互参照も記述しています。

『**TMS320C55x Optimizing C/C++ Compiler User's Guide**』(文献番号 SPRU281)では、TMS320C55x™ の C/C++ コンパイラについて説明しています。この C/C++ コンパイラは、ISO 標準の C および C++ ソース・コードに対応し、TMS320C55x デバイス用のアセンブリ言語ソース・コードを生成します。

『**TMS320C55x Assembly Language Tools User's Guide**』(文献番号 SPRU280)では、TMS320C55x デバイス用のアセンブリ言語ツール(アセンブリ言語コードの開発に使用するアセンブラやリンカなどのツール)、アセンブラ擬似命令、マクロ、共通オブジェクト・ファイル・フォーマット、およびシンボリック・デバッグの擬似命令について説明しています。

『**TMS320C55x DSP Programmer's Guide**』(文献番号 SPRU376)では、TMS320C55x DSP の C とアセンブリのコードを最適化する方法、また DSP の特殊な機能と命令を使用するコードの書き方について説明しています。

商標

TMS320、TMS320C5000、TMS320C55x、および C55x は Texas Instruments の商標です。

その他の商標は各社の所有物です。

目次

DMA コントローラ	11
1 DMA コントローラの概要.....	12
1.1 DMA コントローラの重要な機能	12
1.2 DMA コントローラのブロック図	12
1.3 内部メモリへの DMA リクエストと CPU リクエスト	13
2 チャンネルとポート・アクセス	14
3 チャンネルの自動初期化機能.....	16
3.1 同一コンテキストによる自動初期化	19
3.2 異なるコンテキストによる自動初期化	20
4 サービス・チェーン.....	21
4.1 サービス・チェーンの例	22
5 データの単位：バイト、エレメント、フレーム、およびブロック.....	24
6 チャンネルの開始アドレス.....	25
6.1 メモリの開始アドレス	25
6.2 I/O 空間の開始アドレス	26
7 チャンネルのアドレスの更新.....	27
8 データ・バースト機能	28
9 データ・パッキング.....	29
10 ライト・ポスティング：内部メモリへのバッファリング・ライト.....	31
11 チャンネルの動作とイベントの同期	32
11.1 同期ステータスのチェック	33
11.2 失われた同期イベント	33
12 チャンネルの動作の監視	34
12.1 チャンネル割り込み	35
12.2 タイム・アウト・エラー条件	36
12.3 アドレス・エラー条件	36
12.4 バス・エラー割り込み	37
13 DMA 転送の遅れ.....	38

14	電力、エミュレーション、リセットについての考慮事項	39
14.1	DMA コントローラの消費電力の削減 (アイドル設定)	39
14.2	DMA コントローラのエミュレーション・モード	39
14.3	DSP リセット後の DMA コントローラ	39
15	DMA コントローラのレジスタ	40
15.1	グローバル制御レジスタ (DMAGCR)	41
15.2	グローバル・タイム・アウト制御レジスタ (DMAGTCR)	42
15.3	チャンネル制御レジスタ (DMACCR)	44
15.4	割り込み制御レジスタ (DMACICR) とステータス・レジスタ (DMACSR)	51
15.5	ソースおよびデスティネーション・パラメータ・レジスタ (DMACSDP)	57
15.6	ソース開始アドレス・レジスタ (DMACSSAL と DMACSSAU)	62
15.7	デスティネーション開始アドレス・レジスタ (DMACDSAL と DMACDSAU)	63
15.8	エレメント数レジスタ (DMACEN) とフレーム数レジスタ (DMACFN)	64
15.9	エレメント・インデックス・レジスタ (DMACSEI、DMACDEI) と フレーム・インデックス・レジスタ (DMACFI、DMACDFI)	66
15.10	ソース・アドレス・カウンタ (DMACSAC) と デスティネーション・アドレス・カウンタ (DMACDAC)	69
	改訂履歴	71

目次

図 1.	DMA コントローラ接続の概念ブロック図	13
図 2.	DMA 転送の 2 つの部分	14
図 3.	チャンネルのコンテキストを制御するためのレジスタ	15
図 4.	チャンネル制御レジスタ (DMACCR)	16
図 5.	同一コンテキストによる自動初期化シーケンス (REPEAT = 1)	19
図 6.	異なるコンテキストによる自動初期化シーケンス (REPEAT = 0)	20
図 7.	設定可能なサービス・チェーンの例	21
図 8.	3 つの DMA ポートに適用されるサービス・チェーン	23
図 9.	TMS320C55x DSP のメモリ・マップ	26
図 10.	TMS320C55x DSP の I/O マップ	27
図 11.	チャンネル割り込みリクエストのトリガ	35
図 12.	グローバル制御レジスタ (DMAGCR)	41
図 13.	グローバル・タイム・アウト制御レジスタ (DMAGTCR)	42
図 14.	チャンネル制御レジスタ (DMACCR)	44
図 15.	割り込み制御レジスタ (DMACICR) とステータス・レジスタ (DMACSR)	52
図 16.	ソースおよびデスティネーション・パラメータ・レジスタ (DMACSDP)	58
図 17.	ソース開始アドレス・レジスタ (DMACSSAL と DMACSSAU)	62
図 18.	デスティネーション開始アドレス・レジスタ (DMACDSAL と DMACDSAU)	64
図 19.	エレメント数レジスタ (DMACEN) とフレーム数レジスタ (DMACFN)	65
図 20.	エレメント・インデックス・レジスタ (DMACSEI、DMACDEI) と フレーム・インデックス・レジスタ (DMACSFI、DMACDFI)	68
図 21.	チャンネル・ソース・アドレス・カウンタ (DMACSAC) と チャンネル・デスティネーション・アドレス・カウンタ (DMACDAC)	69

表目次

表 1.	イベント駆動ポートと非イベント駆動ポート	14
表 2.	チャンネル制御レジスタ (DMACCR) のフィールドの説明	17
表 3.	図 8. に示される動作	22
表 4.	DMA 転送の開始アドレスの定義に使用されるレジスタ	25
表 5.	DMA コントローラによるデータ・パッキング	29
表 6.	32 ビット・データ・パッキングの条件†	30
表 7.	16 ビット・データ・パッキングの条件†	30
表 8.	イベント駆動ポートと非イベント駆動ポート	32
表 9.	DMA コントローラの動作イベントおよび関連ビットと割り込み	34
表 10.	DMA コントローラのレジスタ	40
表 11.	DMAGCR のビット・フィールドの説明	42
表 12.	DMAGTCR のビット・フィールドの説明	43
表 13.	DMACCR のビット・フィールドの説明	45
表 14.	TMS320VC5501/5502 DSP の同期イベント・マッピング	50
表 15.	DMACICR のビット・フィールドの説明	52
表 16.	DMACSR のビット・フィールドの説明	55
表 17.	DMACSDP のビット・フィールドの説明	58
表 18.	DMACSSAL のビット・フィールドの説明	63
表 19.	DMACSSAU のビット・フィールドの説明	63
表 20.	DMACDSAL のビット・フィールドの説明	64
表 21.	DMACDSAU のビット・フィールドの説明	64
表 22.	DMACEN のビット・フィールドの説明	65
表 23.	DMACFN のビット・フィールドの説明	65
表 24.	DMACSEI のビット・フィールドの説明	68
表 25.	DMACSFI のビット・フィールドの説明	68
表 26.	DMACDEI のビット・フィールドの説明	69
表 27.	DMACDFI のビット・フィールドの説明	69
表 28.	DMACSAC のビット・フィールドの説明	69
表 29.	DMACDAC のビット・フィールドの説明	69
表 30.	文書改訂履歴	71

DMA コントローラ

本書では、TMS320VC5501/TMS320VC5502 デジタル・シグナル・プロセッサ (DSP) のダイレクト・メモリ・アクセス (DMA) コントローラについて説明します。この DMA コントローラを使用すると、内部メモリ、外部メモリ、およびペリフェラルの間で、CPU からの干渉を受けずに、CPU の動作のバックグラウンドでデータを移動することができます。

1 DMA コントローラの概要

CPU の動作のバックグラウンドで、DMA コントローラは、内部メモリ、外部メモリ、およびオンチップ・ペリフェラルの間でデータを移動できます。

1.1 DMA コントローラの重要な機能

DMA コントローラには、以下のような重要な機能があります。

- DMA は、CPU と独立して動作します。
- 4 つの標準ポート。内部メモリのデュアル・アクセス RAM (DARAM) 用 2 ポート、外部メモリ用 1 ポート、およびペリフェラル用が 1 ポートがあります。
- 6チャンネルのDMA。DMA コントローラは、標準ポート間での6つの独立したブロック転送のコンテキストを各チャンネルで監視します。
- 各チャンネルごとに優先順位の高低を設定するビット。詳細については、21 ページの 4 節「サービス・チェーン」を参照してください。
- イベント同期。各チャンネルの設定イベントに応じて DMA 転送を実行させることができます。詳細については、32 ページの 11 節「チャンネルの動作とイベントの同期」を参照してください。
- 各チャンネルの割り込み。各チャンネルは、特定のイベント動作完了時に CPU 割り込みをかけることができます。34 ページの 12 節「チャンネルの動作の監視」を参照してください。
- データ転送のソースおよびディスティネーションのアドレス変更をソフトウェアで設定できます。
- 専用のアイドル・ドメイン。DMA コントローラは、このドメインをオフにすることで、ローパワー状態に移行できます。C55x DSP 上の各マルチチャンネル・バッファード・シリアル・ポート (McBSP) には、McBSP が DMA コントローラを必要とする際に、DMA ドメインをこのアイドル状態から一時的に移行させる機能があります。

DMA コントローラのプログラムに使用するレジスタについては、40 ページの 15 節「DMA コントローラのレジスタ」を参照してください。

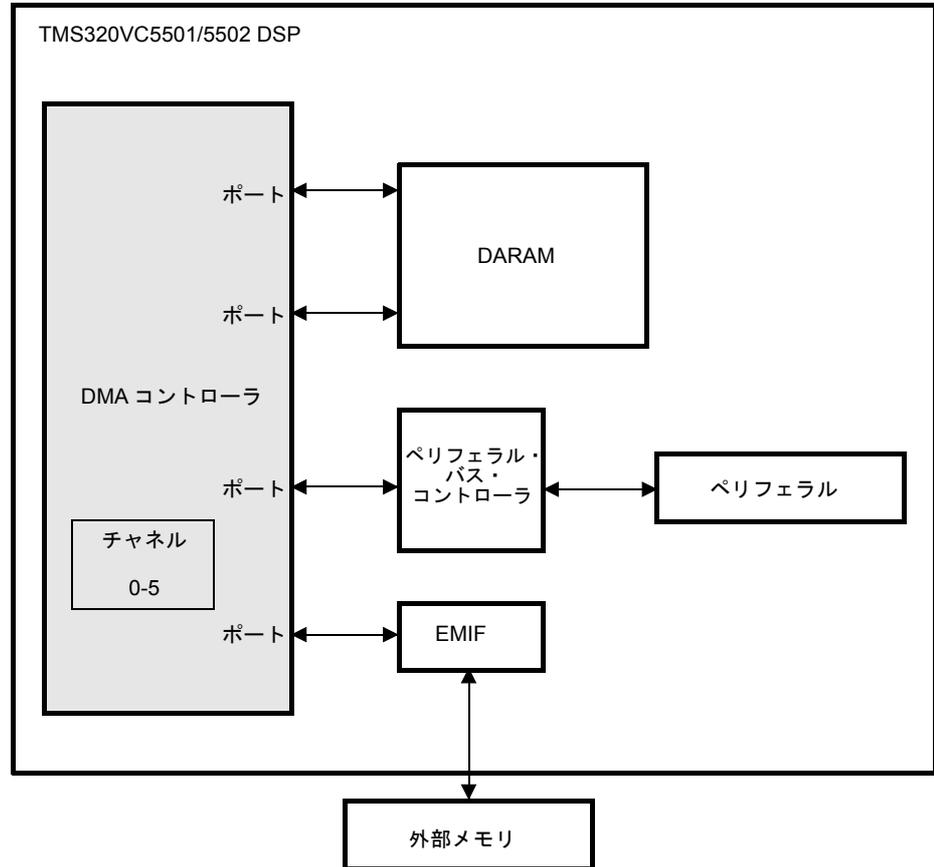
1.2 DMA コントローラのブロック図

図 1 は、DSP の DMA コントローラとその他の部分の接続の概念図です。DMA コントローラには、以下の 4 つのポートがあります。

- 内部メモリのデュアル・アクセス RAM (DARAM) 用に 2 ポート。参照しやすいように、本書では、これらの 2 つのポートは内部メモリ・ポート 0 および内部メモリ・ポート 1 と呼ばれます。
- 外部メモリ用に 1 ポート。外部メモリ・インターフェイス (EMIF) は、外部メモリへのポートに接続します。
- ペリフェラル用に 1 ポート。ペリフェラル・バス・コントローラは、ペリフェラルへのポートに接続します。

ポート間のデータ転送は、6つのDMAチャンネルで発生します（DMAチャンネルについては14ページを参照）。複数のチャンネルが同時に同じポートへのアクセスをリクエストすることは可能です。同時に発生するリクエストを調停するために、DMAコントローラには、各ポートによって使用されるプログラマブル・サービス・チェーンが1つ存在します。サービス・チェーンの詳細については、21ページを参照してください。

図1. DMAコントローラ接続の概念ブロック図



1.3 内部メモリへのDMAリクエストとCPUリクエスト

CPUとDMAコントローラが同時に、内部メモリの同じDARAMブロックへのアクセスを要求した場合、DMAからの要求よりもCPUからの要求が常に優先されます。DMAからDARAMブロックへの要求は、ほかにCPUからの要求が存在しないときに処理されます。各DARAMブロックの開始および終了アドレスの具体的な情報についてはデバイス別のデータ・マニュアルを参照してください。

2 チャンネルとポート・アクセス

DMA コントローラには、4 つのポート（DARAM 用に 2 ポート、外部メモリ用に 1 ポート、およびペリフェラル用に 1 ポート）間でデータを転送するための、**チャンネル**と呼ばれる 6 つのパスがあります。各チャンネルは、あるポート（ソース）からデータをリードし、同じポートまたは別のポート（デスティネーション）にデータをライトします。

各チャンネルは先入れ先出し（FIFO）バッファであり、データ転送はこのバッファを使用して 2 段階で実行されます（図 2. を参照）。

ポート・リード・アクセス ソース・ポートからチャンネル FIFO バッファへのデータ転送
ポート・ライト・アクセス チャンネル FIFO バッファからデスティネーション・ポートへのデータ転送

各チャンネルの FIFO バッファは、32 ビットワード 8 個から構成されます。

図 2. DMA 転送の 2 つの部分



DMA の各ポートは、イベント駆動と非イベント駆動にカテゴリ分けされます（表 1. を参照）。ポート間のこの違いは、32 ページの 11 節「チャンネルの動作とイベントの同期」で説明するように、同期を使用したデータ転送に大きく影響します。

