

# TMS320DM643x DMP DSP サブシステム リファレンス・ガイド





# 目次

<b>最初にお読みください</b> .....	<b>3</b>
<b>1 概要</b> .....	<b>9</b>
1.1 はじめに.....	10
1.2 ブロック図.....	10
1.3 TMS320DM643x DMP の DSP サブシステム .....	11
1.3.1 DSP サブシステムのコンポーネント .....	11
<b>2 TMS320C64x+ メガモジュール</b> .....	<b>13</b>
2.1 はじめに.....	14
2.2 TMS320C64x+ CPU .....	14
2.3 メモリ・コントローラ.....	16
2.3.1 L1P コントローラ .....	16
2.3.2 L1D コントローラ.....	18
2.3.3 L2 コントローラ .....	18
2.3.4 外部メモリ・コントローラ (EMC).....	19
2.3.5 内部 DMA (IDMA).....	19
2.4 内部ペリフェラル.....	20
2.4.1 割り込みコントローラ (INTC).....	20
2.4.2 パワー・ダウン・コントローラ (PDC).....	20
2.4.3 帯域マネージャ .....	21
<b>3 システム・メモリ</b> .....	<b>23</b>
3.1 メモリ・マップ.....	24
3.1.1 DSP 内部メモリ (L1P、L1D、L2).....	24
3.1.2 外部メモリ .....	24
3.1.3 内部ペリフェラル.....	24
3.1.4 デバイス・ペリフェラル.....	24
3.2 メモリ・インターフェイスの概要.....	25
3.2.1 DDR2 外部メモリ・インターフェイス .....	25
3.2.2 外部メモリ・インターフェイス.....	25
<b>4 デバイス・クロッキング</b> .....	<b>27</b>
4.1 概要.....	28
4.2 クロック・ドメイン.....	28
4.2.1 コア・ドメイン .....	28
4.2.2 コア周波数の柔軟性.....	30
4.2.3 DDR2/EMIF クロック .....	31
4.2.4 I/O ドメイン.....	32
4.2.5 ビデオ・プロセッシング・バック・エンド.....	33
<b>5 PLL コントローラ</b> .....	<b>35</b>
5.1 PLL モジュール.....	36
5.2 PLL1 による制御.....	36
5.2.1 デバイス・クロックの生成.....	37
5.2.2 PLL1 / コア・ドメイン周波数を変更する手順.....	38

5.3	PLL2 による制御.....	41
5.3.1	デバイス・クロックの生成.....	42
5.3.2	PLL2 周波数を変更するステップ.....	42
5.4	PLL コントローラのレジスタ.....	46
5.4.1	ペリフェラル ID レジスタ (PID).....	47
5.4.2	リセット・タイプ・ステータス・レジスタ (RSTYPE).....	47
5.4.3	PLL コントロール・レジスタ (PLLCTL).....	48
5.4.4	PLL マルチプライア・コントロール・レジスタ (PLLM).....	49
5.4.5	PLL コントローラ・ディバイダ 1 レジスタ (PLLDIV1).....	49
5.4.6	PLL コントローラ・ディバイダ 2 レジスタ (PLLDIV2).....	50
5.4.7	PLL コントローラ・ディバイダ 3 レジスタ (PLLDIV3).....	50
5.4.8	オシレータ・ディバイダ 1 レジスタ (OSCDIV1).....	51
5.4.9	バイパス・ディバイダ・レジスタ (BPDIV).....	52
5.4.10	PLL コントローラ・コマンド・レジスタ (PLLCMD).....	52
5.4.11	PLL コントローラ・ステータス・レジスタ (PLLSTAT).....	53
5.4.12	PLL コントローラ・クロック・アライン・コントロール・レジスタ (ALNCTL).....	54
5.4.13	PLLDIV レシオ・チェンジ・ステータス・レジスタ (DCHANGE).....	55
5.4.14	クロック・イネーブル・コントロール・レジスタ (CKEN).....	56
5.4.15	クロック・ステータス・レジスタ (CKSTAT).....	57
5.4.16	SYSClk ステータス・レジスタ (SYSTAT).....	58
<b>6</b>	<b>パワー・スリープ・コントローラ .....</b>	<b>59</b>
6.1	はじめに.....	60
6.2	パワー・ドメインおよびモジュール・トポロジ.....	61
6.3	パワー・ドメインおよびモジュール・ステート.....	62
6.3.1	パワー・ドメインのステート.....	62
6.3.2	モジュール・ステート.....	62
6.3.3	ローカル・リセット.....	63
6.4	ステート遷移の実行.....	63
6.4.1	パワー・ダウンのステート遷移.....	63
6.4.2	モジュールのステート遷移.....	63
6.5	PSC での PSC エミュレーション・サポート.....	64
6.6	PSC 割り込み.....	64
6.6.1	割り込みイベント.....	64
6.6.2	インタラプト・レジスタ.....	65
6.6.3	割り込み処理.....	66
6.7	PSC レジスタ.....	66
6.7.1	ペリフェラル・リビジョン・クラス・インフォメーション・レジスタ (PID).....	67
6.7.2	インタラプト・エバリュエーション・レジスタ (INTEVAL).....	67
6.7.3	モジュール・エラー・ペンディング・レジスタ 1 (MERRPR1).....	68
6.7.4	モジュール・エラー・クリア・レジスタ 1 (MERRCR1).....	68
6.7.5	パワー・ドメイン・トランジション・コマンド・レジスタ (PTCMD).....	69
6.7.6	パワー・ドメイン・トランジション・ステータス・レジスタ (PTSTAT).....	69
6.7.7	パワー・ドメイン・ステータス 0 レジスタ (PDSTAT0).....	70
6.7.8	パワー・ドメイン・コントロール 0 レジスタ (PDCTL0).....	71
6.7.9	モジュール・ステータス $n$ レジスタ (MDSTAT $n$ ).....	72
6.7.10	モジュール・コントロール $n$ レジスタ (MDCTL $n$ ).....	74
<b>7</b>	<b>パワー・マネージメント .....</b>	<b>75</b>
7.1	概要.....	76
7.2	PSC および PLLC の概要.....	76

7.3	クロック管理.....	77
7.3.1	モジュール・クロックのオン/オフ.....	77
7.3.2	モジュール・クロックの周波数スケーリング.....	77
7.3.3	PLL のバイパスおよびパワー・ダウン.....	77
7.4	DSP によるスリープ・モード管理.....	78
7.4.1	DSP スリープ・モード.....	78
7.4.2	DSP モジュール・クロックのオン/オフ.....	78
7.5	3.3 V 入出力 (I/O) パワー・ダウン.....	80
7.6	ビデオ DAC のパワー・ダウン.....	80
<b>8</b>	<b>割り込みコントローラ.....</b>	<b>81</b>
<b>9</b>	<b>システム・モジュール.....</b>	<b>83</b>
9.1	概要.....	84
9.2	デバイス識別.....	84
9.3	デバイス設定.....	84
9.3.1	ピン・マルチプレクス制御.....	84
9.3.2	デバイス・ブート時の設定状態.....	84
9.4	3.3 V 入出力 (I/O) パワー・ダウン.....	85
9.5	ペリフェラルの状態および制御.....	85
9.5.1	タイマ制御.....	85
9.5.2	VPSS クロックおよび DAC の制御.....	85
9.5.3	DDR2 VTP 制御.....	85
9.5.4	HPI 制御.....	85
9.6	帯域管理.....	86
9.6.1	バス・マスタ DMA のプライオリティ制御.....	86
9.6.2	EDMA 転送コントローラの設定.....	87
9.7	ブート制御.....	87
<b>10</b>	<b>リセット.....</b>	<b>89</b>
10.1	概要.....	90
10.2	リセット・ピン.....	90
10.3	リセット時に行われるデバイス設定.....	90
10.4	DSP リセット.....	91
10.4.1	DSP ローカル・リセット.....	91
10.4.2	DSP モジュール・リセット.....	91
<b>11</b>	<b>ブート・モード.....</b>	<b>93</b>
<b>A</b>	<b>改訂履歴.....</b>	<b>95</b>



# 最初にお読みください

---

---

---

## 本書について

本書では、TMS320DM643x デジタル・メディア・プロセッサ (DMP) に搭載されている DSP サブシステムについて説明します。

## 表記規則

本書では、次の表記規則を使用します。

- 16 進数は末尾に h を付けて表されています。たとえば、16 進数の 40 (10 進数 64) は、40h と表されています。
- 本書ではレジスタは図で表され、表形式で説明されます。
  - レジスタの図は、複数のフィールドで構成される長方形で示されます。各フィールドには、ビット名が付けられています。フィールドの始まりと終わりを示すビットがその上に、リード/ライト属性がその下に書かれています。凡例は、その属性を表すために使用される表記を示しています。
  - レジスタの図に示されている予約ビットは、将来的なデバイスの拡張を考慮しているビットを表しています。

## Texas Instruments 社からの関連資料

TMS320DM643x デジタル・メディア・プロセッサ (DMP) 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](http://www.ti.com) から入手可能です。[www.ti.com](http://www.ti.com) にアクセスして、検索ボックスに資料番号を入力してください。

DM643x DMP、関連ペリフェラル、およびその他の技術資料は、C6000 DSP 製品フォルダ [www.ti.com/c6000](http://www.ti.com/c6000) から入手できます。

[SPRU983](#) - TMS320DM643x DMP Peripherals Overview Reference Guide では、TMS320DM643x デジタル・メディア・プロセッサ (DMP) 上で使用可能なペリフェラルについて説明しています。

[SPRAA84](#) - TMS320C64x to TMS320C64x+ CPU Migration Guide では、Texas Instruments TMS320C64x デジタル・シグナル・プロセッサ (DSP) から TMS320C64x+ DSP への移行方法について説明しています。本書の目的は、2 つのコア間の相違点を的確に示すことです。2 つのデバイスの機能が等価な場合には、説明は省略されています。

[SPRU732](#) - TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide では、TMS320C6000 DSP ファミリーの TMS320C64x と TMS320C64x+ のデジタル・シグナル・プロセッサ (DSP) の CPU アーキテクチャ、パイプライン、命令セット、および割り込みについて説明しています。C64x/C64x+ DSP 世代は、C6000 DSP プラットフォームの固定小数点デバイスを構成しています。C64x+ DSP は C64x DSP の機能性を高め、命令セットを拡張した機能強化版です。

[SPRU871](#) - TMS320C64x+ DSP Megamodule Reference Guide では、TMS320C64x+ デジタル・シグナル・プロセッサ (DSP) のメガモジュールについて説明しています。具体的には、内部ダイレクト・メモリ・アクセス (IDMA) コントローラ、割り込みコントローラ、パワー・ダウン・コントローラ、メモリ保護、帯域管理、およびメモリとキャッシュについて説明しています。

[SPRU862](#) - TMS320C64x+ DSP Cache User's Guide では、メモリ・キャッシュの基本概念を説明し、TMS320C6000 DSP ファミリーの TMS320C64x+ デジタル・シグナル・プロセッサ (DSP) に搭載されている 2 レベル・キャッシュ・ベースの内部メモリ・アーキテクチャを DSP アプリケーションで効率的に使用方法について解説しています。外部メモリ使用時にコヒーレンスを維持する方法、DMA を使用してメモリ・レイテンシを削減する方法、およびキャッシュ効率の向上を実現するようにコードを最適化する方法について説明します。C64x+ DSP の内部メモリ・アーキテクチャは、ファー

スト・レベルでは専用プログラム・キャッシュ (L1P) および専用データ・キャッシュ (L1D) からなる 2 レベル階層で構成されます。CPU によるこれらのファースト・レベル・キャッシュへのアクセスは、CPU パイプラインをストールさせることなく完了できます。CPU から要求されたデータがキャッシュに含まれていない場合、データは次のメモリ・レベルである L2 または外部メモリからフェッチされます。



項目	ページ
1.1 はじめに .....	10
1.2 ブロック図.....	10
1.3 TMS320DM643x DMP の DSP サブシステム.....	11

## 1.1 はじめに

TMS320DM643x デジタル・メディア・プロセッサ (DMP) には、画像、ビデオ、オーディオなどの処理タスクを効率的に処理する強力なデジタル・シグナル・プロセッサ (DSP) が含まれています。DM643x DMP は、次の主要なコンポーネントおよびサブシステムから構成されています。

- C64x+ メガモジュールおよび関連付けられたメモリをはじめとする DSP サブシステム (DSPSS)
- ビデオ・プロセッシング・フロント・エンド (VPFE) サブシステム、画像入力、画像処理サブシステム、ビデオ・プロセッシング・バック・エンド (VPBE) ディスプレイ・サブシステムをはじめとするビデオ・プロセッシング・サブシステム (VPSS)
- 一連の入出力 (I/O) ペリフェラル
- 強力な DMA サブシステムおよび DDR2 メモリ・コントローラ・インターフェース

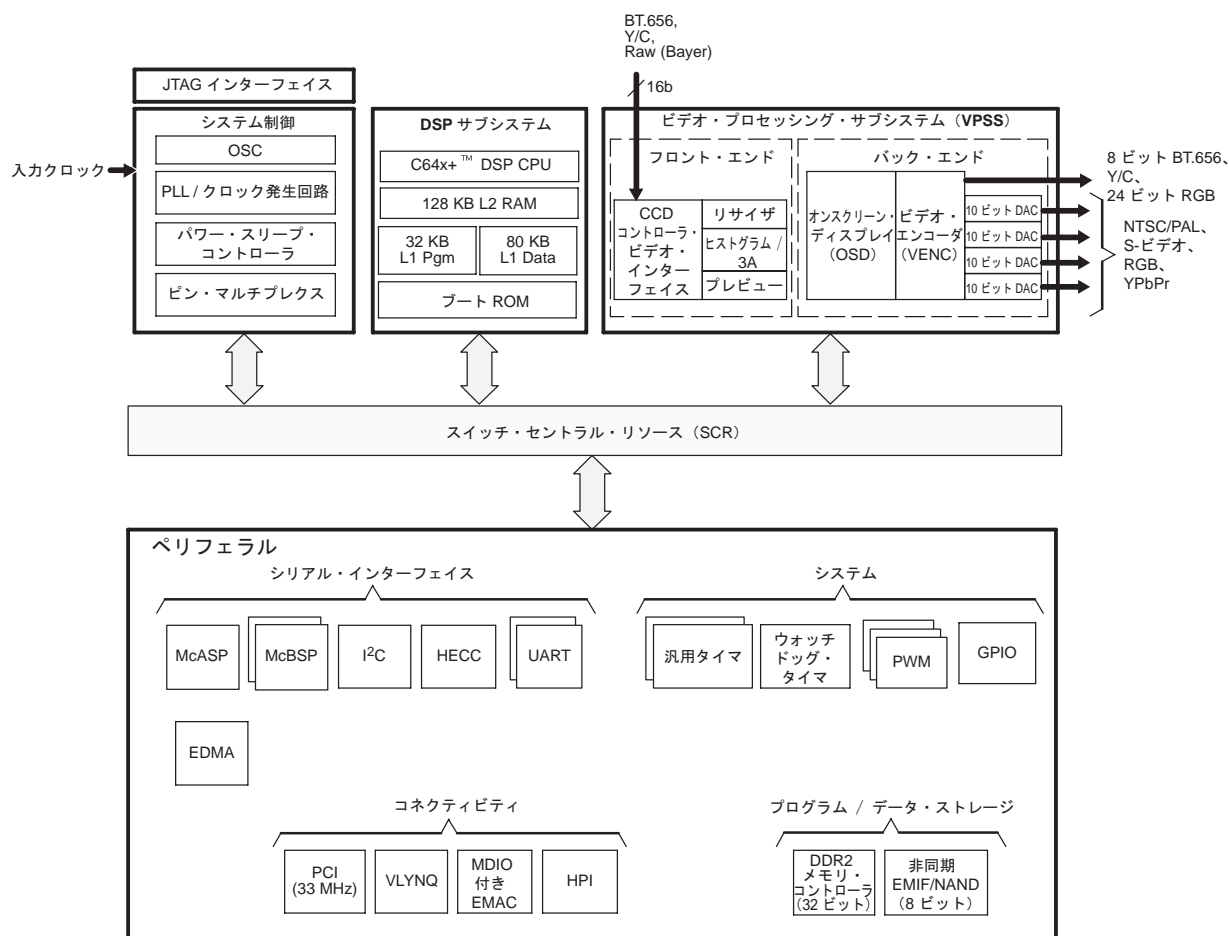
DSP サブシステムには、TI 標準の TMS320C64x+ メガモジュールおよびいくつかの内部メモリ・ブロック (L1P、L1D、L2) が組み込まれています。

詳細については、『TMS320C64x+ DSP Megamodule Peripherals Reference Guide』(資料番号 [SPRU871](#))、『TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide』(資料番号 [SPRU732](#))、および『TMS320C64x+ DSP Cache User's Guide』(資料番号 [SPRU862](#)) を参照してください。

## 1.2 ブロック図

TMS320DM643x DMP のブロック図の例を図 1-1 に示します。

図 1-1. TMS320DM643x DMP のブロック図



## 1.3 TMS320DM643x DMP の DSP サブシステム

DM643x DMP において、DSP サブシステムはデジタル・メディア・アプリケーション用のデジタル信号処理を担っています。また、DSP サブシステムは、システムレベルでの初期化、コンフィギュレーション、ユーザ・インターフェース、ユーザ・コマンド実行、接続機能、および全体のシステム制御など多くのシステム機能を処理する全体のシステム・コントローラとして機能します。

### 1.3.1 DSP サブシステムのコンポーネント

DM643x DMP の DSP サブシステムは、次のコンポーネントから構成されています。

- C64x+ メガモジュール
- DSP 内部メモリ
  - レベル 1 プログラム・メモリ (L1P)
  - レベル 1 データ・メモリ (L1D)
  - レベル 2 共有メモリ (L2)

また DSP は、デバイス上のすべてのペリフェラルを管理し、制御します。ペリフェラルの詳細なリストについては、各デバイスのデータ・マニュアルを参照してください。

図 1-1 に、DM643x DMP の機能ブロック図、および DSP サブシステムと他のモジュール間の接続関係を示します。DM643x DMP アーキテクチャは、システム・インフラストラクチャ (スイッチ・セントラル・リソース) を使用して、システム内にデータを転送します。

第 2 章では、詳細なブロック図とともに、C64x+ メガモジュールについて説明します。



## TMS320C64x+ メガモジュール

---

---

---

項目	ページ
2.1 はじめに .....	14
2.2 TMS320C64x+ CPU .....	14
2.3 メモリ・コントローラ .....	16
2.4 内部ペリフェラル .....	20

## 2.1 はじめに

C64x+ メガモジュール (図 2-1) は、次のコンポーネントから構成されています。

- TMS320C64x+ CPU
- 内部メモリ・コントローラ：
  - レベル 1 プログラム・メモリ・コントローラ (L1P コントローラ)
  - レベル 1 データ・メモリ・コントローラ (L1D コントローラ)
  - レベル 2 共有メモリ・コントローラ (L2 コントローラ)
  - 外部メモリ・コントローラ (EMC)
  - 内部ダイレクト・メモリ・アクセス (IDMA) コントローラ
- 内部ペリフェラル
  - 割り込みコントローラ (INTC)
  - パワー・ダウン・コントローラ (PDC)

## 2.2 TMS320C64x+ CPU

C64x+ メガモジュールには、C64x+ CPU が組み込まれています。C64x+ CPU は、TMS320C6000™ 世代のデバイスを構成する要素です。C6000™ デバイスは、1 サイクルで最大 8 個の 32 ビット命令を実行します。この CPU は、64 個の汎用 32 ビット・レジスタおよび 8 個の機能ユニットから構成されています。8 個の機能ユニットには、2 つの逓倍器および 6 つの算術論理演算ユニット (ALU) が含まれています。この CPU の詳細については、『TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide』(資料番号 [SPRU732](#)) を参照してください。

C6000 デバイスの機能は、次のとおりです。

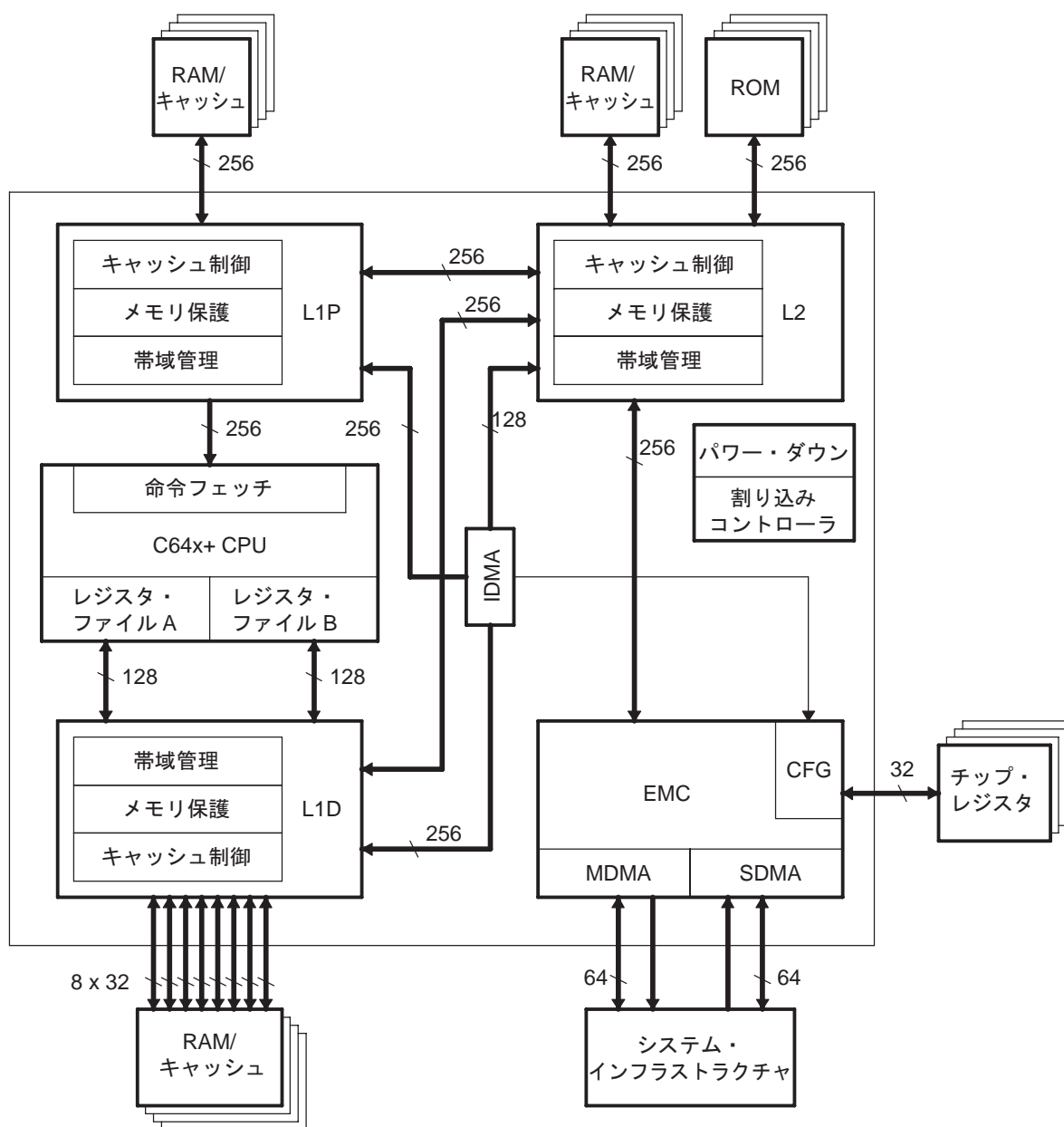
- 2 つの逓倍器および 6 個の算術論理演算ユニットをはじめとする、8 個の機能ユニットを備えた VLIW アーキテクチャを拡張した CPU
  - 通常の DSP と比較して最大で 10 倍の性能を実現する、1 サイクルで最大 8 個の命令を実行
  - 設計者の開発時間短縮を促す高効率な RISC ライクなコードを開発可能
- 命令のパッキング
  - 8 個の命令を連続または並列に実行することにおいて変わらないコード・サイズを実現
  - コード・サイズ、プログラム・フェッチ、消費電力の削減
- ほとんどの命令で条件付き実行可能
  - オーバーヘッドの大きい分岐を抑制
  - 高い処理性能を維持する並列化の向上
- 独立した機能ユニット上で効率的なコード実行
  - DSP ベンチマーク・パッケージソフトで確認された業界で最も効率的な C コンパイラ
  - 開発時間の短縮および並列度を向上させた業界初のアセンブリ・オプティマイザ
- さまざまなアプリケーションに対して効率的なメモリ・サポートを実現する、8/16/32 ビットのデータをサポート
- ボコードおよび計算集約型アプリケーションに求められる拡張精度を提供する 40 ビットの算術論理演算オプション
- 飽和および正規化による重要な算術論理演算のサポートを実現
- フィールド操作および命令 (抽出、セット、クリア、ビット・カウント) により、制御アプリケーションおよびデータ操作アプリケーションに見られる一般的な演算をサポート