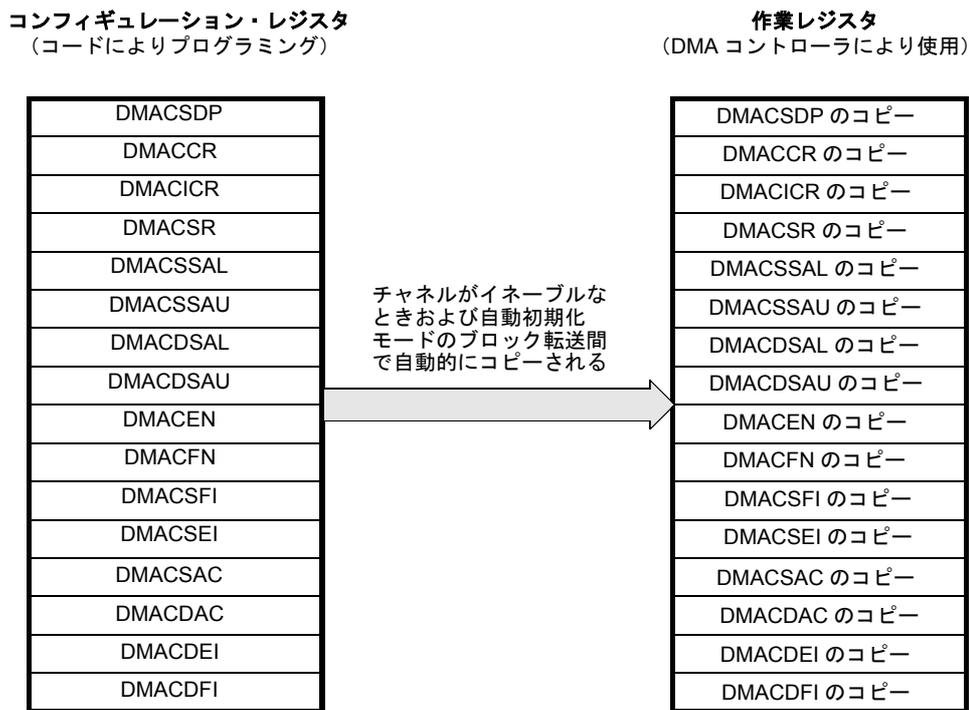
表 1. イベント駆動ポートと非イベント駆動ポート

ポート	カテゴリ
DARAM	非イベント駆動
外部メモリ	非イベント駆動
ペリフェラル	イベント駆動

チャンネルで転送が発生する条件セットは、**チャンネル・コンテキスト**と呼ばれます。6つのチャンネルにはそれぞれチャンネル・コンテキストをプログラミングおよび更新するためのレジスタ構造が含まれています（図3.を参照）。**コンフィギュレーション・レジスタ**は、コードによって変更します。データ転送を行う際は、**コンフィギュレーション・レジスタ**の内容が**作業レジスタ**にコピーされ、DMA コントローラが**作業レジスタ**の値を使用してチャンネルの動作を制御します。**コンフィギュレーション・レジスタ**から**作業レジスタ**へのコピーは、コードでチャンネルが有効にされる（DMACCRでEN=1）たびに行われます。また、自動初期化モードがオン（DMACCRでAUTOINIT=1）の場合も、そのコピーがブロック転送間で行われます。DMA コントローラのレジスタの詳細については、40 ページを参照してください。

一部の**コンフィギュレーション・レジスタ**は、DMA が**作業レジスタ**に基づいて現在の**コンテキスト**で動作している間に、次のブロック転送のためにプログラムすることができます。次の転送では、DMA を停止することなく、新しい**コンフィギュレーション**が使用されます。この方法で設定してはいけない**コンフィギュレーション・レジスタ**として、DMAGCR、DMAGTCR、DMACSDP、DMACCR、DMACICR、およびDMACSRがあります。DMA チャンネルの動作中にこれらの**レジスタ**を変更すると、チャンネルが予測不可能な動作をする可能性があります。

図3. チャンネルのコンテキストを制御するためのレジスタ



3 チャンネルの自動初期化機能

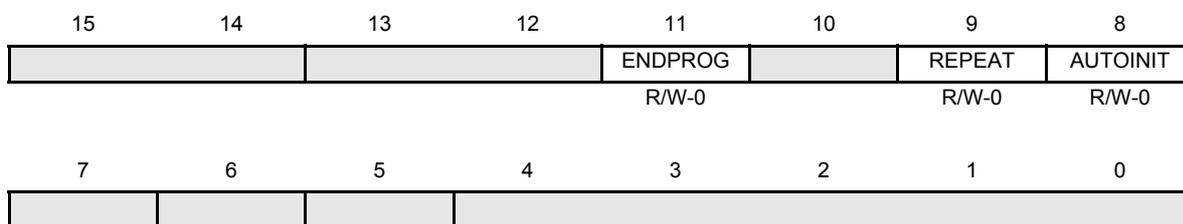
ブロック転送が完了した後（ブロック内のすべてのエレメントとフレームが移動された後）、DMA コントローラは自動的にチャンネルをディスエーブルにします。チャンネルを再び使用する必要がある場合は、CPU が新しいチャンネルコンテキストを再プログラムして DMA チャンネルを再度イネーブルにするか、または DMA コントローラが自動的に新しいコンテキストを初期化してチャンネルを再度イネーブルにします。

自動初期化を使用する場合は、ブロック転送が完了するたびに、DMA コントローラが自動的にチャンネル・コンテキストをコンフィギュレーション・レジスタから作業レジスタに再コピーし、チャンネルを再度イネーブルにすることによって、チャンネルが再動作できるようにします。自動初期化をイネーブルにするには、チャンネル・コントローラ・レジスタ (DMACCR) の AUTOINIT ビットをセットします。

DMACCR の REPEAT および ENDPROG の 2 つのビットは、自動初期化の操作中に使用されます。REPEAT は、CPU によるコンフィギュレーション・レジスタでコピーの準備ができたかの通知を DMA コントローラが待つかどうかを制御します。ENDPROG は、CPU と DMA コントローラ間でのレジスタ・コピーの処理をハンドシェイクするためのビットです。図 4. に DMACCR を示します。また、表 2. では AUTOINIT、REPEAT、および ENDPROG について説明します。DMACCR の詳細については、42 ページの 15.2 節「グローバル・タイム・アウト制御レジスタ (DMAGTCR)」を参照してください。

自動初期化を使用する方法は 2 つあります。1 つは各ブロック転送で同じチャンネル・コンテキストをリピートする方法、もう 1 つは各ブロック転送で新しいコンテキストを使用する方法です。この 2 つの方法について、以下の節で説明します。

図 4. チャンネル制御レジスタ (DMACCR)



凡例： R = リード、W = ライト、-r=DSP リセット後の値

表 2. チャネル制御レジスタ (DMACCR) のフィールドの説明

ビット	フィールド	値	説明
11	ENDPROG		<p>エンド・オブ・プログラミング・ビット。各 DMA チャンネルにつき、コンフィギュレーション・レジスタと作業レジスタの 2 つのレジスタ・セットが存在します。自動初期化 (AUTOINIT = 1) が原因でブロック転送が繰り返し発生する場合は、現在のブロック転送中にコンフィギュレーション・レジスタにライトすることによって、次の DMA 転送のコンテキストを変更できます。現在の転送の最後にコンフィギュレーション・レジスタの内容が作業レジスタにコピーされ、DMA コントローラは新しいコンテキストを使用して次の転送を開始します。自動初期化を正しく実行するには、DMA コントローラがコンフィギュレーション・レジスタの内容をコピーする前に、CPU がコンフィギュレーション・レジスタのプログラミングを完了する必要があります。</p> <p>DMA コントローラは、コンフィギュレーション・レジスタを作業レジスタにコピーした後で、自動的に ENDPROG ビットをクリアします。この後、CPU は、コンフィギュレーション・レジスタをプログラムすることによって、次の転送で使用する DMA チャンネル・コンテキストをプログラムできるようになります。</p> <p>CPU の動作を待つ自動初期化が行われるようにするには、以下のシーケンスに従います。</p> <ol style="list-style-type: none"> 1) REPEAT ビットをクリアして (REPEAT = 0)、自動初期化が ENDPROG = 1 になるまで待機するように設定します。 2) ENDPROG = 0 をポーリングします。ENDPROG = 0 は、DMA コントローラが前のコンテキストのコピーを完了したことを示します。この時点で、コンフィギュレーション・レジスタは、次の転送のためにプログラムできるようになります。 3) コンフィギュレーション・レジスタをプログラムします。 4) ENDPROG をセットして (ENDPROG = 1)、レジスタのプログラミングの完了を示します。
		0	コンフィギュレーション・レジスタはプログラミング可能 / プログラミング中
		1	プログラミング完了
9	REPEAT		<p>リピート条件ビット。自動初期化が選択されている (AUTOINIT = 1) チャンネルでは、REPEAT が 2 つの特殊なリピート条件のうちのどちらかを指定します。</p> <ol style="list-style-type: none"> 0 ENDPROG = 1 の場合のみリピート 現在の DMA 転送が完了したら、自動初期化はエンド・オブ・プログラミング・ビット (ENDPROG) がセットされるまで待機します。 1 ENDPROG に関係なくリピート 現在の DMA 転送が完了したら、自動初期化は ENDPROG が 0 または 1 のどちらであるかに関係なく実行されます。

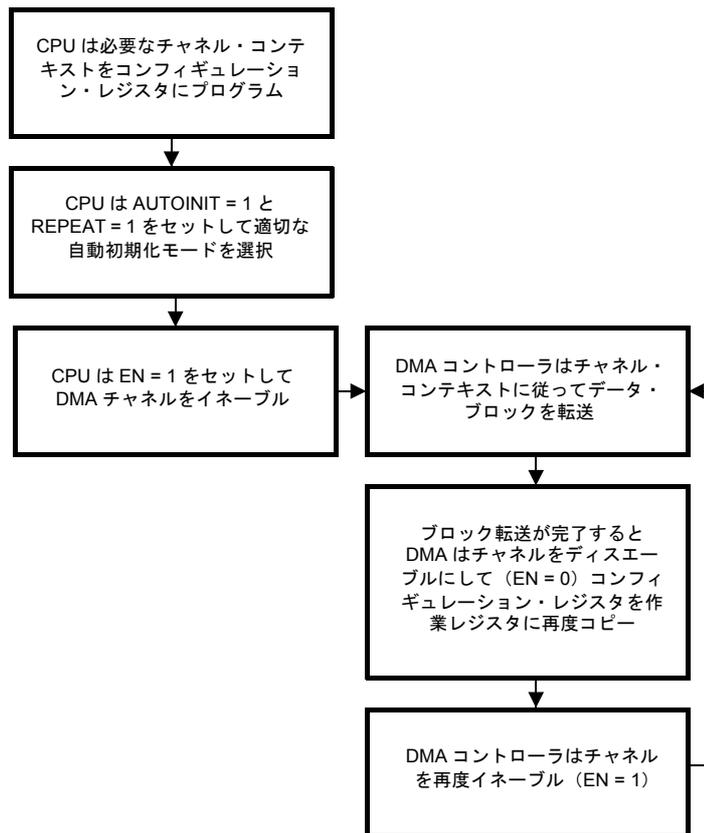
表 2. 「チャンネル制御レジスタ (DMACCR) のフィールドの説明」 (続き)

ビット	フィールド	値	説明
8	AUTOINIT		ビットの自動初期化。DMA コントローラは自動初期化をサポートします。自動初期化では、DMA ブロック転送間でチャンネルの自動再初期化が行われます。AUTOINIT を使用して、この機能をイネーブルまたはディスエーブルにします。
		0	自動初期化はディスエーブル チャンネルの動作は、現在のブロック転送が完了すると停止します。転送を即時停止するには、チャンネル・イネーブル・ビット (EN) をクリアします。
		1	自動初期化はイネーブル 現在のブロック転送が完了すると、DMA コントローラはチャンネルを再初期化し、新しいブロック転送を開始します。チャンネルの動作を停止するには、2 つの選択肢があります。 <input type="checkbox"/> 転送を即時停止するには、チャンネル・イネーブル・ビットをクリア (EN = 0) します。 <input type="checkbox"/> 現在のブロック転送の完了後に動作を停止するには、AUTOINIT をクリアします (AUTOINIT = 0)。

3.1 同一コンテキストによる自動初期化

チャンネルのコンテキストを繰り返し使用し、変更する必要がない場合は、DMA コントローラを AUTOINIT = 1、REPEAT = 1 に設定します。REPEAT = 1 の場合は、DMA コントローラは ENDPROG ビットの状態を無視します。CPU が DMA チャンネルを初期設定した後は、チャンネルが動作を継続するために、それ以外の CPU の干渉は必要ありません。このモードでの詳細なイベント・シーケンスを、図 5. に示します。

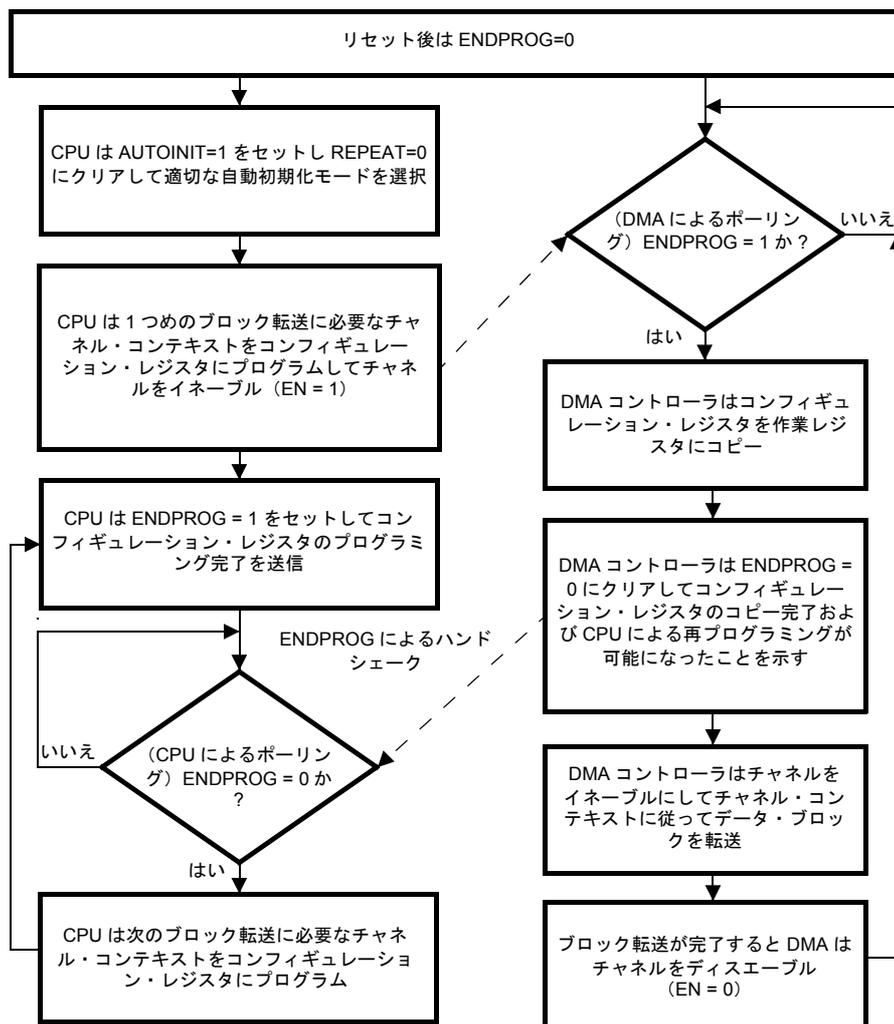
図 5. 同一コンテキストによる自動初期化シーケンス (REPEAT = 1)



3.2 異なるコンテキストによる自動初期化

チャンネルのコンテキストを繰り返し使用する必要があるが、各ブロック転送で異なる場合は、DMA コントローラを AUTOINIT = 1、REPEAT = 0 で設定する必要があります。REPEAT = 0 の場合は、CPU が ENDPROG = 1 とライトするまで、DMA コントローラは、コンフィギュレーション・レジスタをコピーするのを待ちます。これにより DMA のハンドシェイクが行われ、CPU がレジスタを設定している間に DMA がそのレジスタをコピーするのを防ぎます。このモードでの詳細なイベント・シーケンスを、図 6. に示します。

図 6. 異なるコンテキストによる自動初期化シーケンス (REPEAT = 0)