C64x+ デバイスには、次の機能が追加されています。

- それぞれの逓倍器は、クロック・サイクルごとに 2 つの 16 × 16 ビット乗算または 4 つの 8 × 8 ビット乗算を実行可能
- データ・フロー・サポート機能を備えた 4 つの 8 ビット命令セット機能拡張および 2 つの 16 ビット命令セット機能拡張
- アラインされない 32 ビット (ワード) および 64 ビット (ダブル・ワード) メモリ・アクセスをサポート

- エラー訂正コードにおいて、一般的な操作に対応する特別な通信固有の命令
- ビット・カウントおよびローテート・ハードウェアによる、ビットレベルのアルゴリズムのサポートを拡張
- コンパクト命令：コード・サイズの削減を目的とする 16 ビット・バージョンの一般的な命令( AND、ADD、LD、MPY )
- プロテクト・モードでの動作：メモリ保護機能など、より高度な機能を実現するオペレーティング・システムおよびシステム機能をサポートする、特権プログラムの命令実行を使用できる 2 レベル・システム
- 堅牢なコード実行を実現するためのエラー検出機能およびプログラムのリダイレクト機能に対する例外サポート
- コード・サイズを削減するモジュール・ループ演算に対するハードウェア・サポート
- 開発時間の短縮および並列度を向上させた業界初のアセンブリ・オプティマイザ

図 2-1. TMS320C64x+ メガモジュールのブロック図



## 2.3 メモリ・コントローラ

C64x+ メガモジュールには、外部メモリ・サポート機能を備えた 2 レベル内部キャッシュ・ベースのメモリ・アーキテクチャが実装されています。レベル 1 メモリは、プログラム・メモリ (L1P メモリ) とデータ・メモリ (L1D メモリ) に分割されます。図 2-2 に、メモリ・アーキテクチャのブロック図を示します。L1P および L1D は、一部を L1 RAM (通常アクセス可能なオンチップ・メモリ) および一部を L1 キャッシュとして設定可能です。CPU はストールすることなく、L1 メモリへアクセス可能です。またレベル 2 メモリ (L2) は、L2 RAM (通常アクセス可能なオンチップ・メモリ) および外部メモリ・ロケーションをキャッシュする L2 キャッシュに分割することもできます。

RAM/キャッシュ設定およびキャッシュ・データ・パスを管理しているのは、次のコントローラです。

- L1P コントローラ
- L1D コントローラ
- L2 コントローラ
- 外部メモリ・コントローラ (EMC)

内部ダイレクト・メモリ・アクセス (IDMA) コントローラは、L1P、L1D、L2 メモリ間の DMA を管理します。

ここでは、キャッシュおよび DMA コントローラについて簡単に説明します。これらのコントローラの詳細については、『TMS320C64x+ DSP Cache User's Guide』(資料番号 [SPRU862](#)) および『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) を参照してください。

---

**注：** C64x+ メガモジュールには、メモリ・コントローラが組み込まれていますが、ただし物理的な L1P、L1D、L2 メモリは、DSP サブシステムに内蔵されていてもメガモジュールの一部ではありません。このように、C64x+ メガモジュールはさまざまなメモリ設定をサポートしているため、物理メモリの説明は別々になっています。DM643x DMP 固有の L1P、L1D、L2 メモリ設定の詳細については、3.1 節を参照してください。

---

### 2.3.1 L1P コントローラ

L1P コントローラは、C64x+ メガモジュールに搭載されているレベル 1 プログラム・メモリ (L1P メモリ) とその他のコンポーネント (C64x+ CPU、L2 コントローラ、EMC など) 間とのハードウェア・インターフェイスです。L1P コントローラは、C64x+ CPU からの命令フェッチ・リクエストに回答し、L1P メモリと L2 コントローラ間および L1P メモリと EMC 間の転送動作を管理します。

デバイスに搭載されている L1P のメモリ容量については、各デバイスのデータ・マニュアルを参照してください。L1P コントローラには、L1P RAM の一部または全てを通常の RAM またはキャッシュとして設定できるレジスタ・インターフェイスがあります。RAM のキャッシュ・サイズは、0 KB、4 KB、8 KB、16 KB、32 KB のいずれかに設定できます。

L1P は、L1P 領域 0 と L1P 領域 1 で表される 2 つの領域に分割されます。DM643x DMP の L1P アーキテクチャを次に示します。

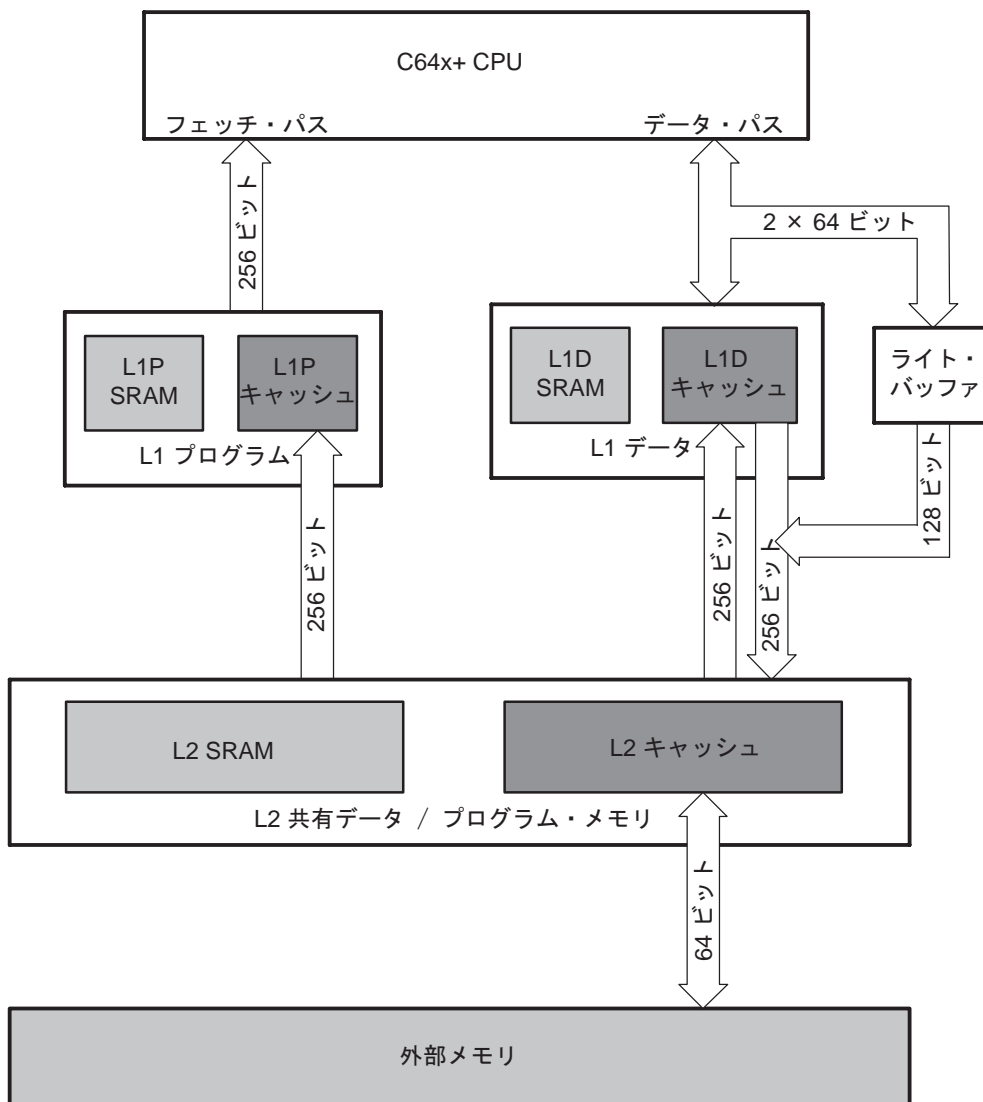
- L1P 領域 0：このメモリ領域を持ちません。
- L1P 領域 1：マップド・メモリまたはキャッシュとして設定可能なメモリ領域を持ちます。L1P 領域 1 メモリは、0 ウェイト・ステートです。この領域は、デバイスのデータ・マニュアルでは「L1P RAM/キャッシュ」として説明されます。

DM643x DMP は、標準的な C64x+ メガモジュールに搭載されている L1P メモリ保護機能をサポートしていません。

L1P コントローラの詳細およびその制御レジスタの説明については、『TMS320C64x+ DSP Cache User's Guide』(資料番号 [SPRU862](#)) および『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) の「L1P コントローラ」を参照してください。



図 2-2. C64x+ のキャッシュ・メモリ・アーキテクチャ



凡例 :

- アドレス可能なメモリ
- キャッシュ・メモリ

キャッシュ・コントローラによって管理されるデータ・パス

### 2.3.2 L1D コントローラ

L1D コントローラは、C64x+ メガモジュールに搭載されているレベル 1 データ・メモリ (L1D メモリ) とその他のコンポーネント (C64x+ CPU、L2 コントローラ、EMC など) 間とのハードウェア・インターフェイスです。L1D コントローラは、C64x+ CPU からのデータ・リクエストにตอบสนองし、L1D メモリと L2 コントローラ間および L1D メモリと EMC 間の転送動作を管理します。

デバイスに搭載されている L1D のメモリ容量については、各デバイスのデータ・マニュアルを参照してください。L1D コントローラには、L1D RAM の一部を通常のデータ RAM またはキャッシュのいずれかとして設定できるレジスタ・インターフェイスがあります。RAM のキャッシュ・サイズは、0 KB、4 KB、8 KB、16 KB、32 KB のいずれかに設定できます。

L1D は、L1D 領域 0 と L1D 領域 1 で表される 2 つの領域に分割されます。DM643x DMP の L1D アーキテクチャを次に示します。

- L1D 領域 0: 一部の DM643x デバイスでは、この領域はマップド・メモリ領域となります。このメモリ領域を持つ場合、この領域はデバイスのデータ・マニュアルでは「L1D RAM/ キャッシュ」として説明されます。
- L1D 領域 1: マップド・メモリまたはキャッシュとして設定可能なメモリ領域を持ちます。この領域は、デバイスのデータ・マニュアルでは「L1D RAM/ キャッシュ」として説明されます。

DM643x DMP は、標準的な C64x+ メガモジュールに搭載されている L1D メモリ保護機能をサポートしていません。

L1D コントローラの詳細およびその制御レジスタの説明については、『TMS320C64x+ DSP Cache User's Guide』(資料番号 [SPRU862](#)) および『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) の「L1D コントローラ」を参照してください。

### 2.3.3 L2 コントローラ

L2 コントローラは、C64x+ メガモジュールのレベル 2 メモリ (L2 メモリ) とその他のコンポーネント (L1P コントローラ、L1D コントローラ、EMC など) 間のハードウェア・インターフェイスです。L2 コントローラは、L2 メモリとその他のメモリ・コントローラ (L1P コントローラ、L1D コントローラ、EMC など) 間の転送動作を管理します。

デバイスに搭載されている L2 のメモリ容量については、各デバイスのデータ・マニュアルを参照してください。L2 コントローラには、L2 RAM の一部またはすべてを通常の RAM またはキャッシュとして設定できるレジスタ・インターフェイスがあります。RAM のキャッシュ・サイズは、0 KB、32 KB、64 KB、128 KB のいずれかに設定できます。

L2 メモリは、2 つの異なるメモリ・ポートを実装しています。DM643x DMP の L2 アーキテクチャを次に示します。

- ポート 0
  - デバイスのデータ・マニュアルでは、「L2 RAM/ キャッシュ」として説明されます。
  - バンク構造: 2 × 128 ビット・バンク
  - レイテンシ: 1 サイクル (0 ウェイト・ステート)
- ポート 1
  - デバイスのデータ・マニュアルでは、「ブート ROM」として説明されます。
  - バンク構造: 1 × 256 ビット・バンク
  - レイテンシ: 1 サイクル (0 ウェイト・ステート)

DM643x DMP は、標準的な C64x+ メガモジュールに搭載されている L2 メモリ保護機能をサポートしていません。

L2 コントローラの詳細およびその制御レジスタの説明については、『TMS320C64x+ DSP Cache User's Guide』(資料番号 [SPRU862](#)) および『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) の「L2 コントローラ」を参照してください。

### 2.3.4 外部メモリ・コントローラ (EMC)

外部メモリ・コントローラ (EMC) は、C64x+ メガモジュールに搭載されている外部メモリ・マップ (外部メモリと外部レジスタ) とその他のコントローラ (L1P コントローラ、L1D コントローラ、L2 コントローラなど) 間とのハードウェア・インターフェイスです。EMC は、外部メモリと外部レジスタ、およびその他のメモリ・コントローラ (L1P コントローラ、L1D コントローラ、L2 コントローラなど) 間の転送動作を管理します。

EMC は、標準的な C64x+ メガモジュールに搭載されているメモリ保護機能をサポートしていません。

EMC の詳細およびその制御レジスタの説明については、『TMS320C64x+ DSP Cache User's Guide』(資料番号 [SPRU862](#)) および『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) を参照してください。

### 2.3.5 内部 DMA (IDMA)

内部 DMA (IDMA) コントローラは、2 つの内部メモリ・マップド・ロケーション間での DMA 転送を容易にします。L1P、L1D、L2、および内部ペリフェラル・コンフィギュレーション・レジスタが内部のメモリ・マップド・ロケーションに該当します。

---

**注：** IDMA は、外部メモリ・マップド・ロケーション間での DMA 転送を容易にするわけではありません。EDMA を利用することで、外部 DMA 転送が容易になります。EDMA については、3.1 節および『TMS320DM643x DMP Enhanced Direct Memory Access (EDMA) Controller User's Guide』(資料番号 [SPRU987](#)) を参照してください。

---

IDMA コントローラを使用すると、他のローカル・メモリマップド RAM ヘデータ・セクションを高速に転送することができます。IDMA を使用する主な利点は、低速な L2 メモリと高速な L1D データ・メモリ間で転送できることです。これらの転送は、CPU の介在やキャッシュ・ストールが発生することなく行われます。

その他の利点として、IDMA コントローラを使用して CPU が介在することなく、内部ペリフェラル・コンフィギュレーション・レジスタをプログラムできます。

IDMA コントローラの詳細およびその制御レジスタの説明については、『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) の「内部 DMA (IDMA) コントローラ」を参照してください。

## 2.4 内部ペリフェラル

C64x+ メガモジュールには、次の内部ペリフェラルが組み込まれています。

- 割り込みコントローラ (INTC)
- パワー・ダウン・コントローラ (PDC)

ここでは、INTC および PDC について簡単に説明します。これらのペリフェラルの詳細については、『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) を参照してください。

### 2.4.1 割り込みコントローラ (INTC)

C64x+ メガモジュールには、CPU 割り込みを管理するために割り込みコントローラ (INTC) が組み込まれています。INTC は、0 ~ 127 個の DSP デバイス・イベントを 12 個の CPU 割り込みへマップします。すべての DSP デバイス・イベントのリストについては、各デバイスのデータ・マニュアルを参照してください。『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) の「割り込みコントローラ」では、INTC および DSP デバイス・イベントを 12 個の CPU 割り込みへマップする方法について詳細に説明しています。

### 2.4.2 パワー・ダウン・コントローラ (PDC)

C64x+ メガモジュールには、パワー・ダウン・コントローラ (PDC) が組み込まれています。PDC は、C64x+ メガモジュールに搭載されている次のコンポーネントをすべてパワー・ダウンすることができます。

- C64x+ CPU
- L1P コントローラ
- L1D コントローラ
- L2 コントローラ
- 拡張メモリ・コントローラ (EMC)
- 内部ダイレクト・メモリ・アクセス (IDMA) コントローラ

DM643x DMP は、DSP サブシステムに搭載されている内部メモリのパワー・ダウン機能をサポートしません。

C64x+ メガモジュールは、動的パワー・ダウンと静的パワー・ダウンの両方を行うことができますが、ただし DM643x DMP では静的パワー・ダウンのみがサポートされています。『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) では、パワー・ダウン制御について詳細に説明しています。

- 静的パワー・ダウン：PDC は、ソフトウェアからコマンドが発行されるとただちに C64x+ メガモジュール全体およびすべてのメモリのパワー・ダウンを開始します。

DM643x DMP では、静的パワー・ダウンを行うと C64x+ メガモジュールに搭載されているすべてのコンポーネントが影響を受けます。DM643x DMP は、内部メモリのパワー・ダウン機能をサポートしません。ソフトウェアで、PDC レジスタのレジスタ・ビットを使用して静的パワー・ダウンを開始することができます。PDC の詳細については、『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) を参照してください。

---

**注：** DM643x DMP は、動的パワー・ダウンをサポートしません。

---

### 2.4.3 帯域マネージャ

帯域マネージャでは、リソースを要求するリクエスト間で最適な帯域を割り当てるプログラム可能なインターフェイスを提供します。リクエストは、次のとおりです。

- EDMA によって開始される DMA 転送（およびそれに起因するコヒーレンシ動作）
- IDMA によって開始される転送（およびそれに起因するコヒーレンシ動作）
- プログラム可能なキャッシュ・コヒーレンシ動作
  - ブロック・ベースのコヒーレンシ動作
  - グローバル・コヒーレンシ動作
- CPU によって直接開始される転送
  - データ・アクセス（ロード/ストア）
  - プログラム・アクセス

対象となるリソースは、次のとおりです。

- L1P メモリ
- L1D メモリ
- L2 メモリ
- C64x+ メガモジュールの外部のリソース：外部メモリ、オンチップ・ペリフェラル、レジスタ

どのリクエストもリソースを長時間にわたりブロックできてしまうため、帯域マネージャはすべてのリクエストに対して公平にリソースを割り当てるために実装されています。

帯域マネージャは、重み付けプライオリティ制御による帯域割り当てを実装しています。それぞれのリクエスト（EDMA、IDMA、CPU など）には、転送単位ごとにプライオリティ・レベルが割り当てられています。プログラム可能なプライオリティ・レベルには、システム全体で 1 つの意味があります。合計で 9 つのプライオリティ・レベルがありますが、プライオリティ 0 が最上位で、プライオリティ 8 が最下位になります。複数のリクエストが単一リソースを求めて競合している場合、最上位のプライオリティが設定されたリクエストにアクセス権を認めることで競合状態を解決します。複数の連続したサイクルで競合が発生している場合、競合カウンタによって、 $n$  回のアービトレーション・サイクルにつき 1 回、優先順位の最も低いリクエストに対してリソースへのアクセスが保証されます。ここで、 $n$  はプログラム可能な値です。プライオリティ・レベル -1 は、競合カウンタの終了によって、優先順位が引き上げられた転送、もしくはあるリソースへの最も優先度の高い転送として固定された転送を表しています。



## システム・メモリ

項目	ページ
3.1 メモリ・マップ.....	24
3.2 メモリ・インターフェイスの概要.....	25

## 3.1 メモリ・マップ

メモリ・マップについては、各デバイスのデータ・マニュアルを参照してください。

### 3.1.1 DSP 内部メモリ (L1P、L1D、L2)

ここでは、L1P、L1D、およびL2から構成されるDM643x DMPに搭載されているDSP内部のメモリ設定について説明します。DM643x DMPにおける設定は次のとおりです。

- L1Pメモリ：L1Pコントローラは、L1P RAMの一部またはすべてを通常のプログラムRAMまたはダイレクト・マップド・キャッシュとして設定することができます。RAMのキャッシュ・サイズは、0 KB、4 KB、8 KB、16 KB、32 KBのいずれかに設定できます。
- L1Dメモリ：L1Dコントローラは、L1D RAMの一部を通常のリデータRAMまたはキャッシュのいずれかとして設定できます。RAMのキャッシュ・サイズは、0 KB、4 KB、8 KB、16 KB、32 KBのいずれかに設定できます。
- L2メモリ：L2コントローラは、L2 RAMの一部またはすべてを通常のRAMまたはキャッシュのいずれかとして設定できます。RAMのキャッシュ・サイズは、0 KB、32 KB、64 KB、128 KBのいずれかに設定できます。

RAM / キャッシュの正確な容量については、各デバイスのデータ・マニュアルを参照してください。キャッシュの設定方法については、『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) を参照してください。

### 3.1.2 外部メモリ

DSPは、次の外部メモリへアクセスできます。

- DDR2 シンクロナス DRAM
- 非同期 EMIF/NOR/NAND フラッシュ

外部メモリ・コントローラ (EMC) により、これらのメモリへのDSPによるアクセスが容易になります。DSPがアクセス可能な外部メモリは、次のとおりです。

- DDR2 ポート
- 非同期 EMIF (4つのEM\_CS領域にあるNORおよびNANDフラッシュ)

これらの外部メモリのメモリマップ・ロケーションについては、各デバイスのデータ・マニュアルの「メモリマップ」を参照してください。

### 3.1.3 内部ペリフェラル

DSPがアクセス可能な内部ペリフェラルは、次のとおりです。

- パワー・ダウン・コントローラ (PDC)
- 割り込みコントローラ (INTC)

内部ペリフェラルの詳細については、『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) を参照してください。

### 3.1.4 デバイス・ペリフェラル

DSPは、デバイスに搭載されているすべてのペリフェラルへアクセスできます。ペリフェラルの詳細なリストについては、各デバイスのデータ・マニュアルを参照してください。



## 3.2 メモリ・インターフェイスの概要

ここでは、DM643x DMP の各種メモリ・インターフェイスについて説明します。DM643x DMP は、次に示すようないくつかのメモリおよび外部デバイス・インターフェイスをサポートしています。

- DDR2 シンクロナス DRAM
- 非同期 EMIF/NOR/NAND フラッシュ

### 3.2.1 DDR2 外部メモリ・インターフェイス

DDR2 外部メモリ・インターフェイス (EMIF) ポートは、DDR2 SDRAM 専用のインターフェイスです。JESD79D-2A 規格に準拠した DDR2 SDRAM デバイスをサポートし、16 ビットと 32 ビットのインターフェイスをいずれもサポートしています。

DDR2 SDRAM は、DM643x DMP を中心としたシステムで重要な役割を果たしています。高速な外部メモリを大量に必要とするこのようなシステムでは、次のようなケースが考えられます。

- センサまたはビデオ・ソースからの入力画像データのバッファリング
- ビデオ・プロセッシング・フロント・エンド (VPFE) で画像データの処理またはリサイズを行うための中間バッファリング
- ビデオ・プロセッシング・バック・エンド (VPBE) のディスプレイ・バッファ
- スチル・カメラ処理機能を実行中の、大容量の raw Bayer データ・イメージ・ファイルの中間バッファリング
- ビデオのエンコードまたはデコード機能を実行中に一時的に作成される中間データのバッファリング
- DSP 用の実行可能なファームウェアの保存

### 3.2.2 外部メモリ・インターフェイス

DM643x DMP の外部メモリ・インターフェイス (EMIF) では、メモリ制御信号とともに 8 ビットのデータ・バス、最大 24 ビットのアドレス・バス幅、および 4 つの専用チップ・セレクトを提供しています。これらの信号は、非同期 EMIF を実現する非同期 EMIF (EMIFA) モジュールと NAND インターフェイス間で静的にマルチプレクスされます。

EMIFA 信号は、デバイス上の他のペリフェラル信号とマルチプレクスされています。ピン・マルチプレクスの詳細については、各デバイスのデータ・マニュアルを参照してください。

#### 3.2.2.1 EMIF 非同期インターフェイス

非同期 EMIF (EMIFA) インターフェイスでは、非同期 EMIF インターフェイスと NAND インターフェイスの両方を使用することができます。4 つのチップ・セレクトが用意されています。非同期 EMIF または NAND をサポートするために、チップ・セレクトそれぞれを個別に設定可能です。

- 非同期 EMIF モードでは、非同期デバイス (RAM、ROM、NOR フラッシュ) をサポート
- 4 つのチップ・セレクト (それぞれ 16MB) にまたがる 64MB の非同期アドレス領域
- 8 ビットのデータ・バス幅をサポート
- プログラム可能な非同期サイクル・タイミング
- 拡張ウェイト・サイクルをサポート
- 選択ストローブ・モードをサポート
- TI DSP HPI インターフェイスをサポート
- CS2 (SRAM/NOR フラッシュ) から DM643x DMP をブートする機能をサポート

### 3.2.2.2 NAND インターフェイス

非同期 EMIF (EMIFA) インターフェイスでは、非同期 EMIF インターフェイスと NAND インターフェイスの両方を使用することができます。4 つのチップ・セレクトが用意され、EMIFA または NAND をサポートするために、チップ・セレクトそれぞれを個別に設定可能です。

- NAND モードでは、最大 4 つの非同期チップ・セレクトで NAND フラッシュをサポート
- 8 ビットのデータ・バス幅をサポート
- サイクル・タイミングをプログラム可能
- ECC の計算を実行
- ブート ROM に含まれているブートルoader・コードが CS2 にある NAND フラッシュから DM643x DMP をブートする機能をサポート

# デバイス・クロッキング

---

---

---

項目	ページ
4.1 概要.....	28
4.2 クロック・ドメイン .....	28

## 4.1 概要

DM643x DMP には、プライマリ・リファレンス・クロックが1つ必要です。プライマリ・リファレンス・クロックは、発振子入力にすることも外部発振器からドライブすることもできます。システム PLL には、MXI/CLKIN ピンの 27 MHz の発振子入力をお勧めします。これを使用して DSP、ペリフェラル、DMA、画像処理ペリフェラルにクロックを供給します。推奨する 27 MHz の発振子入力を用いることで、NTSC/PAL テレビ信号を適切な周波数でドライブするビデオ DAC に使用することができます。