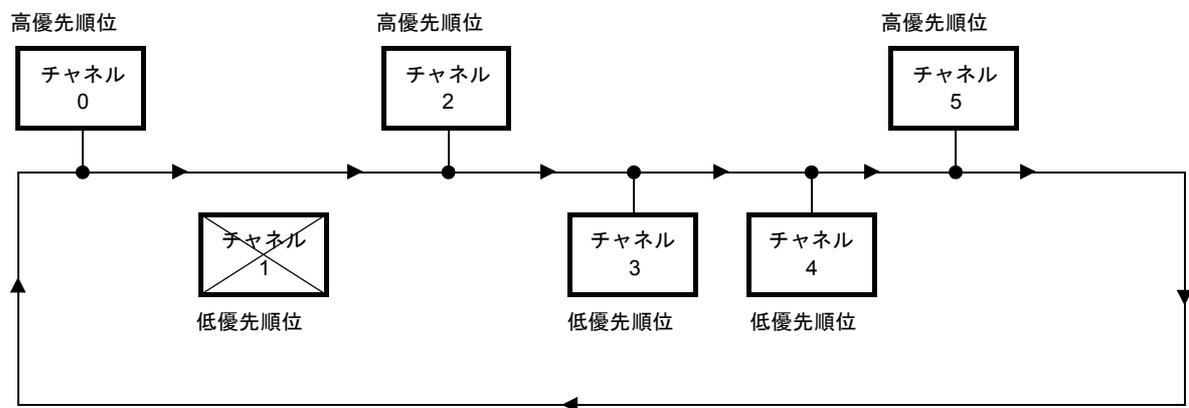


4 サービス・チェーン

DMA コントローラの各ポートは、6つのDMAチャンネルによって同時に送信されたアクセス・リクエストを調停することができます。各ポートには、独立して動作するサービス・チェーン、すなわちアクセス・リクエストを処理するための、ソフトウェアおよびハードウェアによって制御された仕組みが備わっています。4つのサービス・チェーンは独立して動作しますが、共通の設定を共有しています。たとえば、チャンネル2をディスエーブルにすると4つのポートすべてでチャンネル2がディスエーブルになり、チャンネル4の優先順位を高くすると4つのポートすべてでチャンネル4の優先順位が高くなります。設定可能なサービス・チェーンの例を図7. に示します。サービス・チェーンの重要な特性を、図の次のリストに示します。

22 ページの 4.1 節「サービス・チェーンの例」に、3つのポートに適用されるサービス・チェーンの設定例が示されています。

図7. 設定可能なサービス・チェーンの例



- チャンネルには、プログラマブルな優先順位レベルがあります。各チャンネルに対応して DMACCR の PRIO ビットが存在し、このビットで高い優先順位または低い優先順位を選択します。DMA コントローラが低い優先順位のアイテムを処理するのは、高い優先順位のアイテムの処理がすべて完了したか、またはストールされている場合だけです。DSP のリセット後は、すべてのチャンネルが低い優先順位になります。

図では、チャンネル0、2、および5が高い優先順位です（この3つのチャンネルでは PRIO = 1）。DMA チャンネル1、3、および4は低い優先順位です（この3つのチャンネルでは PRIO = 0）。

- チャンネルの位置は、サービス・チェーン内で固定されています。プログラムされている優先順位と関係なく、ポートは、チャンネルを、0、1、2、3、4、5、0、1、2、3、4、5 というように、サーキュラーなシーケンスで繰り返しチェックします。サービス・チェーン内の各位置で、ポートは、チャンネルの準備が完了し、処理可能な状態であることをチェックします。処理可能であれば処理を行い、それ以外の場合は次の位置にスキップします。DSP のリセット後は、ポートは、チャンネル0からサーキュラーなシーケンスを再開します。

- チャンネルは、個別にサービス・チェーンへの接続または切り離しを、ソフトウェアから実行できます。チャンネルがイネーブル（DMACCR で EN = 1）の場合は、サービス・チェーンに接続されます。チャンネルがディスエーブル（EN = 0）の場合は、サービス・チェーンから切り離されます。DSP のリセット後は、すべてのチャンネルが切り離されます。

図では、チャンネル 1 だけが切り離されています。ポートがチャンネルをサーキュラーなシーケンスで繰り返しチェックするときは、チャンネル 1 が再接続されるまでは、チャンネル 1 をスキップし続けます。

- チャンネルが同期イベントに関連付けられている場合は、同期イベントが発生するまで、チャンネルは DMA リクエストを生成しません（したがって、処理されることもありません）。

4.1 サービス・チェーンの例

図 8. は、DARAM ポート、外部メモリ・ポート、およびペリフェラル・ポートに適用される DMA サービス・チェーンを示しています。サービス・チェーンには、以下の特性がプログラムされています。

- チャンネル 0、2、および 5 は高い優先順位（DAMCCR で PRIO = 1）です。チャンネル 1、3、および 4 は低い優先順位（PRIO = 0）です。
- チャンネル 1、2、および 4 はイネーブル（DAMCCR で EN = 1）です。チャンネル 0、3、および 5 はディスエーブル（EN = 0）です。

表 3. にのポートの動作の概要を示します。

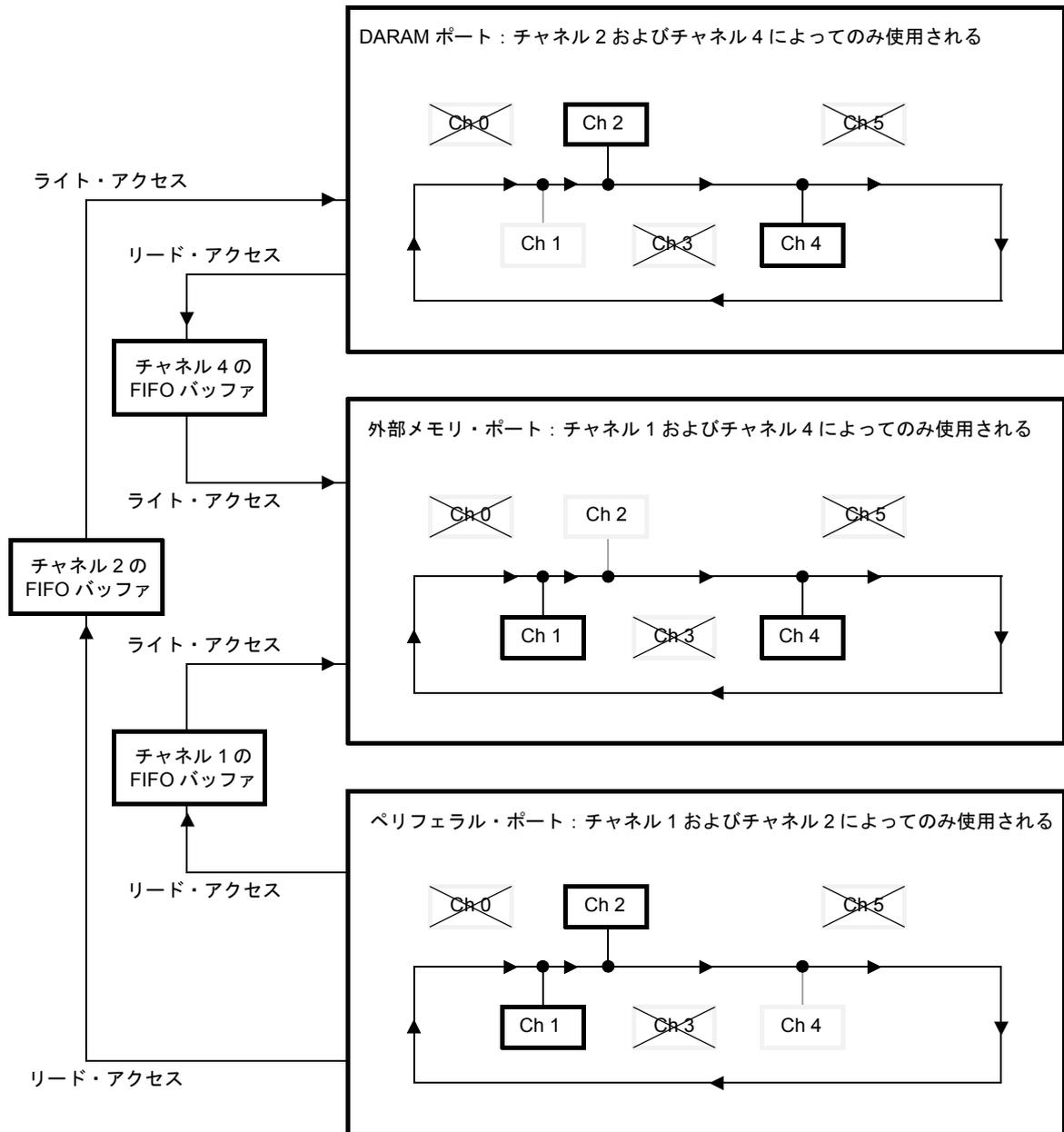
表 3. 図 8. に示される動作

ポート	このポートでの調停対象
DARAM	チャンネル 2 からのライト・アクセス・リクエスト チャンネル 4 からのリード・アクセス・リクエスト
外部メモリ	チャンネル 1 からのライト・アクセス・リクエスト チャンネル 4 からのライト・アクセス・リクエスト
ペリフェラル	チャンネル 1 からのリード・アクセス・リクエスト チャンネル 2 からのリード・アクセス・リクエスト

最後に、図の各ポートについて、サービス・チェーンに接続されているのにそのポートを使用しないチャンネルが存在することに注意してください。たとえば、ペリフェラル・ポートは、チャンネル 4 によって使用されていません。チャンネル 4 がソースまたはデスティネーションとしてペリフェラル・ポートを含むように再定義された場合は、そのポートが、サービス・チェーン内での位置と優先順位に従って、チャンネル 4 を処理します。

図8. 3つのDMAポートに適用されるサービス・チェーン

サービス・チェーンの 設定	{ 高優先順位 : 0, 2, 5 低優先順位 : 1, 3, 4	ディセーブル : 0, 3, 5
		イネーブル : 1, 2, 4



5 データの単位：バイト、エレメント、フレーム、およびブロック

本書では、DMA コントローラの説明に際して、データを 4 つのレベルのまとまりにて取り扱います。

- バイト** 8 ビット値。バイトは DMA チャンネルで転送されるデータの最小単位です。
- エレメント** 1 バイト以上の転送単位。プログラムされたデータ・タイプに応じて、エレメントは 8 ビット、16 ビット、または 32 ビットの値になります。エレメント転送は割り込み不可能であり、そのすべてのバイトがポートに転送されるまで、別のチャンネルはそのポートを支配することはできません。
- フレーム** 1 つ以上のエレメントからなる転送単位。フレーム転送は、エレメント転送の間で割り込み可能です。
- ブロック** 1 つ以上のフレームからなる転送単位。各チャンネルは、1 ブロックのデータを (1 回以上) 転送できます。ブロック転送は、フレーム転送およびエレメント転送の間で割り込み可能です。

6 つの DMA チャンネルのそれぞれについて、1 ブロックあたりのフレーム数 (DMACFN)、1 フレームあたりのエレメント数 (DMACEN)、および 1 エレメントあたりのバイト数 (DMACSDP の DATATYPE ビット) を定義できます。DMACFN、DMACEN、DMACSDP、および DMA コントローラのほかのレジスタについては、40 ページの 15 節「DMA コントローラのレジスタ」を参照してください。

6 チャンネルの開始アドレス

DMA チャンネルのデータ転送では、データがリードされる先頭アドレスはソース開始アドレスと呼ばれます。データをライトする先頭アドレスはデスティネーション開始アドレスと呼ばれます。これらはバイト・アドレスです。DMA コントローラから見ると、メモリまたは I/O 空間の 8 ビットごとに固有のアドレスが割り当てられます。各チャンネルでは、以下のレジスタで、開始アドレスが指定されます。

表 4. DMA 転送の開始アドレスの定義に使用されるレジスタ

レジスタ	内容
DMACSSAL	ソース開始アドレス (下位)
DMACSSAU	ソース開始アドレス (上位)
DMACDSAL	デスティネーション開始アドレス (下位)
DMACDSAU	デスティネーション開始アドレス (上位)

以下の節では、メモリ・アクセスと I/O アクセスの開始アドレス・レジスタにロードする方法について説明します。DMA コントローラは、すべての内部メモリと外部メモリ、および I/O 空間全体 (DSP ペリフェラルのレジスタを含む) にアクセスできます。

6.1 メモリの開始アドレス

図 9. は、TMS320C55x DSP のメモリ・マップです。この図では、CPU が使用するワード・アドレス (23 ビット・アドレス) と DMA コントローラが使用するバイト・アドレス (24 ビット・アドレス) の両方が示されています。ソース / デスティネーション開始アドレス・レジスタにロードするには、以下の手順に従います。

- 1) 適切な開始アドレスを指定します。データ・タイプのアラインメントの制約をチェックします。57 ページの 15.5 節「ソースおよびデスティネーション・パラメータ・レジスタ (DMACSDP)」を参照してください。ワード・アドレスが既知の場合は、1 ビット左にシフトして、24 ビットのバイト・アドレスを生成します。たとえば、ワード・アドレスの 02 4000h は、バイト・アドレスの 04 8000h に変換されます。
- 2) バイト・アドレスの最下位 16 ビット (LSB) を DMACSSAL (ソース用) または DMACDSAL (デスティネーション用) にロードします。
- 3) バイト・アドレスの最上位 8 ビット (MSB) を DMACSSAU (ソース用) または DMACDSAU (デスティネーション用) の最下位 8 ビットにロードします。

注：

ワード・アドレス 00 0000h ~ 00 005Fh (バイト・アドレス 00 0000h ~ 00 00BFh に対応) は、DSP CPU のメモリ・マップド・レジスタ (MMR) 用に予約されています。

図9. TMS320C55x DSP のメモリ・マップ

	ワード・アドレス (16進数で表した範囲)	メモリ	バイト・アドレス (16進数で表した範囲)
メイン・データ・ページ 0	MMRs 00 0000-00 005F		00 0000-00 00BF
	00 0060-00 FFFF		00 00C0-01 FFFF
メイン・データ・ページ 1	01 0000-01 FFFF		02 0000-03 FFFF
メイン・データ・ページ 2	02 0000-02 FFFF		04 0000-05 FFFF
⋮	⋮		⋮
⋮	⋮		⋮
⋮	⋮		⋮
⋮	⋮		⋮
メイン・データ・ページ 127	7F 0000-7F FFFF		FE 0000-FF FFFF

6.2 I/O 空間の開始アドレス

図 10. は、TMS320C55x DSP の I/O マップです。この図では、CPU が使用するワード・アドレス（16 ビット・アドレス）と DMA コントローラが使用するバイト・アドレス（17 ビット・アドレス）の両方が示されています。ソース / デスティネーション開始アドレス・レジスタにロードするには、以下の手順に従います。

- 適切な開始アドレスを指定します。データ・タイプのアラインメントの制約をチェックします。57 ページの 15.5 節「ソースおよびデスティネーション・パラメータ・レジスタ (DMACSDP)」を参照してください。ワード・アドレスが既知の場合は、1 ビット左にシフトして、17 ビットのバイト・アドレスを生成します。たとえば、ワード・アドレスの 8000h は、バイト・アドレスの 1 0000h に変換されます。
- バイト・アドレスの最下位 16 ビット (LSB) を DMACSSAL (ソース用) または DMACDSAL (デスティネーション用) にロードします。
- バイト・アドレスの最上位ビット (MSB) を DMACSSAU (ソース用) または DMACDSAU (デスティネーション用) にロードします。

図 10. TMS320C55x DSP の I/O マップ

ワード・アドレス (16 進数で表した範囲)	I/O 空間	バイト・アドレス (16 進数で表した範囲)
0000-FFFF		0 0000-1 FFFF

7 チャンネルのアドレスの更新

DMA チャンネルでデータ転送を行う際、DMA コントローラは指定された開始アドレスからリード・アクセスおよびライト・アクセスを開始します（25 ページの 6 節「チャンネルの開始アドレス」を参照）。多くの場合、データ転送が始まった後で、連続する位置またはインデックスで示された位置のデータをリードおよびライトできるように、これらのアドレスを更新する必要があります。アドレス更新は、以下の 2 つのレベルで設定できます。

- **ブロック・レベルのアドレス更新。**自動初期化モード(DMACCRでAUTOINIT = 1)では、ブロック転送は、自動初期化をオフにするか、またはチャンネルをディスエーブルにするまで、連続的に実行できます。別の開始アドレスをブロック転送で使用する場合は、それらのブロック転送の間に開始アドレスを更新できます。
- **エレメント・レベルのアドレス更新。**DMA コントローラに、エレメント転送のたびにソース・アドレスまたはデスティネーション・アドレス、あるいはその両方を更新させることができます。確実にソース・アドレスが次のエレメントの先頭を指し、確実にエレメントがデスティネーションの正しい位置にライトされるようにすることができます。ソースのアドレッシング・モードは、DMACCR の SRCAMODE ビットで選択します。デスティネーションのアドレッシング・モードは、DMACCR の DSTAMODE ビットで選択します。