クロック周波数および電圧要件に関する詳細な仕様については、各デバイスのデータ・マニュアルを参照してください。

クロッキング・モードは2つあります。

- PLL バイパス・モード：節電（デバイスはこのモードがデフォルト）
- PLL モード：PLL で、入力クロックを目的の動作周波数に引き上げます。

それぞれのモード内で1次システム / DSP クロック周波数が一定の比率で動作するように、主要なチップ・サブシステムのクロックをプログラムする必要があります（表 4-1 を参照）。DM643x DMP のクロッキング・アーキテクチャを図 4-1 に示します。

**表 4-1. システム・クロック・モードおよびクロック・ドメインの一定の比率**

サブシステム	コア・クロック・ドメイン	一定の比率と DSP 周波数
DSP	CLKDIV1	1:1
EDMA	CLKDIV3	1:3
VPSS		
ペリフェラル (CLKDIV3 ドメイン)	CLKDIV3	1:3
ペリフェラル (CLKDIV6 ドメイン)	CLKDIV6	1:6

## 4.2 クロック・ドメイン

### 4.2.1 コア・ドメイン

コア・ドメインは、DSP / EDMA / ペリフェラルなど DM643x DMP の内部処理エレメントすべてに対するクロック・ドメインを表します。DSP とモジュール間の内部通信はすべてコア・ドメイン・クロック周波数で動作します。コア・クロック・ドメインはすべて相互に同期し、単一 PLL (PLL1) によってもたらされ、アラインされたクロック・エッジを備え、一定の分周比要件があります（表 4-1 および図 4-1 を参照）。ユーザの責任において、これらのコア・クロック・ドメイン間で一定の分周比を確保してください。

DSP は、CLKDIV1 ドメインにあり、PLL1 周波数 (PLL コントローラ 1 (PLL1) の PLLDIV1) を直接受け取るか、分周された PLL1 周波数 (2 分周、3 分周などにセットされた PLL1 の PLLDIV1) を受け取ります。DSP は、オンチップの他のコンポーネントと通信する目的で、 $DSP \div 3$  クロック周波数を作成するために使用する内部クロック分周器を備えています。

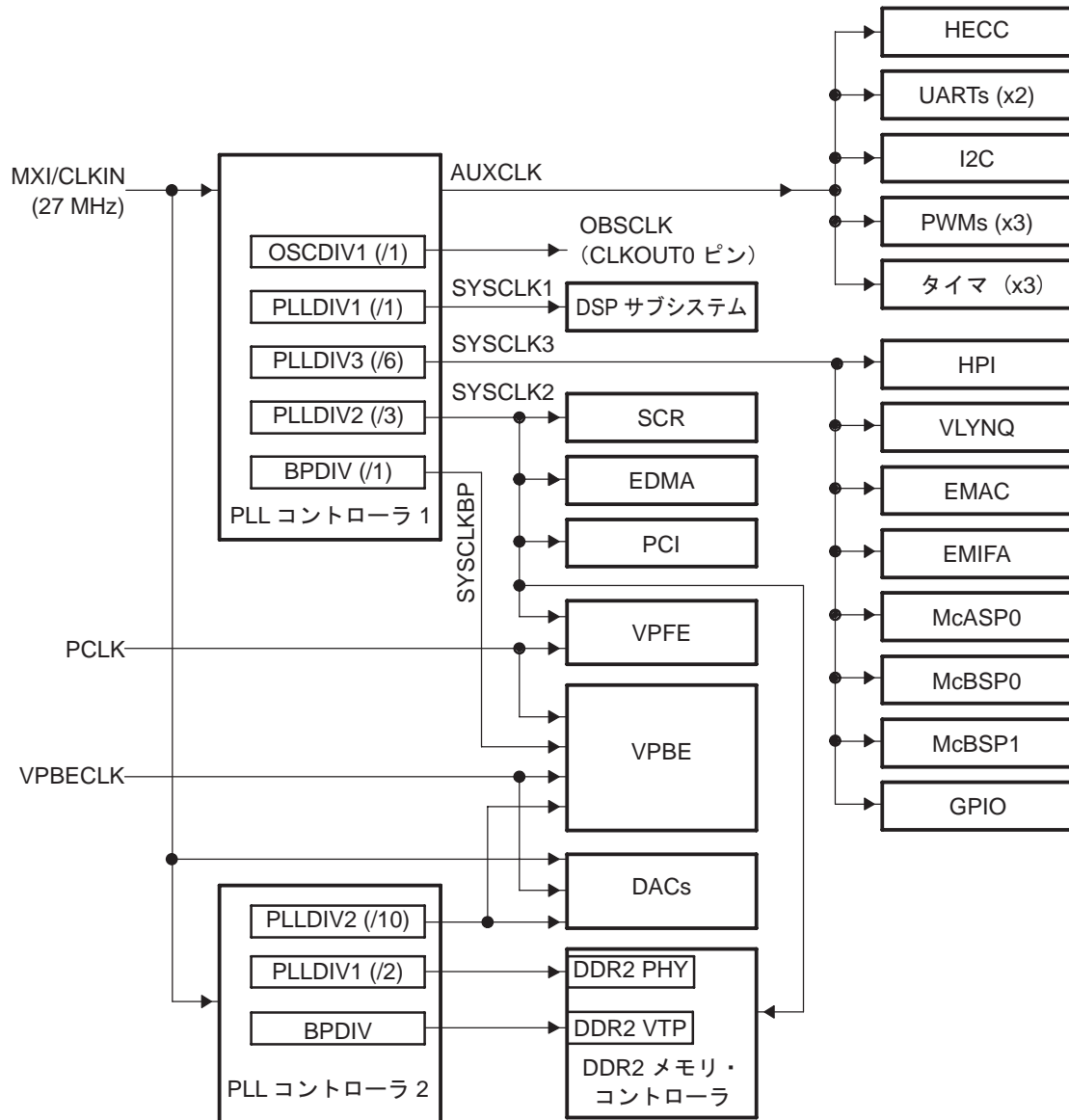
CLKDIV3 ドメインのモジュール (EDMA、VPSS、CLKDIV3 ドメイン・ペリフェラルなど) は、DSP の周波数の 1/3 で動作する必要があります。

CLKDIV6 ドメインのモジュール (CLKDIV6 ドメイン・ペリフェラルなど) は、DSP の周波数の 1/6 で動作する必要があります。

CLKIN ドメインのモジュール (UART、タイマ、I2C、PWM、HECC など) は、DSP とは非同期に、MXI/CLKIN 周波数で動作します。これらのペリフェラルの周波数と DSP の周波数の間には、一定の比率で動作するという要求事項はありません。

ペリフェラルごとのコア・クロック・ドメインについては、各デバイスのデータ・マニュアルを参照してください。

図 4-1. クロッキング全体図



## 4.2.2 コア周波数の柔軟性

コア周波数ドメイン・クロックは、PLL コントローラ 1 (PLL1) から供給されます。これらのドメイン・クロックは、多少柔軟性がありますが、各デバイスのデータ・マニュアルで指定されている制限事項もあります。データ・マニュアルに記述されている次の周波数範囲および通倍器 / 分周器の比率はすべて、次の条件に従う必要があります。

- 入力クロック周波数範囲 (MXI/CLKIN)
- PLL1 通倍器 (PLLM) 範囲
- デバイスのコア電圧 (1.05V または 1.2V) に基づく PLL1 出力 (PLLOUT) 周波数範囲
- デバイスの最大動作速度
- PLL1 の SYSCLK3:SYSCLK2:SYSCLK1 周波数比は、1:3:6 に固定しておく必要があります。たとえば、SYSCLK1 が 600 MHz で動作する場合、SYSCLK2 は 200 MHz で、SYSCLK3 は 100 MHz でそれぞれ動作するように設定する必要があります。

データ・マニュアルで指定されているように、20 ~ 30 MHz の入力周波数範囲で PLL をドライブすることができます。ただし、ビデオ・プロセッシング・バック・エンド (VPBE) サブシステムが内蔵のビデオ DAC を使用してテレビ・ディスプレイをドライブする必要がある場合、27 MHz で入力することが必要です。

表 4-2 に、MXI/CLKIN 周波数が 27 MHz における PLL1 通倍器および分周器の設定値の例を示します。[ デバイスへ適用可能なコア電圧 ] の列は、設定値が特定のデバイス・コア電圧に対して許可されているかを示しています。たとえば、表 4-2 の一番下の行 (クロック入力周波数 27 MHz の PLL1 通倍器 22) は、PLL1 出力 (PLLOUT) 周波数範囲に対応するコア電圧 1.2V のデバイスにのみ適用されます。これは、データ・マニュアルで要求されている仕様です。また、SYSCLK1 周波数はデバイスの動作速度を超えないようにする必要があります。たとえば、定格 400 MHz で動作するデバイスの場合、SYSCLK1 の周波数は 400 MHz を超えてはいけません。

表 4-2. PLL1 の周波数および分周比の例 (クロック入力周波数 27 MHz)

PLL1 通倍器	PLL1 PLLOUT 周波数 (MHz)	CLKDIV1 ドメイン (SYSCLK1)		CLKDIV3 ドメイン (SYSCLK2)		CLKDIV6 ドメイン (SYSCLK3)		デバイスへ適用コア 電圧	
		分周比 <sup>(1)</sup>	周波数 (MHz)	分周比 <sup>(1)</sup>	周波数 (MHz)	分周比 <sup>(1)</sup>	周波数 (MHz)	1.2V	1.05V
15	405.0	1	405.0	3	135.0	6	67.5	Y	Y
16	432.0	1	432.0	3	144.0	6	72.0	Y	Y
17	459.0	1	459.0	3	153.0	6	76.5	Y	Y
18	486.0	1	486.0	3	162.0	6	81.0	Y	Y
19	513.0	1	513.0	3	171.0	6	85.5	Y	Y
20	540.0	1	540.0	3	180.0	6	90.0	Y	-
21	567.0	1	567.0	3	189.0	6	94.5	Y	-
22	594.0	1	594.0	3	198.0	6	99.0	Y	-
22	594.0	2	297.0	6	99.0	12	49.5	Y	-

<sup>(1)</sup> PLLDIV<sub>n</sub> の RATIO ビットでは、分周比 -1 としてプログラムされます。たとえば、SYSCLK1 の分周比が 1 の場合、PLLDIV1.RATIO = 0、PLLDIV2.RATIO = 2、PLLDIV3.RATIO = 5 となるようにプログラムしてください。

### 4.2.3 DDR2/EMIF クロック

DDR2 インターフェイスは、PLL2 から駆動される専用クロックを備えています。これは、システムの他のコンポーネントに提供される PLL1 クロックとは別のクロック・システムです。この専用クロックにより、DDR2 への必要最小限のクロック・レート (125 MHz) を保持しながら節電するためにコア・クロック・レートを減少させることができます。DDR2 PHY インターフェイスに対して 2 倍のクロックを出力するように、PLL2 を設定する必要があります。

PLL2 から供給される 54 MHz の入力クロック周波数で駆動された場合、DM643x DMP のビデオ DAC は、高性能なプロダクション・テレビ・ディスプレイをドライブする機能を持っています (詳細については、『TMS320DM643x DMP Video Processing Back End (VPBE) User's Guide』(資料番号 SPRU952) を参照)。これにより PLL2 が取りうる設定値は 54 MHz の倍数に制限されます。そのため、VPBE クロックはシンプルな整数クロック分周比で計算できます。

PLL2 を設定するときは、各デバイスのデータ・マニュアルに記述されている次の周波数範囲および通倍器 / 分周器の比率はすべて、次の条件に従う必要があります。

- 入力クロック周波数範囲 (MXI/CLKIN)
- PLL2 通倍器 (PLLM) 範囲
- デバイスのコア電圧 (1.05V または 1.2V) に基づく PLL2 出力 (PLLOUT) 周波数範囲

表 4-3 および表 4-4 に、MXI/CLKIN 周波数が 27 MHz の場合の PLL2/DDR2 のクロック・レートを一部示します。またこれらの表では、54 MHz の倍数になっている設定値を示します。

表 4-3. PLL2 周波数の例 (コア電圧 = 1.2V)

PLL2 通倍器	PLL2 PLLOUT 周波数 (MHz)	SYSCLK1 分周比 <sup>(1)</sup>	PHY [クロックの 2 倍] (MHz)	DDR2 クロック (MHz)	54 MHz の倍数
28	756.0	3	252.0	126.0	はい
19	513.0	2	256.5	128.3	いいえ
29	783.0	3	261.0	130.5	いいえ
20	540.0	2	270.0	135.0	はい
31	837.0	3	279.0	139.5	いいえ
21	567.0	2	283.5	141.8	いいえ
32	864.0	3	288.0	144.0	はい
22	594.0	2	297.0	148.5	はい
23	621.0	2	310.5	155.3	いいえ
24	648.0	2	324.0	162.0	はい
25	675.0	2	337.5	168.8	いいえ

<sup>(1)</sup> PLLDIV<sub>n</sub> の RATIO ビットでは、分周比 -1 としてプログラムされます。たとえば、SYSCLK1 の分周比が 3 の場合、PLLDIV1.RATIO = 2 となるようにプログラムしてください。

**表 4-4. PLL2 周波数の例 (コア電圧 = 1.05V)**

PLL2 通倍器	PLL2 PLLOUT 周波数 (MHz)	PHY [クロックの2倍] DDR2 クロック SYSCLK1 分周比 <sup>(1)</sup> (MHz)	54 MHz の倍数
19	513.0	2	いいえ
20	540.0	2	はい
21	567.0	2	いいえ
22	594.0	2	はい
23	621.0	2	いいえ
24	648.0	2	はい

<sup>(1)</sup> PLLDIV<sub>n</sub> の RATIO ビットでは、分周比 -1 としてプログラムされます。たとえば、SYSCLK1 の分周比が 3 の場合、PLLDIV1.RATIO = 2 となるようにプログラムしてください。

#### 4.2.4 I/O ドメイン

I/O ドメインは、デバイス・ピンを使用して通信を行うペリフェラルの周波数を表します。多くの場合、外部規格によって規定され、対応する必要があるペリフェラル・ピンのインターフェイスには周波数に関する要求事項があります。必ずしもオンチップ・クロック生成回路からこれらの周波数が得られるとは限りません。そのため、周波数は外部ソースから供給され、コア周波数ドメインと非同期で動作することになります。

表 4-5 に、外部入出力 (I/O) インターフェイスを備えたペリフェラルと、その I/O ドメイン・クロック / 周波数を示します。また内部通信で使われるコア・クロックを示すために、コア・クロック・ドメインを参照用に示します。コア・クロック・ドメインの詳細については、4.2.1 項を参照してください。デバイスでサポートされている正しい入出力 (I/O) クロック周波数については、各デバイスのデータ・マニュアルを参照してください。

**表 4-5. ペリフェラル I/O ドメイン・クロック**

ペリフェラル	I/O ドメイン・クロック 周波数	I/O (外部) ドメイン・クロック・ソース・オプション		コア・クロック・ドメイン
		内部クロック・ソース	外部クロック・ソース	
DDR2	125 ~ 166 MHz	PLL2 SYSCLK1	-	CLKDIV3
VPFE	10 ~ 98 MHz	-	PCLK	CLKDIV3
VPBE	6.25 ~ 75 MHz	PLL1 SYSCLKBP (通常 27 MHz)	VPBECLK	CLKDIV3
		PLL2 SYSCLK2 (通常 54 MHz)	PCLK	
PCI	33 MHz	-	PCICLK	CLKDIV3
EMAC	25 MHz	-	MTXCLK、MRXCLK	CLKDIV6
VLYNQ	最大 80 MHz	PLL1 SYSCLK3	VLYNQ_CLOCK	CLKDIV6
McBSP	最大 40 MHz	PLL1 SYSCLK3	CLKS、CLKX、CLKR	CLKDIV6
McASP	最大 40 MHz	PLL1 SYSCLK3	AHCLKX、AHCLKR、ACLKX、ACLKR	CLKDIV6
GPIO	該当なし (非同期インターフェイス)	-	-	CLKDIV6
EMIFA	該当なし (非同期インターフェイス)	-	-	CLKDIV6
HPI	該当なし (非同期インターフェイス)	-	-	CLKDIV6
I2C	最大 400 kHz	MXI/CLKIN (通常 27 MHz)	SCL	CLKIN
タイマ	最大 1/2 CLKIN 出力周波数 最大 1/4 CLKIN 入力周波数	MXI/CLKIN (通常 27 MHz)	TINP0L (タイマ 0)、 TINP1L (タイマ 1)	CLKIN



表 4-5. ペリフェラル I/O ドメイン・クロック (続き)

ペリフェラル	I/O ドメイン・クロック 周波数	I/O (外部) ドメイン・クロック・ソース・オプション		コア・クロック・ドメイン
		内部クロック・ソース	外部クロック・ソース	
ウォッチドッグ・タイマ	該当なし	MXI/CLKIN (通常 27 MHz)	-	CLKIN
PWM	該当なし	-	-	CLKIN
UART	該当なし	-	-	CLKIN
HECC	該当なし	-	-	CLKIN

### 4.2.5 ビデオ・プロセッシング・バック・エンド

ビデオ・プロセッシング・バック・エンド (VPBE) は、ビデオ・プロセッシング・サブシステム (VPSS) のサブモジュールです。VPBE は 4 チャンネルの DAC モジュールだけでなく、さまざまな LCD と接続されます。LCD にはさまざまな種類があり、そのため必要とする周波数が異なります。ピン・インターフェイスの動作に必要な周波数範囲は、6.25 MHz ~ 75 MHz です。

VPBE には、外部クロック・ドメイン (6.25 MHz ~ 75 MHz) と、 $DSP \div 3$  クロック・レートで動作する内部 (システム) クロック・ドメインの 2 つの非同期クロック・ドメインがあります。

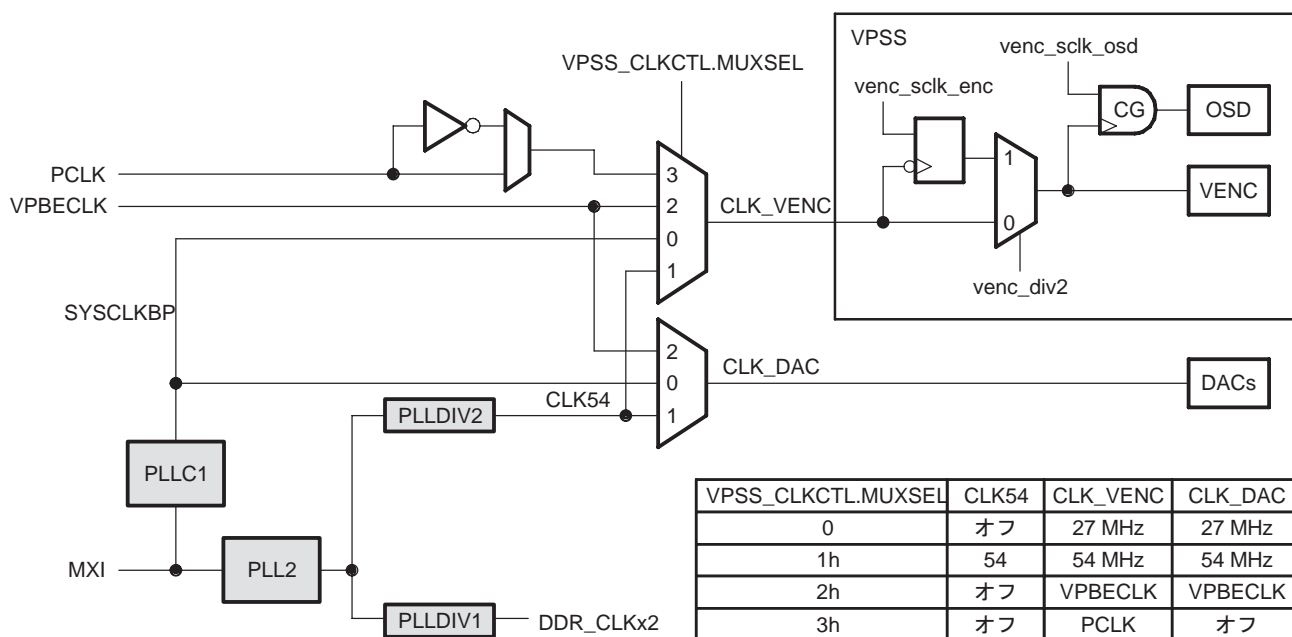
外部クロック・ドメインは、次の 4 種類のソースからクロックが供給されます。

- PLLC1 SYSCLKBP (通常 27 MHz、MXI/CLKIN の 1 分周)
- VPBECLK 入力ピン
- VPFE ピクセル・クロック入力 (PCLK)
- PLLC2 SYSCLK2 (PLL2 からの分周)

4 つの DAC は、VPBE 内部の VENC モジュールに接続されています。VPBE と DAC とのデータ・フローは同期動作です。対応可能なクロッキング・モードを図 4-2 に示し、表 4-6 で説明します。

DAC のクロックは非使用時に、独立してゲートオフしておくこともできます。これについては、第 7 章で説明します。

図 4-2. VPBE/DAC クロッキング



**表 4-6. 対応可能なクロッキング・モード**

VPSS_CLKCTL.MUXSEL Bit	クロッキング・モード	説明
0	MXI モード	VENC と DAC は両方とも MXI 27 MHz の発振子入力 の 1 分周をデフォルトとする PLLC1 SYSCLKBP からクロック供給されています。
1h	PLL2 モード	PLL2 (分周) は 54 MHz のクロックを生成します。DAC と VENC は両方とも 54 MHz のクロックを受け取ります。VENC は、27 MHz のクロックを作成するためにオプションで 2 分周することもできます。このモードでは、54 MHz の DAC クロックを作成するために整数の除数が使用できるように、( PLL2 からの ) DDR2 クロック設定を 27 MHz の偶数倍にセットする必要があることに注意してください。そのため、このモードでは使用できる DDR2 クロック周波数が制限されています。
2h	VPBECLK モード	DAC と VENC は両方とも VPBECLK を受け取ります。VPBECLK 上でプログレッシブ・スキャンによる 54 MHz での駆動をサポートするために、VENC にはこのクロックを 2 分周するオプションがあります。
3h	PCLK モード	VENC は PCLK を受け取ります。DAC はクロックを受け取らないので、ディスエーブルしてください。PCLK はメモリ・マップド・レジスタ・ビットで選択することにより、ネガティブ・エッジをサポートするために極性を反転することが可能です。

## PLL コントローラ

項目	ページ
5.1 PLL モジュール.....	36
5.2 PLL1 による制御.....	36
5.3 PLL2 による制御.....	41
5.4 PLL コントローラのレジスタ .....	46

## 5.1 PLL モジュール

DM643x DMP には 2 つの PLL (PLL1 と PLL2) があり、システムのさまざまな部分へクロックを供給しています。PLL1 は DM643x DMP に搭載されているほとんどのコンポーネントへ (分周比は異なりますが) クロックを供給しています。PLL2 は、DDR2 ポートおよびビデオ・プロセッシング・サブシステム (VPSS) のコンポーネントへのみクロックを供給します。通常のリファレンス・クロックは、27 MHz の発振子入力です (第 4 章を参照)。

PLL コントローラが行っている内容は、次のとおりです。

- グリッチ・フリー遷移 (クロック設定値の変更時)
- ドメイン・クロック・アライメント
- クロック・ゲート制御
- PLL のパワー・ダウン

コントローラによってもたらされる各種クロック出力を次に示します。

- ドメイン・クロック : SYSCLK[1:n]
- リファレンス・クロック・ソースからの補助クロック : AUXCLK
- バイパス・ドメイン・クロック : SYSCLKBP
- 監視クロック : OBSCLK

DM643x DMP 上で使用できる各種分周器は、次のとおりです。

- PLL コントローラ分周器 (SYSCLK[1:n] の場合): PLLDIV1、...、PLLDIV $n$
- バイパス分周器 (SYSCLKBP の場合): BPDIV
- オシレータ分周器 (OBSCLK の場合): OSCDIV1

サポートされている他の各種制御機能は、次のとおりです。

- PLL 逡倍器の制御 : PLLM
- ソフトウェアでプログラム可能な PLL のバイパス : PLEN

## 5.2 PLL1 による制御

DM643x DMP の PLL1 は、プライマリ・システム・クロックを供給します。PLL1 の動作はシステム PLL コントローラ 1 レジスタ (PLLC1) を介して、ソフトウェアにて制御します。PLLC1 で使用されるレジスタを 5.4 節に示します。図 5-1 に、DM643x DMP の PLL1 のカスタマイズした内容を示します。ドメイン・クロックはコア・ブロック・ドメイン (4.2.1 項を参照) および他のデバイスに供給されます。

- SYSCLK1 : CLKDIV1 ドメイン
- SYSCLK2 : CLKDIV3 ドメイン
- SYSCLK3 : CLKDIV6 ドメイン
- AUXCLK : CLKIN ドメイン
- OBSCLK : CLKOUT0 ピン
- SYSCLKBP : VPBE 内部クロック・ソース