8 データ・バースト機能

DMA チャンネルに関連付けられたポートのどちらかまたは両方がバースト機能をサポートしている場合は、データ・バーストを使用して DMA スループットを向上させることができます。バーストがイネーブルな場合は、DMA コントローラは、チャンネルが処理されるたびに、1 つのエレメントを移動するのではなく、4 つのエレメントのバーストを実行します。DARAM ポートは、バースト機能をサポートします。EMIF ポートは、リクエストされたアドレス範囲が同期（バースト）メモリ・タイプとして設定されている場合にのみ、バースト機能をサポートします。リクエストされたアドレスが非同期メモリとして設定されている場合は、DMA は 4 回のシングル・アクセスを実行してバースト・データを移動します。ペリフェラル・ポートは、バースト機能をサポートしません。DMA は、4 回のペリフェラル・ポートへのシングル・アクセスを実行してバースト・データを移動します。

バーストが使用される場合は、ソースおよびデスティネーションの開始アドレスは、バースト境界にアラインしている必要があります。バースト境界は、最下位 4 ビットが 0h のバイト・アドレスに相当します。

バーストを使用するには、以下の条件が満たされる必要があります。

- バーストがイネーブルなポートの開始アドレスは、バースト境界上にある必要があります。
- エレメント・インデックスは 1 である必要があります。
- フレーム・インデックスによって、各バースト・アクセスはバースト境界にアラインされる必要があります。
- (エレメント数×エレメント・サイズ) の値は、バースト境界にアラインされている必要があります。これは、各フレームの最後でアドレスがバースト境界にアラインされている必要があることを意味します。

ソースとデスティネーションの両方でバーストがイネーブルであっても、ソース開始アドレスがバースト境界から始まらない場合は、ソース・バーストは内部で自動的にディスエーブルになります。ソースからチャンネル FIFO にロードし、十分なデータが集まったら、デスティネーション・バーストが実行されます。デスティネーションがバースト境界から始まらない場合は、デスティネーション・アクセスはシングル・アクセスとして実行されます。

フレーム・サイズが 4 つのエレメントの倍数でない場合は、フレームの最後に残る 1 ~ 3 つのエレメントはシングル（非バースト）アクセスとして転送されます。

9 データ・パッキング

DMA コントローラは、データ・パッキングを実行して、1 回の転送でデスティネーションまたはソースに渡すデータ量を 2 倍または 4 倍にします。たとえば、8 ビット・データ・タイプが選択されていて、デスティネーション・ポートに 32 ビット・データ・パスが存在する場合は、4 つの 8 ビット・データを、デスティネーションに送信する前に、32 ビットにパッキングすることができます。生成されるパッキング・データは、表 5. に示すように、デスティネーション・ポートまたはソース・ポートのバス・サイズに応じて異なります。

表 5. DMA コントローラによるデータ・パッキング

ポート	データ・タイプ	データ・パッキング
DARAM	8 ビット	4 つの 8 ビット・データ値を 32 ビットにパッキング
	16 ビット	2 つの 16 ビット・データ値を 32 ビットにパッキング
EMIF	8 ビット	4 つの 8 ビット・データ値を 32 ビットにパッキング
	16 ビット	2 つの 16 ビット・データ値を 32 ビットにパッキング
ペリフェラル	8 ビット	2 つの 8 ビット・データ値を 16 ビットにパッキング

デスティネーション・ポートとソース・ポートで使用されるアドレッシング・モードも、DMA コントローラがデータをパッキングするかどうかに影響します。32 ビット・データ・パッキングと 16 ビット・データ・パッキングで満たす必要がある条件を、それぞれ表 6. と表 7. に示します。

表6. 32 ビット・データ・パッキングの条件†

データ・タイプ	バイト・アドレスの 下位ビット	アドレッシング・モード	エレメント・インデックス	アクセス
8 ビット	00b	ポスト・インクリメント	-	パッキング
		シングル/ダブル・インデックス	1	パッキング
		定数	-	シングル
	11b	シングル/ダブル・インデックス	-1	パッキング
		その他	-	シングル
		-	-	シングル
16 ビット	00b	ポスト・インクリメント	-	パッキング
		シングル/ダブル・インデックス	1	パッキング
		定数	-	シングル
	10b	シングル/ダブル・インデックス	-3	パッキング
		その他	-	シングル
		-	-	シングル

† 残り転送バイト数が4バイト以上

表7. 16 ビット・データ・パッキングの条件†

データ・タイプ	バイト・アドレスの 下位ビット	アドレッシング・モード	エレメント・インデックス	アクセス
8 ビット	00b または 10b	ポスト・インクリメント	-	パッキング
		シングル/ダブル・インデックス	1	パッキング
		定数	-	シングル
	11b または 01b	シングル/ダブル・インデックス	-1	パッキング
		その他	-	シングル
		-	-	シングル

† 残り転送バイト数が2バイト以上

エレメント同期を使用する場合、ソース・ポートがイベント駆動（14 ページの 2 節「チャンネルとポート・アクセス」を参照）であれば、そのソース・ポートからのリードの際に、データ・パッキングは実行されません。ソース・ポートが非イベント駆動であれば、エレメント同期を使用しているかどうかに関係なく、ソースでパッキングがイネーブル（SRCPACK = 1）でありさえすれば、データ・パッキングは実行されます。ただし、1 回のイベントあたりでデスティネーションに転送されるエレメントは1つだけです。

10 ライト・ポスティング：内部メモリへのバッファリング・ライト

DMA コントローラは、内部メモリ・インターフェイスのライト・ポスティング機能を利用できます。ライト・ポスティング・ビットが設定されている（DMACCR で WP = 1）場合は、DMA コントローラはライトを実行した後で、そのデータが実際にメモリにライトされる前に内部メモリ・インターフェイスから受信確認を受けることができます。DMA コントローラは、内部メモリ・インターフェイスがポストされたデータを制御している間に、自由に次の動作を始めることができます。

ライト・ポスティングがディスエーブル（WP = 0）の場合は、DMA コントローラは次の動作に移行する前に、内部メモリ・インターフェイスがメモリ・アクセスを完了するのを待ちます。デバッグの際は、ライト・ポスティングをディスエーブルすると便利な場合があります。

11 チャンネルの動作とイベントの同期

チャンネルの動作は、DSP ペリフェラル内のイベントまたは外部割り込みピンのドライブによって発生したイベントに同期させることができます。DMACCR の SYNC ビットを使用して、同期イベント（存在する場合）を指定することができます。

また、各チャンネルの DMACCR に FS ビットが存在し、このビットを使用して 2 つの同期モードから選択することができます。

- エlement同期モード (FS = 0) では、1 回の Element 転送を行うために 1 つのイベントが必要です。選択された同期イベントが発生すると、1 つの Element がチャンネルの FIFO バッファからデスティネーション・ポートに転送されます。現在の Element のすべてのバイトが転送されると、チャンネルは同期イベントが次に発生するまで、デスティネーション・ポートへのリクエストを行いません。チャンネルは、後述するように、ソース・ポートからのデータをリクエストします。
- フレーム同期モード (FS = 1) では、複数の Element からなるフレーム全体をトリガするために 1 つのイベントが必要です。イベントが発生すると、複数の Element からなるフレーム全体がチャンネルの FIFO バッファからデスティネーション・ポートに転送されます。フレーム内のすべての Element が転送されると、チャンネルはイベントが次に発生するまで、デスティネーション・ポートへのリクエストを行いません。チャンネルは、後述するように、ソース・ポートからのデータをリクエストします。

同期イベントが指定され、ソース・ポートがイベント駆動（表 8. を参照）である場合は、チャンネルは、そのイベントが発生するまで、ソース・ポートへのアクセス・リクエストを行いません。イベントが発生すると、チャンネルはソース・ポートからデータをリードし、それを FIFO バッファにライトしてから、デスティネーション・ポートへのアクセス・リクエストを行います。ソース・ポートまたはデスティネーション・ポートで受信されたリクエストは、DMA サービス・チェーンのチャンネルの事前定義された位置とプログラムされた優先順位に従って処理されます（21 ページの 4 節「サービス・チェーン」を参照）。

表 8. イベント駆動ポートと非イベント駆動ポート

ポート	カテゴリ
DARAM	非イベント駆動
外部メモリ	非イベント駆動
ペリフェラル	イベント駆動

一方、使用するソース・ポートが非イベント駆動である場合は、チャンネルは、イベントを待機することなく、イネーブル（DMACCR で EN = 1）になった直後にソース・ポートにアクセス・リクエストを行います。ブロックの残りのデータについて、チャンネルは引き続きソース・リード・リクエストを行って、FIFO バッファがフルである状態を維持します。ソースからのリード・リクエストは、FIFO がフルになるか、または現在のブロックのソース・データがすべて FIFO に移動されたときのみ、停止します。チャンネルは、同時に1ブロックしかアクセスしません。イベントが発生した後、チャンネルは FIFO バッファからデスティネーション・ポートにデータを転送します。1回のイベントでデスティネーション・ポートに転送されるデータ量は、フレーム同期とエレメント同期のどちらを使用するかによって異なります。チャンネルは、チャンネル FIFO に空きができるとすぐに、ブロック全体がチャンネル FIFO に完全にコピーされるまで、ソース・ポートからデータをリクエストします。異なるポートへのリード・リクエストまたはライト・リクエストは、DMA サービス・チェーンのチャンネルの事前定義された位置とプログラムされた優先順位に従って処理されます（6ページを参照）。

チャンネルを同期しないことを選択する（SYNC = 00000b）場合は、チャンネルはイネーブルになる（DMACCR で EN = 1）と同時にアクセス・リクエストをソース・ポートに送信します。EN = 1にセットすると、チャンネルに定義されているブロック全体の転送が開始されます。

11.1 同期ステータスのチェック

各チャンネルには、そのステータス・レジスタの DMACSR に同期フラグ（SYNC）があります。同期イベントが発生すると、DMA コントローラはこのフラグをセットします（SYNC = 1）。フラグは、以下のようにクリア（SYNC = 0）されます。

- ソース・ポートがイベント駆動である場合は、ソース・ポートが DMA チャンネルからのリクエストの処理を開始するときに、SYNC ビットがクリアされます。
- ソース・ポートが非イベント駆動である場合は、デスティネーション・ポートが DMA チャンネルからのリクエストの処理を開始するときに、SYNC ビットがクリアされます。

11.2 失われた同期イベント

同期イベントが発生したときに、DMA コントローラがその前に発生した同期イベントの処理を完了していない（DMA コントローラが DMACSR の SYNC ビットをクリアしていない）場合は、その同期イベントは失われます。DMA コントローラは、イベントのドロップ（喪失）に対して、以下の方法で対応します。

- 現在のエレメント転送が完了すると、チャンネルの動作は停止します。
- 対応する割り込みイネーブル・ビットがセットされている（DMACICRのDROPIE = 1）場合は、DMA コントローラはイベント・ドロップ・ステータス・ビットもセットし（DMACSR で DROP = 1）、割り込みリクエストを CPU に送信します。詳細については、34 ページの12節「チャンネルの動作の監視」を参照してください。

次の DMA 転送を開始する前に、CPU は、DMACCR で EN = 0 にしてエラー条件をクリアする必要があります。

12 チャンネルの動作の監視

DMA コントローラは、以下の表にリストした動作イベントに対応して、CPU に割り込みを送信できます。各チャンネルについて、割り込み制御レジスタ (DMACICR) の割り込みイネーブル (IE) ビットとステータス・レジスタ (DMACSR) の対応するいくつかのステータス・ビットが存在します (DMACICR と DMACSR については 51 ページの 15.4 節「割り込み制御レジスタ (DMACICR) とステータス・レジスタ (DMACSR)」を参照)。表にある動作イベントの 1 つが発生した場合は、DMA コントローラは対応する IE ビットをチェックし、結果に応じて動作します。

- IE ビットが 1 (割り込みがイネーブル) の場合は、DMA コントローラは対応するステータス・ビットをセットし、関連付けられた割り込みリクエストを CPU に送信します。プログラムがレジスタをリードした場合は、DMACSR は自動的にクリアされます。
- IE ビットが 0 の場合は、DMA コントローラは対応するステータス・ビットをセットしますが、割り込みを CPU に送信することはありません。

また、DMACSR には SYNC ビットがあり、チャンネルの同期イベントを選択した場合に使用されます。SYNC は選択された同期イベントが発生した時 (SYNC = 1)、およびそれが処理された時 (SYNC = 0) を示します。同期イベントの詳細については、32 ページの 11 節「チャンネルの動作とイベントの同期」を参照してください。

表 9. DMA コントローラの動作イベントおよび関連ビットと割り込み

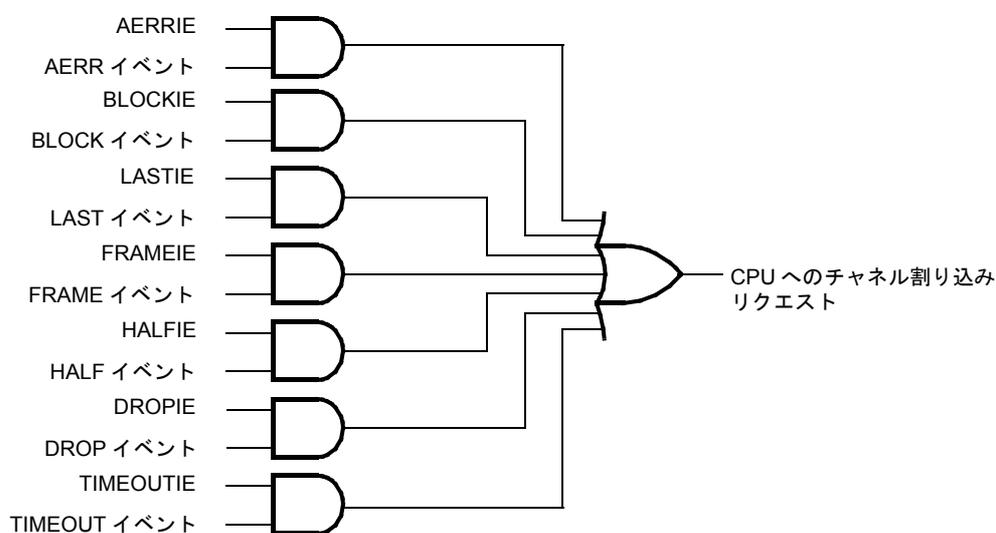
動作イベント	割り込みイネーブル・ビット	ステータス・ビット	関連付けられた割り込み
アドレス・エラーの発生	AERRIE	AERR	チャンネル割り込み
ブロック転送の完了	BLOCKIE	BLOCK	チャンネル割り込み
最後のフレーム転送の開始	LASTIE	LAST	チャンネル割り込み
フレーム転送の完了	FRAMEIE	FRAME	チャンネル割り込み
現在のフレームの前半分を転送済み [†]	HALFIE	HALF	チャンネル割り込み
同期イベントの喪失	DROPIE	DROP	チャンネル割り込み
タイムアウト・エラーの発生	TIMEOUTIE	TIMEOUT	チャンネル割り込み

[†] 奇数個の要素を含むフレームの場合は、転送済み要素数が転送待ち要素数を上回ったときにハーフ・フレーム・イベントが発生します。たとえば、5 つの要素からなるフレームの場合は、DMA コントローラが 3 つの要素を転送した時点でハーフ・フレーム・イベントが発生します。