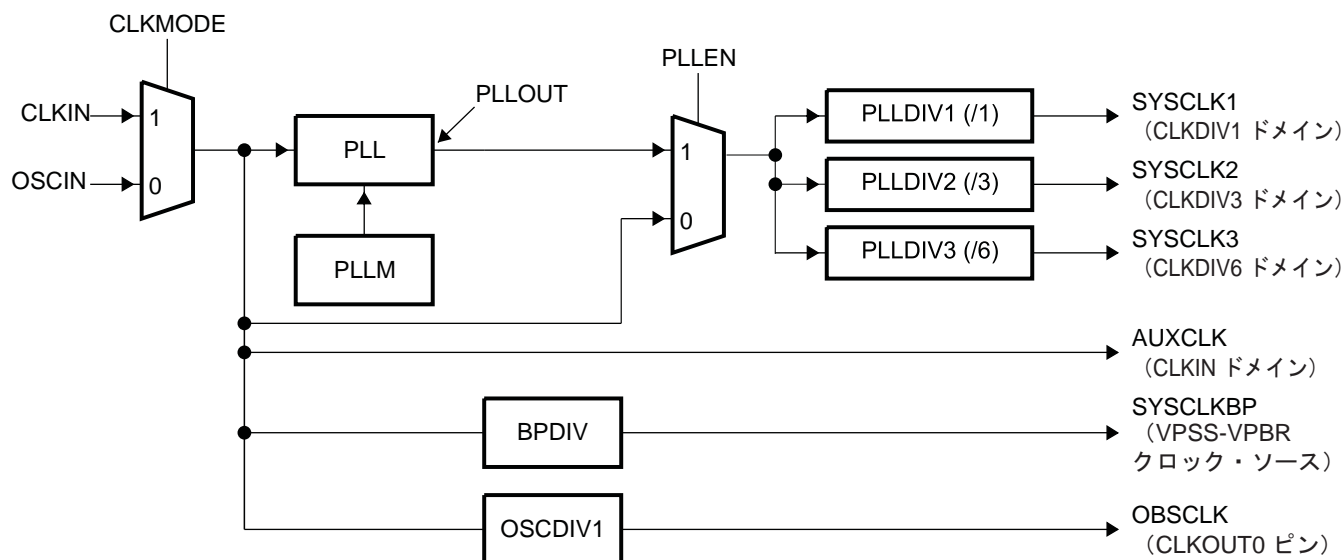
PLL1 逡倍器を制御するには、PLL マルチプライア・コントロール・レジスタ (PLLM) の PLLM ビットを使用します。PLL1 出力クロックは、より低速なデバイスが動作できるようにするために PLLC1 SYSCLK 分周器である PLLDIV1、PLLDIV2、PLLDIV3 を使用して分周します。

データ・マニュアルに指定されている PLLC1 周波数範囲および逡倍器 / 分周器の比率に従ってプログラムしてください。また 4.2.1 項や 4.2.2 項も参照してください。

電源投入時に、PLL1 はパワー・ダウン状態でディスエーブルです。ソフトウェアで PLL コントロール・レジスタ (PLLCTL) の PLL1 PLLPWRDN ビットを介してパワー・アップする必要があります。デフォルトでは、システムはバイパス・モードで動作し、システム・クロックは入力リファレンス・クロック (MXI/CLKIN ピン) から直接供給されます。PLL がバ

ワー・アップされ、かつロックされると、ソフトウェアで PLLCTL の PLEN ビットを 1 にセットしてデバイスを PLL モードで動作するように切り替えることができます。デバイスのブート時のモードが高速ブート (FASTBOOT = 1) にセットされている場合、ブート ROM に含まれているブートローダのコードは、PLL をパワー・アップしロックするために前のプロセスに従い、デバイスを PLL モードに切り替えて、ブート・プロセスを短縮します。したがって、高速ブート後のデバイスは PLL モードで動作しています。

図 5-1. TMS320DM643x DMP に搭載されている PLL1 の構成



### 5.2.1 デバイス・クロックの生成

PLL1 は、各種プロセッサおよびモジュールで使用できるさまざまなクロックを PLL1 出力クロックから生成します。表 5-1 にこのリストを示します。SYSCLK1、SYSCLK2、SYSCLK3 は、リファレンス・クロック (PLL またはバイパス) と PLL 周波数のどちらが使われていても一定の周波数比に関する要件を保たなければなりません。

表 5-1. システム PLL1 出力クロック

PLL1 出力クロック	使用するコンポーネント	デフォルトの分周比
SYSCLK1	DSP サブシステム	/1
SYSCLK2	SCR、EDMA、VPSS、CLKDIV3 ドメイン・ペリフェラル	/3
SYSCLK3	CLKDIV6 ドメイン・ペリフェラル	/6
AUXCLK	CLKIN ドメイン・ペリフェラル	該当なし
OBSCLK	CLKOUT0 ソース	/1
SYSCLKBP	VPBE クロック・ソース	/1

## 5.2.2 PLL1 / コア・ドメイン周波数を変更する手順

PLL1 / コア・ドメイン・クロックをプログラムする方法については、該当のサブセクションを参照してください。

- PLL がパワー・ダウン状態の場合 (PLLCTL の PLLPWRDN ビットが 1 にセットされている)、5.2.2.1 項に示す PLL を完全に初期化する手順に従います。
- PLL がパワー・ダウン状態ではない場合 (PLLCTL の PLLPWRDN ビットが 0 にクリアされている)、5.2.2.2 項に示す PLL 逡倍器を変更する手順に従います。
- PLL が指定の乗算比ですでに動作し、かつ SYSCLK の分周比だけを変更する場合、5.2.2.3 項に示す手順に従います。

次のデバイス・レベルでグローバル・リセットされた後、PLL はパワー・ダウン状態になることに注意してください。

- パワー・オン・リセット ( $\overline{\text{POR}}$ )
- ウォーム・リセット ( $\overline{\text{RESET}}$ )
- マックス・リセット

### 5.2.2.1 PLL パワー・ダウン・モードから PLL モードへの初期化

PLL がパワー・ダウン状態の場合 (PLLCTL の PLLPWRDN ビットが 1 にセットされている) PLL1 の周波数を変更するためには下記に示す手順に従う必要があります。PLL1 の周波数を変更する前に、C64x+ DSP および DDR2 を除くすべてのペリフェラルの動作を停止することをお勧めします。C64x+ DSP は、PLL コントローラをプログラムするために動作していなければなりません。DDR2 は、PLL2 からクロックの動作をオフにします。

1. PLLCTL の CLKMODE ビットをプログラムすることにより、クロック・モードを選択します。
2. PLL の周波数を変更する前に、PLL バイパス・モードに切り替えます。
  - a. PLLCTL.PLEN を有効にするために、PLLCTL の PLENSRC ビットを 0 にクリアします。
  - b. PLLCTL の PLEN ビットを 0 にクリアします (PLL バイパス・モードを選択します)。
  - c. PLLC が適切にバイパス・モードへ切り替わるために、4 MXI サイクル待ちます。
3. PLLCTL の PLLRST ビットを 0 にクリアします (PLL をリセットします)。
4. PLLCTL の PLLDIS ビットを 1 にセットします (PLL 出力をディスエーブルします)。
5. パワー・ダウン・モードから PLL モードへするために、PLLCTL の PLLPWRDN ビットを 0 にクリアします。
6. PLL 出力が切り替わって動作を開始できるようにするために、PLLCTL の PLLDIS ビットを 0 にクリアします (PLL をイネーブルします)。PLLC は PLL バイパス・モードで動作したままであることを注意してください。そのため、PLL 出力を切り替えたという情報は他のデバイスへ伝播されていません。
7. PLL が安定するまで待ちます。PLL が安定するまでの時間については、各デバイスのデータ・マニュアルを参照してください。
8. PLLM に要求する逡倍器の値をプログラムします。
9. 必要に応じて、SYSCLK1、SYSCLK2、SYSCLK3 の各分周比の値を変更するために、PLLDIV1、PLLDIV2、PLLDIV3 の各レジスタをプログラムします。
  - a. GO 動作が現在進行中ではないことを示すために、PLLSTAT の GOSTAT ビットが 0 にクリアされているかチェックします。
  - b. PLLDIV1、PLLDIV2、PLLDIV3 の各 RATIO フィールドを指定の分周値にプログラムします。分周比は、CLKDIV1、CLKDIV3、CLKDIV6 の各クロック・ドメイン要件を満たすように、1:3:6 の比率を保持する必要があることに注意してください。クロック・ドメインの詳細については、各デバイスのデータ・マニュアルを参照してください。また、このステップでは、PLLDIV1.D1EN、PLLDIV2.D2EN、PLLDIV3.D3EN の各ビットを (デフォルトに) セットしたままにしておいてください。
  - c. 新たな分周比への遷移を開始するために、PLLCMD の GOSET ビットを 1 にセットします。この遷移中に、SYSCLK1、SYSCLK2、SYSCLK3 は一時的に動作を停止します。

- d. 分周比の変更を確実に完了させるために、PLLDIV $n$  ソース・クロックの  $N$  サイクル分待ちます。サイクル数  $N$  の計算式については、5.2.2.3 項を参照してください。
- e. PLLSTAT の GOSTAT ビットが 0 にクリアされるまで待ちます。
10. PLL が適切にリセットされるまで待ちます。PLL がリセットされるまでの時間については、各デバイスのデータ・マニュアルを参照してください。
11. リセット状態から PLL モードへするために、PLLCTL の PLLRST ビットを 1 にセットします。
12. PLL がロックされるまで待ちます。PLL がロックされるまでの時間については、各デバイスのデータ・マニュアルを参照してください。
13. バイパス・モードから PLL モードへ移行するために、PLLCTL の PLEN ビットを 1 にセットします。

### 5.2.2.2 PLL 逡倍器の変更

PLL がパワー・ダウン状態ではなく (PLLCTL の PLLPWRDN ビットが 0 にクリアされていて) かつ PLL がすでに安定している状態の (5.2.2.1 項のステップ 7) 場合、次の手順に従って PLL1 逡倍器を変更します。PLL 逡倍器を変更する前に、C64x+ DSP および DDR2 を除くすべてのペリフェラルの動作を停止することをお勧めします。C64x+ DSP は、PLL コントローラをプログラムするために動作していなければなりません。DDR2 は、PLLC2 からのクロックの動作をオフにします。

1. PLL の周波数を変更する前に、PLL バイパス・モードに切り替えます。
  - a. PLLCTL.PLEN を有効にするために、PLLCTL の PLENSRC ビットを 0 にクリアします。
  - b. PLLCTL の PLEN ビットを 0 にクリアします (PLL バイパス・モードを選択します)。
  - c. PLLC が適切にバイパス・モードへ切り替わるために、4 MXI サイクル待ちます。
2. PLLCTL の PLLRST ビットを 0 にクリアします (PLL をリセットします)。
3. PLL 出力が切り替わって動作を開始できるようにするために、PLLCTL の PLLDIS ビットを 0 にクリアします (PLL をイネーブルします)。PLLC は PLL バイパス・モードで動作したままであることに注意してください。そのため、PLL 出力を切り替えたという情報は他のデバイスへ伝播されていません。
4. PLLM に要求する逡倍器の値をプログラムします。
5. 必要に応じて、SYSCLK1、SYSCLK2、SYSCLK3 の各分周比の値を変更するために、PLLDIV1、PLLDIV2、PLLDIV3 の各レジスタをプログラムします。
  - a. GO 動作が現在進行中ではないことを示すために、PLLSTAT の GOSTAT ビットが 0 にクリアされているかチェックします。
  - b. PLLDIV1、PLLDIV2、PLLDIV3 の各 RATIO フィールドを指定の分周値にプログラムします。分周比は、CLKDIV1、CLKDIV3、CLKDIV6 の各クロック・ドメイン要件を満たすように、1:3:6 の比率を保持する必要があることに注意してください。クロック・ドメインの詳細については、各デバイスのデータ・マニュアルを参照してください。また、このステップでは、PLLDIV1.D1EN、PLLDIV2.D2EN、PLLDIV3.D3EN の各ビットを (デフォルトに) セットしたままにしておいてください。
  - c. 新たな分周比への遷移を開始するために、PLLCMD の GOSET ビットを 1 にセットします。この遷移中に、SYSCLK1、SYSCLK2、SYSCLK3 は一時的に動作を停止します。
  - d. 分周比の変更を確実に完了させるために、PLLDIV $n$  ソース・クロックの  $N$  サイクル分待ちます。サイクル数  $N$  の計算式については、5.2.2.3 項を参照してください。
  - e. PLLSTAT の GOSTAT ビットが 0 にクリアされるまで待ちます。
6. PLL が適切にリセットされるまで待ちます。PLL がリセットされるまでの時間については、各デバイスのデータ・マニュアルを参照してください。
7. リセット状態から PLL モードへ切り替えるために、PLLCTL の PLLRST ビットを 1 にセットします。