12.1 チャンネル割り込み

6つのチャンネルにはそれぞれ固有の割り込みがあります。図11.に示すように、チャンネル割り込みは、すべてのイネーブルな動作イベントの論理和です。これらのイベントは、チャンネルの割り込み制御レジスタ (DMACICR) の該当する割り込みイネーブル (IE) ビットをセットまたはクリアすることによって、任意の組み合わせを選択できます。チャンネルのステータス・レジスタ (DMACSR) のビットをリードすることによって、どのイベントが割り込みを発生させたのかを判断できます。DMACSR をリードすると、すべてのステータス・ビットがクリアされます。割り込みが発生するたびに、保留中のステータス・ビットをクリアするために、DMACSR をリードする必要があります。

図11. チャンネル割り込みリクエストのトリガ



割り込みイネーブル・ビットの使用例としては、チャンネル 1 の動作を監視しており、DMACICR が以下のように設定されているものとします。

```
AERRIE = 0
BLOCKIE = 0
LASTIE = 0
FRAMEIE = 1
HALFIE = 0
DROPIE = 1
TIMEOUTIE = 0
```

現在のフレーム転送が完了したとき、または同期イベントが失われた場合（33 ページの 11.2 節「失われた同期イベント」を参照）に、チャンネル 1 の割り込みリクエストが CPU に送信されます。ほかのイベントは、チャンネル 1 割り込みを生成することはできません。1 つ以上のイベントが割り込みをトリガしたかどうかを確認するために、DMACSR の DROP ビットと FRAME ビットをリードすることができます。

チャンネル 1 割り込みは、CPU の割り込みフラグ・レジスタの対応するフラグ・ビットをセットします。CPU は、割り込みに対応するか、または割り込みを無視します。

DMACICR と DMACSR の詳細については、51 ページの 15.4 節「割り込み制御レジスタ (DMACICR) とステータス・レジスタ (DMACSR)」を参照してください。

12.2 タイム・アウト・エラー条件

DMA コントローラには、4 つの DMA ポート（内部メモリ・ポート 0、内部メモリ・ポート 1、外部メモリ・ポート、およびペリフェラル・ポート）にそれぞれタイム・アウト・カウンタが存在します。DMA コントローラを制御するクロックは、TMS320VC5501/5502 DSP 内にプログラムされている高速ペリフェラル・クロック (SYSCLK1) に基づいて動作します。DMA 転送がいずれかのポートでリクエストされると、対応するタイム・アウト・カウンタが、SYSCLK1 のサイクルごとに増加します。SYSCLK1 の 512 サイクル以内に転送が完了しなかった場合は、タイム・アウト・エラー信号が生成されます。4 つの DMA ポートのタイム・アウト・カウンタはすべてデフォルトでディスエーブルですが、DMAGTCR を使用して個別にイネーブルにすることができます。

タイム・アウト・エラー信号が発生すると、対応する DMA チャンネルの動作は停止します。対応する割り込みイネーブル・ビットがセットされている (DMACICR で TIMEOUTIE = 1) 場合は、DMA コントローラはタイム・アウト・ステータス・ビットもセットし (DMACSR で TIMEOUT = 1)、チャンネル割り込みリクエストを CPU に送信します。CPU は、割り込みリクエストに対応するか、または割り込みリクエストを無視できます。

次の DMA 転送を開始する前に、CPU は、DMACCR で EN = 0 にしてタイム・アウト・エラー条件をクリアする必要があります。

12.3 アドレス・エラー条件

DMA コントローラが、DSP の I/O 空間の予約アドレスにアクセスした場合は、DMA コントローラでアドレス・エラー信号が生成されます。その場合は、対応する DMA チャンネルの動作は停止します。対応する割り込みイネーブル・ビットがセットされている (DMACICR で AERRIE = 1) 場合は、DMA コントローラはアドレス・エラー・ステータス・ビットもセットし (DMACSR で AERR = 1)、チャンネル割り込みリクエストを CPU に送信します。CPU は、割り込みリクエストに対応するか、または割り込みリクエストを無視できます。

次の DMA 転送を開始する前に、CPU は、DMACCR で EN = 0 にしてアドレス・エラー条件をクリアする必要があります。

12.4 バス・エラー割り込み

CPU が以下のようなアクションを実行すると、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。CPU は、割り込みリクエストに対応するか、または割り込みリクエストを無視できます。

- CPU が DMA レジスタ・マップの予約アドレスにアクセスを試みた場合。
- CPU がレジスタに、またはレジスタ内のフィールドに、不正な値 / 予約された値のライトを試みた場合。重要な例を以下に示します。
 - CPU が、アラインされていないアドレス (選択されたデータ・タイプに従って適切にアラインされていないアドレス) をアドレス・レジスタにロードしようと試みた場合。
 - CPU が、アラインされていないアドレスを作成するようなインデックスをインデックス・レジスタにロードしようと試みた場合。

13 DMA 転送の遅れ

チャンネルの各エレメント転送では、1回のリード・アクセス（ソース・ロケーションからチャンネル・バッファへの転送）と1回のライト・アクセス（チャンネル・バッファからデスティネーション・ロケーションへの転送）が行われます。この動作の完了に要する時間は、以下の要因に依存します。

- 選択された高速ペリフェラル・クロック（SYSCLK1）信号周波数。この信号は、DMA コントローラに反映され、すべての DMA 転送のタイミングを決定します。
- インターフェイスによって追加される、またはインターフェイスから生じるウェイト・ステート、またはその他のサイクル。
- ほかのチャンネルの動作。チャンネルは順番に処理されるので、ほかのチャンネルに保留中の DMA サービス・リクエストの数が、所定のチャンネルを処理できる頻度に影響します。チャンネルの処理方法の詳細については、21 ページの 4 節「サービス・チェーン」を参照してください。
- 同期イベントのタイミング（チャンネルが同期している場合）。DMA コントローラは、同期イベントが発生するまで、同期しているチャンネルを処理できません。同期の詳細については、32 ページの 11 節「チャンネルの動作とイベントの同期」を参照してください。

最小遅延（ベストケース）は、使用するポートによって決まります。DARAM ポートでは、DMA が同一メモリ・ブロックへのアクセスで CPU と競合しない場合は、1 サイクルあたり 1 回のアクセスを実行できます。DARAM ポートを使用しているチャンネルのベストケースの転送レートは、ソースをリードする 1 サイクルとデスティネーションにライトする 1 サイクルです。EMIF ポートの最小遅延は、使用するメモリ・タイプ、プログラムされているタイミング、およびメモリ自身に起因するあらゆる遅延（ARDY ピンの制御など）を含む EMIF 設定によって決まります。ペリフェラル・ポートの遅延は、アクセスされるペリフェラル、およびそのペリフェラルに供給される高速および低速のペリフェラル・クロックの設定によって決まります。13 ページの 1.3 節「内部メモリへの DMA リクエストと CPU リクエスト」で説明されているように、CPU は常に DMA コントローラよりも高い優先順位で、内部メモリの同一 DARAM ブロックにアクセスできます。

14 電力、エミュレーション、リセットについての考慮事項

以下の節では、DMA コントローラをローパワー状態に移行する方法、デバッガ・ブレークポイントに対するDMAコントローラの対応をプログラムする方法、およびDSPのリセット後のDMAコントローラ・レジスタの値について説明します。

14.1 DMA コントローラの消費電力の削減（アイドル設定）

DSP は、アイドルまたはアクティブにプログラム可能なアイドル・ドメインに分割されます。すべてのドメインの状態をアイドル設定と呼びます。クロック・ジェネレータ・ドメインまたは DMA ドメイン、あるいはその両方をディスエーブルにするアイドル設定では、DMA クロックが停止し、その結果 DMA コントローラの動作が停止します。チャンネル同期（存在する場合）のタイプによって、DMA コントローラが停止するまでの時間が決まります。

- 同期なし (DMACCR で SYNC = 00000b)。DMA コントローラは、ブロック転送全体が完了した後に停止します。
- フレーム同期 (DMACCR で SYNC は 0 以外、FS = 1)。DMA コントローラは、現在のフレーム転送が完了した後に停止します。
- エレメント同期 (DMACCR で SYNC は 0 以外、FS = 0)。DMA コントローラは、現在のエレメント転送が完了した後に停止します。

DMA ドメインがアイドルのときは、アイドル設定を変更せずに一時的にアクティブにすることができるケースが 1 つあります。マルチチャンネル・バッファード・シリアル・ポート (McBSP) のいずれかでデータ転送するための DMA コントローラが必要になった場合は、DMA コントローラはアイドル状態から脱してデータ転送を実行し、その後再びアイドル状態に入ります。

14.2 DMA コントローラのエミュレーション・モード

DMAGCR の FREE ビットは、エミュレーション・ブレークポイントが検出されたときの DMA コントローラの対応を制御します。FREE = 0 (リセット値) の場合は、ブレークポイントは DMA 転送を一時停止します。FREE = 1 の場合は、DMA 転送がブレークポイントによって割り込まれることはありません。

14.3 DSP リセット後の DMA コントローラ

DSP リセットは、DMA コントローラと DMA コンフィギュレーション・レジスタをリセットします。以下に示すレジスタ定義は、レジスタの内容に対する DSP リセットの影響について説明します。

15 DMA コントローラのレジスタ

表 10 に、ダイレクト・メモリ・アクセス (DMA) コントローラのレジスタのタイプを列挙します。グローバル制御レジスタ (DMAGCR) とグローバル・タイム・アウト制御レジスタ (DMAGTCR) の 2 つのレジスタは、すべてのチャンネルに影響します。さらに、各 DMA チャンネルについて、16 個のチャンネル・コンフィギュレーション・レジスタがあります。各レジスタの I/O アドレスについては、使用している TMS320C55x DSP のデータ・マニュアルを参照してください。

CPU が DMA レジスタ・マップの予約アドレスへのアクセスを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。CPU は、割り込みリクエストに対応するか、または割り込みリクエストを無視できます。

表 10. DMA コントローラのレジスタ

レジスタ	説明	詳細説明ページ
DMAGCR	グローバル制御レジスタ (1 つのみ)	41 ページ
DMAGTCR	グローバル・タイム・アウト制御レジスタ (1 つのみ)	42 ページ
DMACCR	チャンネル制御レジスタ (各チャンネルに 1 つ)	44 ページ
DMACICR	割り込み制御レジスタ (各チャンネルに 1 つ)	51 ページ
DMACSR	ステータス・レジスタ (各チャンネルに 1 つ)	51 ページ
DMACSDP	ソースおよびデスティネーション・パラメータ・レジスタ (各チャンネルに 1 つ)	57 ページ
DMACSSAL	ソース開始アドレス (下位) レジスタ (各チャンネルに 1 つ)	62 ページ
DMACSSAU	ソース開始アドレス (上位) レジスタ (各チャンネルに 1 つ)	62 ページ
DMACDSAL	デスティネーション開始アドレス (下位) レジスタ (各チャンネルに 1 つ)	63 ページ
DMACDSAU	デスティネーション開始アドレス (上位) レジスタ (各チャンネルに 1 つ)	63 ページ
DMACEN	エレメント数レジスタ (各チャンネルに 1 つ)	64 ページ
DMACFN	フレーム数レジスタ (各チャンネルに 1 つ)	64 ページ

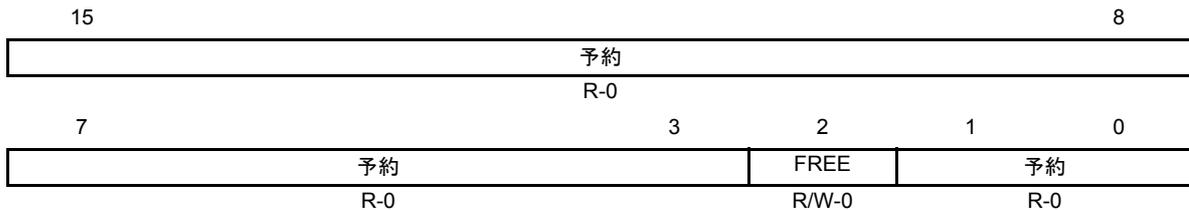
表 10. 「DMA コントローラのレジスタ」(続き)

レジスタ	説明	詳細説明ページ
DMACSEI	ソース・エレメント・インデックス・レジスタ (各チャンネルに1つ)	66 ページ
DMACSFI	ソース・フレーム・インデックス・レジスタ (各チャンネルに1つ)	66 ページ
DMACDEI	デスティネーション・エレメント・インデックス・レジスタ (各チャンネルに1つ)	66 ページ
DMACDFI	デスティネーション・フレーム・インデックス・レジスタ (各チャンネルに1つ)	66 ページ
DMACSAC	ソース・アドレス・カウンタ・レジスタ (各チャンネルに1つ)	69 ページ
DMACDAC	デスティネーション・アドレス・カウンタ・レジスタ (各チャンネルに1つ)	69 ページ

15.1 グローバル制御レジスタ (DMAGCR)

グローバル制御レジスタ (図 12. を参照) は、16 ビット I/O マップド・レジスタで、DMA コントローラのエミュレーション・モードを設定するために使用されます。

図 12. グローバル制御レジスタ (DMAGCR)



凡例: R = リード、W = ライト、-n = DSP リセット後の値

表 11. DMAGCR のビット・フィールドの説明

ビット	フィールド	値	説明
15-3	予約		これらのビットへのライトは何の影響もありません。これらのビットのリードは 0 を返します。
2	FREE		エミュレーション・モード・ビット。FREE は、エミュレーション・ブレークポイントが検出されたときの DMA コントローラの動作を制御します。
		0	ブレークポイントで DMA 転送を一時停止
		1	ブレークポイントが検出されても DMA 転送は中断せずに続行
1-0	予約		これらのビットへのライトは何の影響もありません。これらのビットのリードは 0 を返します。

15.2 グローバル・タイム・アウト制御レジスタ (DMAGTCR)

グローバル・タイム・アウト制御レジスタは 16 ビット・リード / ライト・レジスタで、DMA ポートのタイム・アウト・カウンタをイネーブルまたはディスエーブルにするために使用されます。タイム・アウト・カウンタがディスエーブルな場合は、DMA コントローラはそれらのポートでタイム・アウト・エラー条件を生成しません。タイム・アウト・エラー条件の詳細については、36 ページの 12.2 節「タイム・アウト・エラー条件」を参照してください。

図 13. グローバル・タイム・アウト制御レジスタ (DMAGTCR)



凡例： R = リード、W = ライト、-n = DSP リセット後の値

表 12. DMAGTCR のビット・フィールドの説明

ビット	フィールド	値	説明
15-4	予約		これらのビットへのライトは何の影響もありません。これらのビットのリードは 0 を返します。
3	PTE		ペリフェラル・ポート・タイム・アウト・カウンタ・イネーブル・ビット。このビットは、ペリフェラル・ポートへの DMA リクエストの遅れを監視するために使用するタイム・アウト・カウンタをイネーブル/ディスエーブルします。
		0b	タイム・アウト・カウンタはディスエーブル
		1b	タイム・アウト・カウンタはイネーブル
2	ETE		外部メモリ・ポート・タイム・アウト・カウンタ・イネーブル・ビット。このビットは、外部メモリ・ポートへの DMA リクエストの遅れを監視するために使用するタイム・アウト・カウンタをイネーブル/ディスエーブルします。
		0b	タイム・アウト・カウンタはディスエーブル
		1b	タイム・アウト・カウンタはイネーブル
1	ITE1		内部メモリ・ポート 1 タイム・アウト・カウンタ・イネーブル・ビット。このビットは、内部メモリ・ポート 1 経由で DARAM への DMA リクエストの遅れを監視するために使用するタイム・アウト・カウンタをイネーブル/ディスエーブルします。
		0b	タイム・アウト・カウンタはディスエーブル
		1b	タイム・アウト・カウンタはイネーブル
0	ITE0		内部メモリ・ポート 0 タイム・アウト・カウンタ・イネーブル・ビット。このビットは、内部メモリ・ポート 0 経由で DARAM への DMA リクエストの遅れを監視するために使用するタイム・アウト・カウンタをイネーブル/ディスエーブルします。
		0b	タイム・アウト・カウンタはディスエーブル
		1b	タイム・アウト・カウンタはイネーブル

15.3 チャンネル制御レジスタ (DMACCR)

各チャンネルには、以下の図に示す形式のチャンネル制御レジスタがあります。この I/O マップド・レジスタを使用すると、以下の操作を実行できます。

- ソース・アドレスとデスティネーション・アドレスの更新方法を選択 (SRCAMODE と DSTAMODE)
- DMA 転送のリピートをイネーブルにし、制御する (AUTOINIT、REPEAT、および ENDPROG)
- 内部メモリへのアクセスのライト・ポスティングをイネーブルまたはディスエーブルにする (WP)
- チャンネルをイネーブルまたはディスエーブルにする (EN)
- チャンネルの低い優先順位または高い優先順位を選択 (PRIO)
- エレメント同期またはフレーム同期を選択 (FS)
- チャンネルで転送を開始する同期イベント (存在する場合) を決定 (SYNC)

図 14. チャンネル制御レジスタ (DMACCR)

15	14	13	12	11	10	9	8
DSTAMODE		SRCAMODE		ENDPROG	WP	REPEAT	AUTOINIT
R/W-00		R/W-00		R/W-0	R/W-0	R/W-0	R/W-0
7	6	5	4				0
EN	PRIO	FS	SYNC				
R/W-0	R/W-0	R/W-0	R/W-0 0000				

凡例: R = リード、W = ライト、-n = DSP リセット後の値

表 13. DMACCR のビット・フィールドの説明

ビット	フィールド	値	説明
15-14	DSTAMODE		デスティネーション・アドレッシング・モード。DSTAMODE は、DMA コントローラがチャネルのデスティネーション・ポートにライトするとき使用するアドレッシング・モードを決定します。
		00b	定数アドレス 同じアドレスが各エレメント転送で使用されます。
		01b	自動ポスト・インクリメント エレメント転送が完了するたびに、アドレスは選択されたデータ・タイプに従ってインクリメントされます。 データ・タイプが 8 ビットの場合は アドレス = アドレス + 1 データ・タイプが 16 ビットの場合は アドレス = アドレス + 2 データ・タイプが 32 ビットの場合は アドレス = アドレス + 4
		10b	シングル・インデックス エレメント転送が完了するたびに、アドレスはプログラムされたインデックス量だけインクリメントされます。 アドレス = アドレス + エレメント・インデックス
		11b	ダブル・インデックス (ソート) エレメント転送が完了するたびに、アドレスは適切なインデックス量だけインクリメントされます。 現在のフレームに転送されていないエレメントが残っている場合は アドレス = アドレス + エレメント・インデックス フレーム内の最終エレメントが転送された場合は アドレス = アドレス + フレーム・インデックス

表 13. 「DMACCR のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
13-12	SRCAMODE		ソース・アドレッシング・モード SRCAMODE は、DMA コントローラがチャネルのソース・ポートからリードするときに使用するアドレッシング・モードを決定します。
		00b	定数アドレス 同じアドレスが各エレメント転送で使用されます。
		01b	自動ポスト・インクリメント エレメント転送が完了するたびに、アドレスは選択されたデータ・タイプに従ってインクリメントされます。 データ・タイプが 8 ビットの場合は アドレス = アドレス + 1 データ・タイプが 16 ビットの場合は アドレス = アドレス + 2 データ・タイプが 32 ビットの場合は アドレス = アドレス + 4
		10b	シングル・インデックス エレメント転送が完了するたびに、アドレスはプログラムされたインデックス量だけインクリメントされます。 アドレス = アドレス + エレメント・インデックス
		11b	ダブル・インデックス (ソート) エレメント転送が完了するたびに、アドレスは適切なインデックス量だけインクリメントされます。 現在のフレームに転送されていないエレメントが残っている場合は アドレス = アドレス + エレメント・インデックス フレーム内の最終エレメントが転送された場合は アドレス = アドレス + フレーム・インデックス

表 13. 「DMACCR のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
11	ENDPROG		<p>エンド・オブ・プログラミング・ビット。各 DMA チャンルにつき、コンフィギュレーション・レジスタと作業レジスタの 2 つのレジスタ・セットが存在します。自動初期化 (AUTOINIT = 1) が原因でブロック転送が繰り返し発生する場合は、現在のブロック転送中にコンフィギュレーション・レジスタにライトすることによって、次の DMA 転送のコンテキストを変更できます。現在の転送の最後にコンフィギュレーション・レジスタの内容が作業レジスタにコピーされ、DMA コントローラは新しいコンテキストを使用して次の転送を開始します。自動初期化を正しく実行するには、DMA コントローラがコンフィギュレーション・レジスタの内容をコピーする前に、CPU がコンフィギュレーション・レジスタのプログラミングを完了する必要があります。</p> <p>DMA コントローラは、コンフィギュレーション・レジスタを作業レジスタにコピーした後で、自動的に ENDPROG ビットをクリアします。この後、CPU は、コンフィギュレーション・レジスタをプログラムすることによって、次の転送で使用する DMA チャンル・コンテキストをプログラムできるようになります。</p> <p>CPU の動作を待つで自動初期化が行われるようにするには、以下のシーケンスに従います。</p> <ol style="list-style-type: none"> 1) REPEAT ビットをクリアして (REPEAT = 0)、自動初期化が ENDPROG = 1 になるまで待機するように設定します。 2) ENDPROG = 0 をポーリングします。ENDPROG = 0 は、DMA コントローラが前のコンテキストのコピーを完了したことを示します。この時点で、コンフィギュレーション・レジスタは、次の転送のためにプログラムできるようになります。 3) コンフィギュレーション・レジスタをプログラムします。 4) ENDPROG をセットして (ENDPROG = 1)、レジスタのプログラミングの完了を示します。
		0	コンフィギュレーション・レジスタはプログラミング可能 / プログラミング中
		1	プログラミング完了
10	WP		<p>ライト・ポストイング・ビット。このビットは、31 ページで説明されているように、ライト・ポストイング機能をイネーブルまたはディスエーブルにします。</p>
		0	ライト・ポストイングはディスエーブル
		1	<p>ライト・ポストイングはイネーブル</p> <p>ライトを開始した後、DMA コントローラは、データが実際にメモリにライトされる前に、内部メモリ・インターフェイスから確認を受信することができます。</p>

表 13. 「DMACCR のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
9	REPEAT		リピート条件ビット。自動初期化が選択されている (AUTOINIT = 1) チャネルでは、REPEAT が 2 つの特殊なりピート条件のうちのどちらかを指定します。
		0	ENDPROG = 1 の場合のみリピート 現在の DMA 転送が完了した後は、自動初期化はエンド・オブ・プログラミング・ビット (ENDPROG) がセットされた後にのみ実行されます。
		1	ENDPROG に関係なくリピート 現在の DMA 転送が完了したら、自動初期化は ENDPROG が 0 または 1 のどちらであるかに関係なく実行されます。
8	AUTOINIT		ビットの自動初期化。DMA コントローラは自動初期化をサポートします。自動初期化では、DMA ブロック転送間でチャンネルの自動再初期化が行われます。AUTOINIT を使用して、この機能をイネーブルまたはディスエーブルにします。
		0	自動初期化はディスエーブル チャンネルの動作は、現在のブロック転送が完了すると停止します。転送を即時停止するには、チャンネル・イネーブル・ビット (EN) をクリアします。
		1	自動初期化はイネーブル 現在のブロック転送が完了すると、DMA コントローラはチャンネルを再初期化し、新しいブロック転送を開始します。チャンネルの動作を停止するには、2 つの選択肢があります。 <input type="checkbox"/> 転送を即時停止するには、チャンネル・イネーブル・ビットをクリア (EN = 0) します。 <input type="checkbox"/> 現在のブロック転送の完了後に動作を停止するには、AUTOINIT をクリアします (AUTOINIT = 0)。
7	EN		チャンネル・イネーブル・ビット。EN を使用して、チャンネルでの転送をイネーブルまたはディスエーブルにします。DMA コントローラは、チャンネルでのブロック転送が完了した後に、EN をクリアします。 注: DMA コントローラが EN をクリアするとき同時に CPU が EN にライトを試みた場合は、DMA コントローラに高い優先順位が与えられます。EN がクリアされ、CPU からの値は捨てられます。
		0	チャンネルはディスエーブル チャンネルは DMA コントローラによって処理できません。DMA 転送がチャンネルで既にアクティブな場合は、DMA コントローラはその転送を停止して、チャンネルをリセットします。
		1	チャンネルはイネーブル チャンネルは、次に使用可能なタイム・スロットで、DMA コントローラによって処理できます。

表 13. 「DMACCR のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
6	PRIO	0	低優先順位
		1	高優先順位
5	FS	0	フレーム/エレメント同期 選択された同期イベントが発生すると、1つのエレメントがチャンネルで転送されます。エレメント転送のたびに同期イベントを待機します。
		1	フレーム同期 選択された同期イベントが発生すると、1つのフレーム全体がチャンネルで転送されます。フレーム転送のたびに同期イベントを待機します。
4-0	SYNC	表 14 を参照	同期コントロール・ビット。DMACCR の SYNC は、DSP のどのイベント (たとえばタイマ・カウントダウン) がチャンネルで DMA 転送を開始するかを決定します。複数のチャンネルで同じ SYNC 値を使用することができます。別の言い方をすると、1つの同期イベントで複数のチャンネルの動作を開始できます。 DSP リセットによって SYNC = 00000b が選択されます (同期イベントなし)。SYNC = 00000b の場合は、DMA コントローラは同期イベントを待機せずにチャンネルで DMA 転送を開始します。チャンネルの動作はチャンネルがイネーブルになる (EN = 1) と同時に開始します。 CPU が SYNC ビットに予約値のライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。

表 14. TMS320VC5501/5502 DSP の同期イベント・マッピング

DMACCR の SYNC フィールド		チャンネルの同期イベント
00000b		同期イベントなし
00001b		McBSP 0 受信イベント
00010b		McBSP 0 送信イベント
00011b		予約 (この値を使用しないでください)
00100b		予約 (この値を使用しないでください)
00101b		McBSP 1 受信イベント
00110b		McBSP 1 送信イベント
00111b		予約 (この値を使用しないでください)
01000b		予約 (この値を使用しないでください)
01001b		予約 /McBSP イベント シリアル・ポート・モード [†] = 0 : 予約 シリアル・ポート・モード = 1 : McBSP 2 受信イベント TMS320VC5501 では使用不可
01010b		予約 /McBSP イベント シリアル・ポート・モード [†] = 0 : 予約 シリアル・ポート・モード = 1 : McBSP 2 送信イベント TMS320VC5501 では使用不可
01011b		予約 /UART イベント シリアル・ポート・モード [†] = 0 : UART 受信イベント シリアル・ポート・モード = 1 : 予約
01100b		予約 /UART イベント シリアル・ポート・モード [†] = 0 : UART 送信イベント シリアル・ポート・モード = 1 : 予約
01101b		タイマ 0 割り込みイベント
01110b		タイマ 1 割り込みイベント
01111b		外部割り込み 0
10000b		外部割り込み 1
10001b		外部割り込み 2

[†]シリアル・ポート・モード・ビットの詳細については、デバイス別のデータ・マニュアルを参照してください。

表 14. 「TMS320VC5501/5502 DSP の同期イベント・マッピング」 (続き)

DMACCR の SYNC フィールド	
チャンネルの同期イベント	
10010b	外部割り込み 3
10011b	I2C モジュール受信イベント
10100b	I2C モジュール送信イベント
その他の値	予約 (この値を使用しないでください)

† シリアル・ポート・モード・ビットの詳細については、デバイス別のデータ・マニュアルを参照してください。

15.4 割り込み制御レジスタ (DMACICR) とステータス・レジスタ (DMACSR)

各チャンネルには、割り込み制御レジスタ (DMACICR) とステータス・レジスタ (DMACSR) があります。DMACICR と DMACSR は、I/O マップド・レジスタです。それらのビットを図 15. で示し、表 15. と表 16. で説明します。

DMACICR を使用して、DMA コントローラの 1 つ以上の動作イベントによる割り込みトリガを指定します。動作イベントが発生し、その割り込みイネーブル (IE) ビットが 1 の場合は、割り込みリクエストが DSP CPU に送信され、そこで処理または無視されます。各チャンネルには、CPU に対する固有の割り込みライン、および CPU 内にフラグとイネーブル・ビットのセットが 1 つ存在します。

どの動作イベントが発生したかを調べるには、プログラムで DMACSR をリードします。DMA コントローラは、動作イベントが発生したときに、割り込みフラグ・ビット (ビット 7 および 5-0) のいずれかを設定します。割り込みフラグ・ビットは、プログラムが DMACSR をリードしてそのすべてのビットが自動的にクリアされるまで、セットされたままです。

AERR、DROP、および TIMEOUT の各ビットは、エラー条件を示します。エラー条件が発生した場合は、次の DMA 転送が開始される前にクリアする必要があります。エラー条件をクリアするには、CPU が DMACCR の EN ビットに 0 をライトする必要があります。

SYNC ビット (ビット 6) を使用して、同期イベントが発生したこと (SYNC = 1)、およびそこで生成されたアクセス要求が処理されたこと (SYNC = 0) を検出することができます。

図 15. 割り込み制御レジスタ (DMACICR) とステータス・レジスタ (DMACSR)

DMACICR							
15				9			8
予約							予約 [†]
R-0							R/W-1
7	6	5	4	3	2	1	0
AERRIE	予約	BLOCKIE	LASTIE	FRAMEIE	HALFIE	DROPIE	TIMEOUTIE
R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1
DMACSR							
15				9			8
予約							予約 [‡]
R-0							R-x
7	6	5	4	3	2	1	0
AERR	SYNC	BLOCK	LAST	FRAME	HALF	DROP	TIMEOUT
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

凡例: R = リード、W = ライト、-n = DSP リセット後の値、-x = リセット後の値は未定義

[†] DMACICR のビット 8 に常に 0 をライトします。リセット後はこのビットを 1 から 0 に変更します。

[‡] DMACSR のビット 8 のリード状態は定義されません。

表 15. DMACICR のビット・フィールドの説明

ビット	フィールド	値	説明
15-9	予約		これらのビットへのライトは何の影響もありません。これらのビットのリードは 0 を返します。
8	予約		このビットへは常に 0 をライトします。リセット後はこのビットを 1 から 0 に変更します。
7	AERRIE		アドレス・エラー割り込みイネーブル・ビット。AERRIE は、チャンネルのソース・ポートまたはデスティネーション・ポートでアドレス・エラーが発生したときの DMA コントローラの対応を決定します。アドレス・エラー条件については、36 ページの 12.3 節「アドレス・エラー条件」を参照してください。
		0	このエラーが発生したときは CPU にチャンネル割り込みリクエストを送信しません。
		1	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信します。

表 15. 「DMACICR のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
6	予約		このビットへのライトは何の影響もありません。このビットのリードは 0 を返します。
5	BLOCKIE	0	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信しません。
		1	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信します。
4	LASTIE	0	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信しません。
		1	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信します。
3	FRAMEIE	0	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信しません。
		1	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信します。

表 15. 「DMACICR のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
2	HALFIE		ハーフ・フレーム割り込みイネーブル・ビット。HALFIE は、現在のフレームの前半分がデスティネーション・ポートに転送されたときの DMA コントローラの対応を決定します。奇数個の要素を含むフレームの場合は、転送済み要素数が転送待ち要素数を上回ったときにハーフ・フレーム・イベントが発生します。たとえば、5 つの要素からなるフレームの場合は、DMA コントローラが 3 つの要素を転送した時点でハーフ・フレーム・イベントが発生します。
		0	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信しません。
		1	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信します。
1	DROPIE		同期イベント喪失割り込みイネーブル・ビット。DMA コントローラがその前の DMA リクエストの処理を完了する前に、再び DMA 同期イベントが発生した場合は、同期イベント喪失エラーが発生します。DROPIE は、チャンネルで同期イベントの喪失が発生したときの DMA コントローラの対応を決定します。
		0	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信しません。
		1	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信します。
0	TIMEOUTIE		タイム・アウト割り込みイネーブル・ビット。TIMEOUTIE は、チャンネルのソース・ポートまたはデスティネーション・ポートでタイム・アウト・エラーが発生したときの DMA コントローラの対応を決定します。タイム・アウト・エラー条件については、36 ページの 12.2 節「タイム・アウト・エラー条件」を参照してください。
		0	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信しません。
		1	このエラーが発生したときに CPU にチャンネル割り込みリクエストを送信します。

表 16. DMACSR のビット・フィールドの説明

ビット	フィールド	値	説明
15-9	予約		予約 これらのビットへのライトは何の影響もありません。これらのビットのリードは 0 を返します。
8	予約		ビット 8 のリード状態は定義されていません。
7	AERR		アドレス・エラー・ステータス・ビット。DMA コントローラは、チャンネルのソース・ポートまたはデスティネーション・ポートでアドレス・エラーが発生した場合に、AERR を設定します。アドレス・エラー条件については、36 ページの 12.3 節「アドレス・エラー条件」を参照してください。
		0	アドレス・エラーが発生していないか、または AERR がクリアされています。
		1	アドレス・エラーの発生。チャンネル割り込みリクエストが CPU に送信されています。
6	SYNC		同期イベント・ステータス・ビット。DMA コントローラは、SYNC を更新して、チャンネルの同期イベントが発生したこと、および同期しているチャンネルが処理されたことを示します。
		0	DMA コントローラは直前のアクセス・リクエストの処理を完了していません。
		1	同期イベントの発生。SYNC ビットは、33 ページの 11.1 節「同期ステータスのチェック」で説明されているように、DMA によって自動的にクリアされます。 注 1: DMA コントローラがその前の DMA リクエストの処理を完了する前に、再び同期イベントが発生した場合は、同期イベント喪失エラーが発生します。DROPIE ビットと DROP ビットを使用して、このタイプのエラーを追跡できます。 注 2: チャンネルの同期イベントを選択するには、DMACCR の SYNC ビットを使用します。SYNC ビットは、DMACCR の SYNC ビットが 00000b にセットされている場合に、0 にセットされます。
5	BLOCK		ホール・ブロック・ステータス・ビット。DMA コントローラは、現在のブロックがすべてデスティネーション・ポートに転送されたときに、BLOCK をセットします。
		0	ホール・ブロック・イベントがまだ発生していないか、または BLOCK がクリアされています。
		1	ホール・ブロックが転送済み。チャンネル割り込みリクエストが CPU に送信されています。

表 16. 「DMACSR のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
4	LAST		最終フレーム・ステータス・ビット。DMA コントローラは、最終フレーム・イベントが発生したときに、LAST をセットします。ペリフェラル・ポートがソースの場合は、最終フレーム・イベントは、最終フレームの先頭要素がソースからリードされたときに、発生します。外部メモリ・ポートまたは内部メモリ・ポートがソースの場合は、最終フレーム・イベントは、最終フレームの先頭要素がデスティネーションで受信されたときに、発生します。
		0	最終フレーム・イベントがまだ発生していないか、または LAST がクリアされています。
		1	DMA コントローラは最終フレームの転送を開始済み。チャンネル割り込みリクエストが CPU に送信されています。
3	FRAME		ホール・フレーム・ステータス・ビット。DMA コントローラは、現在のフレームがすべてデスティネーション・ポートに転送されたときに、FRAME をセットします。
		0	ホール・フレーム・イベントがまだ発生していないか、または FRAME がクリアされています。
		1	ホール・フレームが転送済み。チャンネル割り込みリクエストが CPU に送信されています。
2	HALF		ハーフ・フレーム・ステータス・ビット。DMA コントローラは、現在のフレームの前半分がデスティネーション・ポートに転送されたときに、HALF をセットします。奇数個の要素を含むフレームの場合は、転送済み要素数が転送待ち要素数を上回ったときにハーフ・フレーム・イベントが発生します。たとえば、5つの要素からなるフレームの場合は、DMA コントローラが3つの要素を転送した時点でハーフ・フレーム・イベントが発生します。
		0	ハーフ・フレーム・イベントがまだ発生していないか、または HALF がクリアされています。
		1	フレームの前半分を転送済み。チャンネル割り込みリクエストが CPU に送信されています。

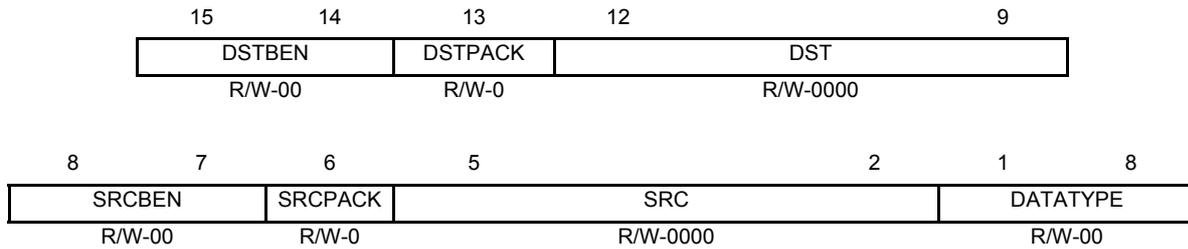
表 16. 「DMACSR のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
1	DROP		同期イベント喪失ステータス・ビット。DMA コントローラがその前の DMA リクエストの処理を完了する前に、再び DMA 同期イベントが発生した場合は、同期イベント喪失エラーが発生します。DMA コントローラは、DMACICR で DROPIE = 1 であり、そのチャンネルで同期イベント喪失が発生した場合にのみ、DROP を設定します。
		0	同期イベント喪失が発生していないか、または DROP がクリアされています。
		1	同期イベント喪失の発生。チャンネル割り込みリクエストが CPU に送信されています。 注 1: DROP ビットは、DMACSR がリードされた後にクリアされます。 注 2: DROP ビットは、DMACCR の SYNC ビットが 00000b にセットされている場合に、0 にセットされます。
0	TIMEOUT		タイム・アウト・ステータス・ビット。DMA コントローラは、DMACICR で TIMEOUTIE = 1 であり、チャンネルのソース・ポートまたはデスティネーション・ポートでタイム・アウト・エラーが発生した場合にのみ、TIMEOUT をセットします。タイム・アウト・エラー条件については、36 ページの 12.2 節「タイム・アウト・エラー条件」を参照してください。
		0	タイム・アウト・エラーが発生していないか、または TIMEOUT がクリアされています。
		1	タイム・アウト・エラーの発生。チャンネル割り込みリクエストが CPU に送信されています。

15.5 ソースおよびデスティネーション・パラメータ・レジスタ (DMACSDP)

各チャンネルには、図 16. に示す形式のソースおよびデスティネーション・パラメータ・レジスタがあります。この I/O マップド・レジスタを使用して、ソース・ポート (SRC) とデスティネーション・ポート (DST) の選択、ポート・アクセスのデータ・タイプ (DATATYPE) の指定、データ・パッキング (SRCPACK と DSTPACK) のイネーブルまたはディスエーブル、およびバースト転送 (SRCBEN と DSTBEN) のイネーブルまたはディスエーブルを実行できます。

図 16. ソースおよびデスティネーション・パラメータ・レジスタ (DMACSDP)



凡例: R = リード、W = ライト、-n = リセット後の値

表 17. DMACSDP のビット・フィールドの説明

ビット	フィールド	値	説明
15-14	DSTBEN		デスティネーション・バースト・イネーブル・ビット。DMA コントローラのバーストとは、DMA ポートでの連続した 4 つの 32 ビット・アクセスを指します。DSTBEN は、DMA コントローラがチャンネルのデスティネーション・ポートでバーストを実行するかどうかを決定します。
		00b	デスティネーションでのバーストはディスエーブル (シングル・アクセスがイネーブル)
		01b	デスティネーションでのバーストはディスエーブル (シングル・アクセスがイネーブル)
		10b	デスティネーションでのバーストはイネーブル。デスティネーションにライトするときは、DMA コントローラは 4 つ連続した 32 ビット・アクセスを実行します。
		11b	予約。CPU が DSTBEN ビットに 11b のライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。
13	DSTPACK		デスティネーション・パッキング・イネーブル・ビット。DMA コントローラは、データ・パッキングを実行して、1 回の転送でデスティネーションに渡すデータ量を 2 倍または 4 倍にします。たとえば、8 ビット・データ・タイプが選択されていて、デスティネーション・ポートに 32 ビット・データ・パスが存在する場合は、4 つの 8 ビット・データを、デスティネーションに送信する前に、32 ビットにパッキングすることができます。DSTPACK は、データ・パッキングをデスティネーション・ポートで使用するかどうかを決定します。
		0	デスティネーションでのパッキングはディスエーブル
		1	デスティネーションでのパッキングはイネーブル。可能であれば、DMA コントローラはデスティネーションにライトする前に毎回データをパッキングします。29 ページの 9 節「データ・パッキング」に、データ・パッキングが実行されている例を示します。

表 17. 「DMACSDP のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
12-9	DST		デスティネーション選択ビット。DST は、チャンネルでのデータ転送のデスティネーションにする DMA ポートを選択します。
		0000b	内部メモリ・ポート 0 経由で DARAM
		0001b	内部メモリ・ポート 1 経由で DARAM
		0010b	外部メモリ・インターフェイス (EMIF) 経由で外部メモリ
		0011b	ペリフェラル・バス・コントローラ経由でペリフェラル
		その他	予約。CPU が DST ビットに予約値のライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。
8-7	SRCBEN		ソース・バースト・イネーブル・ビット。DMA コントローラのバーストとは、DMA ポートでの連続した 4 つの 32 ビット・アクセスを指します。SRCBEN は、DMA コントローラがチャンネルのソース・ポートでバーストを実行するかどうかを決定します。 このフィールドは、以下のいずれかの条件が満たされる場合は無視され ます。 <input type="checkbox"/> ソース・ポートがバースト機能をサポートしない <input type="checkbox"/> ソース・ポートで定数アドレス・モードが選択されている <input type="checkbox"/> チャンネルがエレメント同期されている
		00b	ソースでのバーストはディスエーブル (シングル・アクセスがイネーブル)
		01b	ソースでのバーストはディスエーブル (シングル・アクセスがイネーブル)
		10b	ソースでのバーストはイネーブル。ソースからリードするときは、DMA コントローラは 4 つ連続した 32 ビット・アクセスを実行します。
		11b	予約。CPU が SRCBEN ビットに 11b のライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。
6	SRCPACK		ソース・パッキング・イネーブル・ビット。DMA コントローラは、データ・パッキングを実行して、転送する前にソースで収集するデータ量を 2 倍または 4 倍にします。たとえば、8 ビット・データ・タイプが選択されていて、ソース・ポートに 32 ビット・データ・バスが存在する場合は、4 つの 8 ビット・データを、チャンネルに送出する前に、32 ビットにパッキングすることができます。SRCPACK は、データ・パッキングをソース・ポートで使用するかどうかを決定します。
		0	ソースでのパッキングはディスエーブル
		1	ソースでのパッキングはイネーブル。可能であれば、DMA コントローラはチャンネルでデータ転送を開始する前にソースからデータをパッキングします。29 ページの 9 節「データ・パッキング」に、データ・パッキングが実行されている例を示します。

表 17. 「DMACSDP のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
5-2	SRC		ソース選択ビット。SRC は、チャンネルでのデータ転送のソースにする DMA ポートを選択します。
		0000b	内部メモリ・ポート 0 経由で DARAM
		0001b	内部メモリ・ポート 1 経由で DARAM
		0010b	外部メモリ・インターフェイス (EMIF) 経由で外部メモリ
		0011b	ペリフェラル・バス・コントローラ経由でペリフェラル
		その他	予約。CPU が SRC ビットに予約値のライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。

表 17. 「DMACSDP のビット・フィールドの説明」 (続き)

ビット	フィールド	値	説明
1-0	DATATYPE		データ・タイプ・ビット。DATATYPE は、チャンネルのソースとデスティネーションにおけるデータへのアクセス方法を示します。DMA コントローラはアクセスする際に バイト・アドレス を使用しており、データ空間またはI/O空間のバイトはそれぞれ固有のアドレスを持っていることに注意してください。エレメント転送間のアドレス更新方法については、DMACCR の SRCAMODE ビットと DSTAMODE ビットの説明を参照してください (44 ページの 15.3 節「チャンネル制御レジスタ (DMACCR)」を参照)。
		00b	8 ビット DMA コントローラは、チャンネルのソースとデスティネーションに 8 ビットでアクセスします。ソース開始アドレスとデスティネーション開始アドレスにアラインメントの制約はありません。 開始アドレス: XXXX XXXX XXXX XXXXb (X は 0 または 1 のどちらでも可) ソースまたはデスティネーションで自動ポスト・インクリメント・アドレッシング・モードを選択している場合は、エレメント転送を 1 回実行するたびに対応するアドレスに 1 が加算されます。
		01b	16 ビット DMA コントローラは、ソースとデスティネーションに 16 ビットでアクセスします。ソース開始アドレスとデスティネーション開始アドレスは偶数の 2 バイト境界上に存在する必要があります。最下位ビット (LSB) は 0 である必要があります。 開始アドレス: XXXX XXXX XXXX XXX0b (X は 0 または 1 のどちらでも可) ソースまたはデスティネーションで自動ポスト・インクリメント・アドレッシング・モードを選択している場合は、エレメント転送を 1 回実行するたびに対応するアドレスに 2 が加算されます。
		10b	32 ビット DMA コントローラは、ソースとデスティネーションに 32 ビットでアクセスします。ソース開始アドレスとデスティネーション開始アドレスは偶数の 4 バイト境界上に存在する必要があります。最下位の 2 ビットは 0 である必要があります。 開始アドレス: XXXX XXXX XXXX XX00b (X は 0 または 1 のどちらでも可) ソースまたはデスティネーションで自動ポスト・インクリメント・アドレッシング・モードを選択している場合は、エレメント転送を 1 回実行するたびに対応するアドレスに 4 が加算されます。
		11b	予約。CPU が DATATYPE ビットに 11b のライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。

15.6 ソース開始アドレス・レジスタ (DMACSSAL と DMACSSAU)

各チャンネルには2つのソース開始アドレス・レジスタがあります。このレジスタを図17. で示し、表 18. と表 19. で説明します。チャンネルのソース・ポートに最初にアクセスするとき、DMA コントローラは2つの I/O マップド・レジスタの内容を合わせてバイト・アドレスを生成します。DMACSSAU は上位ビットとして、DMACSSAL は下位ビットとして、それぞれ使用されます。

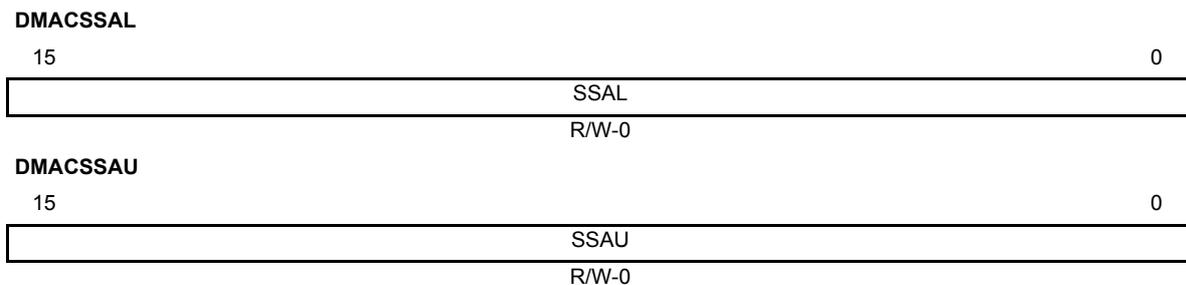
ソース開始アドレス = DMACSSAU:DMACSSAL

注：

- 1) ソース開始アドレス・レジスタにはバイト・アドレスをロードする必要があります。ワード・アドレスが既知の場合は、レジスタにロードする前に1ビット左にシフトします。
- 2) 16ビットまたは32ビットのデータ・タイプを使用している場合は、開始アドレスを適切にアラインする必要があります。DMACSDP の DATATYPE ビットの説明を参照してください (57 ページ)。CPU がアラインされていないアドレスのライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。
- 3) プログラマは、開始アドレス、エレメント・インデックス、およびフレーム・インデックスからポートの範囲内の有効なアドレスが生成されることを保証する必要があります。無効なアドレスが生成される場合は、タイム・アウト・エラーが発生します。

デスティネーション開始アドレスは DMACDSAL と DMACDSAU から生成されます。63 ページの 15.7 節「デスティネーション開始アドレス・レジスタ (DMACDSAL と DMACDSAU)」を参照してください。

図 17. ソース開始アドレス・レジスタ (DMACSSAL と DMACSSAU)



凡例： R = リード、W = ライト、-n = リセット後の値

表 18. DMACSSAL のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	SSAL	0000h-FFFFh	ソース開始アドレス (バイト・アドレス) の下位部分

表 19. DMACSSAU のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	SSAU	0000h-00FFh	ソース開始アドレス (バイト・アドレス) の上位部分
		0100h-FFFFh	予約 (この値を使用しないでください)

15.7 デスティネーション開始アドレス・レジスタ (DMACDSAL と DMACDSAU)

各チャンネルには 2 つのデスティネーション開始アドレス・レジスタがあります。このレジスタを図 18. で示し、表 20. と表 21. で説明します。チャンネルのデスティネーション・ポートに最初にアクセスするとき、DMA コントローラは 2 つの I/O マップド・レジスタの内容を合わせてバイト・アドレスを生成します。DMACDSAU は上位ビットとして、DMACDSAL は下位ビットとして、それぞれ使用されます。

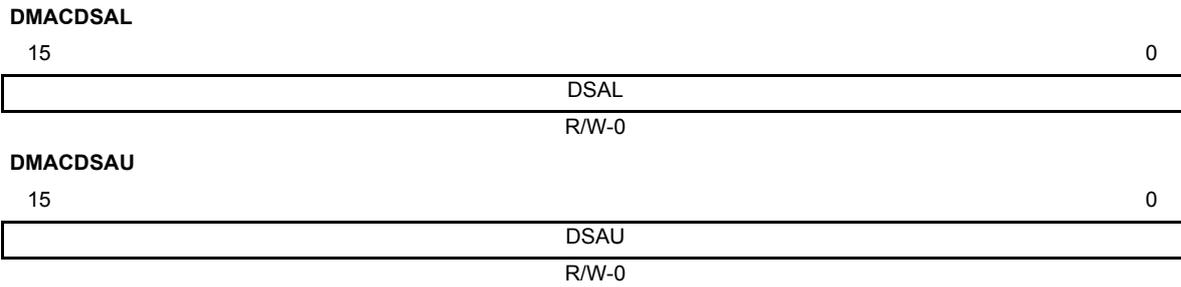
デスティネーション開始アドレス = DMACDSAU:DMACDSAL

注:

- 1) デスティネーション開始アドレス・レジスタにはバイト・アドレスをロードする必要があります。ワード・アドレスが既知の場合は、レジスタにロードする前に 1 ビット左にシフトします。
- 2) 16 ビットまたは 32 ビットのデータ・タイプを使用している場合は、開始アドレスを適切にアラインする必要があります。DMACSDP の DATATYPE ビットの説明を参照してください (57 ページ)。CPU がアラインされていないアドレスのライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。
- 3) プログラマは、開始アドレス、エレメント・インデックス、およびフレーム・インデックスからポートの範囲内の有効なアドレスが生成されることを保証する必要があります。無効なアドレスが生成される場合は、タイム・アウト・エラーが発生します。

ソース開始アドレスは DMACSSAL と DMACSSAU から生成されます。62 ページの 15.6 節「ソース開始アドレス・レジスタ (DMACSSAL と DMACSSAU)」を参照してください。

図 18. デスティネーション開始アドレス・レジスタ (DMACDSAL と DMACDSAU)



凡例： R = リード、W = ライト、-n = リセット後の値

表 20. DMACDSAL のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	DSAL	0000h-FFFFh	デスティネーション開始アドレス (バイト・アドレス) の下位部分

表 21. DMACDSAU のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	DSAU	0000h-00FFh	デスティネーション開始アドレス (バイト・アドレス) の上位部分

15.8 エレメント数レジスタ (DMACEN) とフレーム数レジスタ (DMACFN)

各チャンネルには、エレメント数レジスタとフレーム数レジスタがあります (図 19.、表 22.、および表 23. を参照)。DMACFN には、各ブロックに含まれるフレーム数をロードします。DMACEN には、各フレームに含めるエレメント数をロードします。それぞれ少なくとも 1 フレームおよび 1 エレメントを含める必要があります。最大ではそれぞれ 65535 個まで含めることができます。

$$1 \leq \text{フレーム数} \leq 65535$$

$$1 \leq \text{エレメント数} \leq 65535$$

CPU が DMACEN または DMACFN に 0 のライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。

図 19. エレメント数レジスタ (DMACEN) とフレーム数レジスタ (DMACFN)

DMACEN	
15	0
ELEMENTNUM	
R/W-0001h	
DMACFN	
15	0
FRAMENUM	
R/W-0001h	

凡例: R = リード、W = ライト、-n = リセット後の値

表 22. DMACEN のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	ELEMENTNUM	0000h	予約。CPU がこのフィールドに 0000h のライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。
		0001h-FFFFh	フレームあたりのエレメント数 (1 ~ 65535)

表 23. DMACFN のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	FRAMENUM	0000h	予約。CPU がこのフィールドに 0000h のライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。
		0001h-FFFFh	ブロックあたりのフレーム数 (1 ~ 65535)

15.9 エlement・インデックス・レジスタ (DMACSEI、DMACDEI) とフレーム・インデックス・レジスタ (DMACFI、DMACDFI)

DMACCR の SRCAMODE ビットと DSTAMODE ビットを使用すると、ソース・ポートとデスティネーション・ポートで、シングル・インデックス・アドレッシング・モードまたはダブル・インデックス・アドレッシング・モードをそれぞれ独立して選択することができます (44 ページを参照)。これらのインデックス・アドレッシング・モードをサポートするために、各チャンネルに2つのElement・インデックス・レジスタと2つのフレーム・インデックス・レジスタが存在します。これらの4つのレジスタを図 20. に示し、それに続く表で説明します。

DMA コントローラは、以下のインデックス・レジスタを使用して、ソース・ポートを制御します。

- DMACSEI: シングル・インデックス・アドレッシング・モードまたはダブル・インデックス・アドレッシング・モードのソースで使用するElement・インデックスが含まれます。
- DMACSEFI: ダブル・インデックス・アドレッシング・モードのソースで使用するフレーム・インデックスが含まれます。

DMA コントローラは、以下のインデックス・レジスタを使用して、デスティネーション・ポートを制御します。

- DMACDEI: シングル・インデックス・アドレッシング・モードまたはダブル・インデックス・アドレッシング・モードのデスティネーションで使用するElement・インデックスが含まれます。
- DMACDFI: ダブル・インデックス・アドレッシング・モードのデスティネーションで使用するフレーム・インデックスが含まれます。

Element・インデックスとフレーム・インデックスは 16 ビット符号付き数値であり、以下の範囲を示すことができます。

-32768 バイト = フレーム・インデックス = 32767 バイト
-32768 バイト = Element・インデックス = 32767 バイト

1 回の転送が完了すると、ソース・アドレス・レジスタとデスティネーション・アドレス・レジスタには、転送されたElementの最終バイトのアドレスが格納されます。たとえば、DMA チャンネルがバイト・アドレス 0x2000 の 32 ビット・Elementをリードしている場合を考えます。Elementがリードされた後のソース・アドレスは 0x2003 になりますが、これは DMA チャンネルが合計 4 バイトをリードするためです。DMA チャンネルが 16 ビット・Elementをリードする場合は、Elementがリードされた後のソース・アドレスは 0x2001 になりますが、これは 2 バイトだけリードされるためです。1 バイトがリードされる場合は、そのバイトがリードされた後のソース・アドレスは 0x2000 のままです。

シングル・インデックス・モードが使用されているときは、1 回のElement転送が完了した時点でソース・アドレスまたはデスティネーション・アドレスにElement・インデックスが加算されます。変更後のアドレスは、次のElement転送が始まる時に使用されます。

ソース・アドレスまたはデスティネーション・アドレスでダブル・インデックス・モードが使用されているときは、前述されているように、1回のエレメント転送が完了した時点でソース・アドレスまたはデスティネーション・アドレスにエレメント・インデックスが加算されます。ただし、フレーム内の最終エレメントの場合は例外です。フレーム内の最終エレメントの場合は、ソース・アドレスまたはデスティネーション・アドレスに、エレメント・インデックスではなく、フレーム・インデックスが加算されます。たとえば、フレーム内の最終エレメントがバイト・アドレス 0x801E で始まり、データ・タイプが 16 ビット、フレーム・インデックスが 0x0003 に設定されている場合は、DMA は先頭バイト (0x801E) を移動した後にエレメントの次のバイト (0x801F) を移動します。0x801F にフレーム・インデックスが加算されて、次に移動するエレメントの先頭バイトのアドレスが算出されます (0x801F + 0x0003 = 0x8022)。

ソース・アドレスまたはデスティネーション・アドレスに加算されるエレメント・インデックスは、DMACSDP の DATATYPE フィールドで選択されたデータ・タイプに従って、アラインされたアドレスを生成する必要があります。そのため、特定の値だけがエレメント・インデックスとして有効になります。

エレメント・インデックスとして有効な値を以下に示します。

- データ・タイプが 32 ビットの場合は $[4 \times N] + 1$ (ここで $N = -2, -1, 0, 1, 2, \dots$)
- データ・タイプが 16 ビットの場合は $[2 \times N] + 1$ (ここで $N = -2, -1, 0, 1, 2, \dots$)
- データ・タイプが 8 ビットの場合は任意の値

エレメント・インデックスと同じように、フレーム・インデックスは、DMACSDP の DATATYPE フィールドで選択されたデータ・タイプに従って、アラインされたアドレスを生成する必要があります。

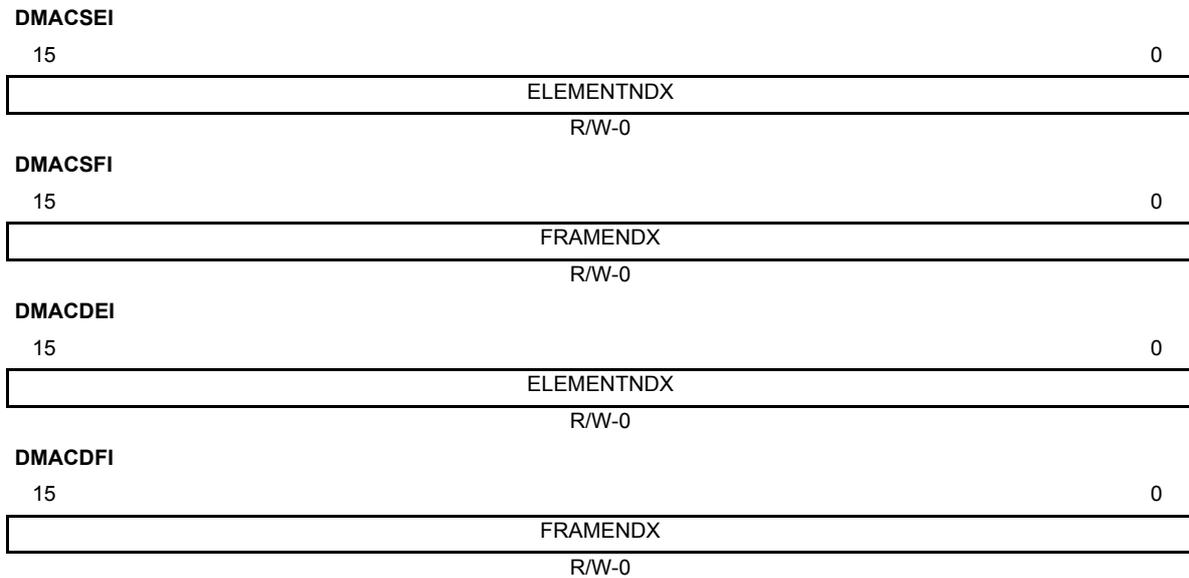
フレーム・インデックスとして有効な値を以下に示します。

- データ・タイプが 32 ビットの場合は $[4 \times N] + 1$ (ここで $N = -2, -1, 0, 1, 2, \dots$)
- データ・タイプが 16 ビットの場合は $[2 \times N] + 1$ (ここで $N = -2, -1, 0, 1, 2, \dots$)
- データ・タイプが 8 ビットの場合は任意の値

プログラマは、開始アドレス、エレメント・インデックス、およびフレーム・インデックスからポートの範囲内の有効なアドレスが生成されることを保証する必要があります。無効なアドレスが生成される場合は、タイム・アウト・エラーが発生します。

CPU がアラインされていないアドレスを生成するエレメント・インデックスまたはフレーム・インデックスのライトを試みた場合は、DMA コントローラは、バス・エラー割り込み (BERRINT) リクエストを CPU に送信します。これは、アドレッシング・インデックスが使用されない場合にも発生します。

図 20. エlement・インデックス・レジスタ (DMACSEI、DMACDEI) とフレーム・インデックス・レジスタ (DMACSF1、DMACDFI)



凡例: R = リード、W = ライト、-n = リセット後の値

表 24. DMACSEI のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	ELEMENTNDX	-32768 ~ 32767	ソース・エレメント・インデックス (バイト数)

表 25. DMACSF1 のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	FRAMENDX	-32768 ~ 32767	ソース・フレーム・インデックス (バイト数)

表 26. DMACDEI のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	ELEMENTNDX	-32768 ~ 32767	デスティネーション・エレメント・インデックス (バイト数)

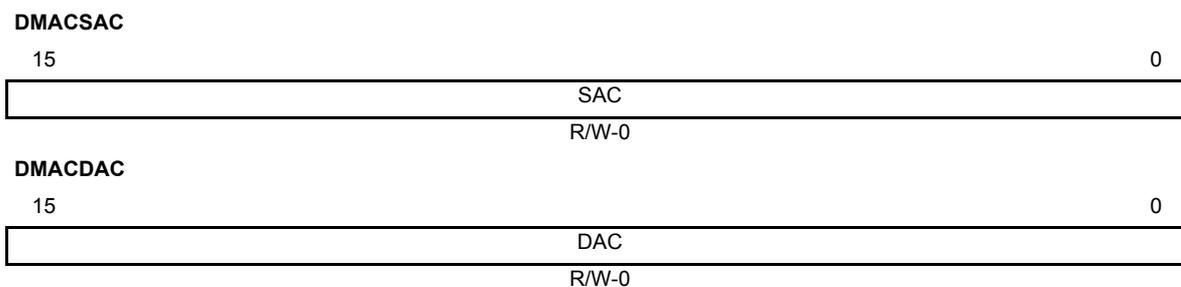
表 27. DMACDFI のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	FRAMENDX	-32768 ~ 32767	デスティネーション・フレーム・インデックス (バイト数)

15.10 ソース・アドレス・カウンタ (DMACCSAC) とデスティネーション・アドレス・カウンタ (DMACDAC)

各 DMA チャンネルの進行状況は、ソースおよびデスティネーション・アドレス・カウンタ (DMACCSAC および DMACDAC) をリードすることによって監視できます。DMACCSAC は現在のソース・アドレスの下位 16 ビットを示します。DMACDAC は現在のデスティネーション・アドレスの下位 16 ビットを示します。

図 21. チャンネル・ソース・アドレス・カウンタ (DMACCSAC) と
チャンネル・デスティネーション・アドレス・カウンタ (DMACDAC)



凡例: R = リード、W = ライト、-n = リセット後の値

表 28. DMACCSAC のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	SAC	0000h-FFFFh	現在のチャンネル・ソース・アドレス

表 29. DMACDAC のビット・フィールドの説明

ビット	フィールド	値	説明
15-0	DAC	0000h-FFFFh	現在のチャンネル・デスティネーション・アドレス

以下余白

改訂履歴

表 30. 文書改訂履歴

ページ	追加 / 変更 / 削除
14	14.3 節 の追加

以下余白