8. PLL がロックされるまで待ちます。PLL がロックされるまでの時間については、各デバイスのデータ・マニュアルを参照してください。
9. バイパス・モードから PLL モード移行するために、PLLCTL の PLEN ビットを 1 にセットします。

### 5.2.2.3 SYSCLK の分周比の変更

ここでは、SYSCLK の分周比を変更するソフトウェア・シーケンスについて説明します。また、分周比の変更を開始するために、GO ビット ( PLLCMD の GOSET ビット ) の起動を伴うので、SYSCLK の分周比を変更するシーケンスは GO 動作ともいいます。SYSCLK の分周比を変更する前に、C64x+ DSP および DDR2 を除くすべてのペリフェラルの動作を停止することをお勧めします。C64x+ DSP は、PLL コントローラをプログラムするために動作していなければなりません。DDR2 は、PLLC2 からのクロックの動作をオフにします。

1. GO 動作が現在進行中ではないことを示すために、PLLSTAT の GOSTAT ビットが 0 にクリアされているかチェックします。
2. PLLDIV1、PLLDIV2、PLLDIV3 の各 RATIO フィールドを指定の分周値にプログラムします。分周比は、CLKDIV1、CLKDIV3、CLKDIV6 の各クロック・ドメイン要件を満たすように、1:3:6 の比率を保持する必要があることに注意してください。クロック・ドメインの詳細については、各デバイスのデータ・マニュアルを参照してください。また、このステップでは、PLLDIV1.D1EN、PLLDIV2.D2EN、PLLDIV3.D3EN の各ビットを ( デフォルトに ) セットしたままにしておいてください。
3. 新たな分周比への遷移を開始するために、PLLCMD の GOSET ビットを 1 にセットします。この遷移中に、SYSCLK1、SYSCLK2、SYSCLK3 は一時的に動作を停止します。
4. 分周比の変更を確実に完了させるために、PLLDIV<sub>n</sub> ソース・クロックの N サイクル分待ちます。サイクル数 N の計算式については、次の式を参照してください。
5. PLLSTAT の GOSTAT ビットが 0 にクリアされるまで待ちます。

PLLDIV<sub>n</sub> ソース・クロック・サイクル数を計算するには、次の式を使用してください。

$$N = (\text{設定前のすべての SYSCLK 分周値の最小公倍数 [LCM] の 2 倍}) + 50 \text{ サイクルのオーバーヘッド}$$

#### 例 5-1. クロック・サイクル数 N の計算

次に、クロック・サイクル数 N を計算する例を示します。

- 分周比変更前の設定値 :
  - PLLDIV1.RATIO = 0 ( 1 分周 )
  - PLLDIV2.RATIO = 2 ( 3 分周 )
  - PLLDIV3.RATIO = 5 ( 6 分周 )
- 分周比変更後の新しい設定値 :
  - PLLDIV1.RATIO = 1 ( 2 分周 )
  - PLLDIV2.RATIO = 5 ( 6 分周 )
  - PLLDIV3.RATIO = 11 ( 12 分周 )

設定前の古い分周比の値 /1、/3、/6 の最小公倍数は /6 になるため、サイクル数 N を求める計算式は次のようになります。

$$N = (2 \times 6) + 50 \text{ サイクルのオーバーヘッド} = 62 \text{ PLLDIV}_n \text{ ソース・クロック・サイクル}$$

PLLC1 が PLL モード ( PLLCTL.PLEN = 1 ) の場合、PLLDIV<sub>n</sub> ソース・クロックは PLL1 出力クロックです。PLLC1 が PLL バイパス・モード ( PLLCTL.PLEN = 0 ) の場合、PLLDIV<sub>n</sub> ソース・クロックはデバイス・クロック・ソース MXI/CLKIN です。



### 5.3 PLL2 による制御

PLL2 は、DDR2 メモリ・コントローラおよびオプションの VPBE クロックの元となるクロックを供給します。DDR の PLL コントローラ 2 ( PLLC2 ) は PLL2 を制御し、オシレータからのクロックを受け付け、必要な各種周波数クロックも生成します。図 5-2 に、DM643x DMP の PLL2 のカスタマイズした内容を示します。PLL2 は、次に示すデバイスにクロックを供給します。

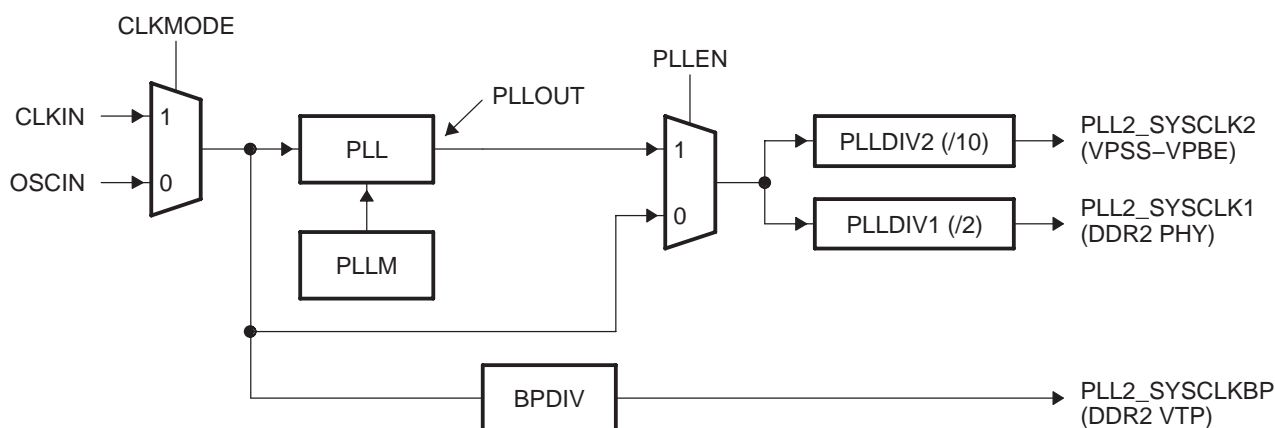
- SYSCLK1:DDR2 PHY
- SYSCLK2:VPSS
- SYSCLKBP:DDR2 VTP

PLL2 は DDR2 メモリ・コントローラにクロックを供給します。ソフトウェアで、DDR PLL コントローラ 2 ( PLLC2 ) レジスタを使用して PLL2 の動作を制御します。 PLLC2 で使用されるレジスタを 5.4 節に示します。 PLL2 逡倍器を制御するには、PLL マルチプライア・コントロール・レジスタ ( PLLM ) の PLLM ビットを使用します。 PLL2 逡倍器は、ソフトウェアで (たとえば、最高のパフォーマンスを発揮するように DDR インターフェイスを調整するように) 変更できます。

PLL2 出力クロックは、SYSCLK1 の分周器を使用して DDR が動作する周波数範囲に分周する必要があります。

電源投入時に、PLL2 はパワー・ダウン状態でディスエーブルです。ソフトウェアで PLL コントロール・レジスタ ( PLLCTL ) の PLL2 PLLPWRDN ビットを使用してパワー・アップする必要があります。デフォルトでは、システムはバイパス・モードで動作し、DDR クロックは入力ファレンス・クロックから直接供給されます。 PLL がパワー・アップされロックされると、ソフトウェアで PLLCTL の PLEN ビットを 1 にセットしてデバイスを PLL モードで動作するように切り替えることができます。

図 5-2. TMS320DM643x DMP に搭載されている PLL2 の構成



### 5.3.1 デバイス・クロックの生成

PLLC2 は、DDR2 メモリ・コントローラおよび VPSS モジュールで使用する 2 つのクロックを PLL2 出力クロックから生成します。これらの関係を表 5-2 に示します。

表 5-2. DDR PLLC2 出力クロック

出力クロック	使用するコンポーネント	デフォルトの分周比
SYSCLK2	VPSS	/10
SYSCLK1	DDR Phy	/2
SYSCCLKBP	DDR VTP コントローラ	/2

SYSCLK1 出力クロックの分周比のデフォルト値は、/2 です。27 MHz の MXI/CLKIN および PLL2 のデフォルトの倍数が 20 倍であるものと仮定すると、DDR Phy クロックは 270 MHz になります( 135 MHz で動作する DDR2 )。指定された DDR クロック・レートを実現するために、ソフトウェア ( PLLDIV1 の RATIO ビット ) を他の PLL 通倍器と組み合わせることにより、設定値を変更できます。SYSCLK2 の分周比は、VPSS で使用できるように偶数倍の PLL 出力周波数から 54 MHz 出力を生成できるようにプログラム可能です。

### 5.3.2 PLL2 周波数を変更する手順

PLLC2 は PLLC1 と同様の手順となります。PLL2 クロックをプログラムする方法については、該当のサブセクションを参照してください。

- PLL がパワー・ダウン状態の場合 ( PLLCTL の PLLPWRDN ビットが 1 にセットされている )、5.3.2.2 項に示す PLL を完全に初期化する手順に従います。
- PLL がパワー・ダウン状態ではない場合 ( PLLCTL の PLLPWRDN ビットが 0 にクリアされている )、5.3.2.3 項に示す PLL 通倍器を変更する手順に従います。
- PLL が指定の通倍器ですでに動作し、かつ SYSCLK の分周比だけを変更する場合、5.3.2.4 項に示す手順に従います。

次のデバイス・レベルでグローバル・リセットされた後、PLL はパワー・ダウン状態になることに注意してください。

- パワー・オン・リセット (  $\overline{\text{POR}}$  )
- ウォーム・リセット (  $\overline{\text{RESET}}$  )
- マックス・リセット

また、PLL2 周波数が DDR2 メモリ・コントローラおよび VPSS VPBE クロック・ソース ( ただし、PLLC2 SYSCLK2 が VPBE のクロック・ソースとして選択されている場合 ) に直接反映されることに注意してください。DDR2 メモリ・コントローラは、PLL2 周波数の変更前後で従うべき特別な手順を必要とします。DDR2 の動作に影響を与えないようにするために、5.3.2.1 項に示す DDR2 メモリ・コントローラに関する他の考慮事項にも従う必要があります。

#### 5.3.2.1 PLL2 周波数変更時の DDR2 に関する考慮事項

PLL2 と PLLC2 周波数の両方またはそのいずれかを変更する前に、DDR2 メモリ・コントローラの要件を考慮する必要があります。DDR2 メモリ・コントローラをシステムで使用する場合、DDR2 に影響を与えずに PLL2 と PLLC2 周波数の両方またはそのいずれかを変更するために、ここで示す追加手順に従ってください。

- PLL2 周波数を変更する際、DDR2 メモリ・コントローラがリセット状態の場合、5.3.2.1.1 項の手順に従ってください。
- PLL2 周波数を変更する際、DDR2 メモリ・コントローラがリセット状態ではない場合、5.3.2.1.2 項の手順に従ってください。

### 5.3.2.1.1 DDR2 メモリ・コントローラがリセット状態のときの PLL2 周波数変更手順

ここでは、DDR2 メモリ・コントローラがリセット状態のときに PLL2 周波数を変更する手順について説明します。パワー・オン・リセット、ウォーム・リセット、マックス・リセットといったデバイス・レベルでのグローバル・リセットが行われた後でも DDR2 メモリ・コントローラはリセット状態であることに注意してください。

1. DDR2 メモリ・コントローラはリセット状態のままにしておいてください。
2. 5.3.2.2 項、5.3.2.3 項、5.3.2.4 項のいずれかの手順に従って PLL2 クロックをプログラムします (5.3.2 項では、該当のサブセクションで説明する内容を示しています)。
3. DDR2 メモリ・コントローラを初期化します。DDR2 メモリ・コントローラを初期化する手順については、『TMS320DM643x DMP DDR2 Memory Controller User's Guide』(資料番号 [SPRU986](#)) を参照してください。

### 5.3.2.1.2 DDR2 メモリ・コントローラがリセット状態ではないときの PLL2 周波数変更手順

ここでは、DDR2 メモリ・コントローラがリセット状態ではないときに PLL2 周波数を変更する手順について説明します。

1. DDR2 メモリ・コントローラのアクセスを停止し、未処理のリクエストを取り除きます。
2. DDR2 メモリをセルフ・リフレッシュ・モードにし、DDR2 メモリ・コントローラのクロックを停止します。DDR2 メモリ・コントローラのクロックをシャットダウンする手順については、『TMS320DM643x DMP DDR2 Memory Controller User's Guide』(資料番号 [SPRU986](#)) を参照してください。
3. 5.3.2.2 項、5.3.2.3 項、5.3.2.4 項のいずれかの手順に従って PLL2 クロックをプログラムします (5.3.2 項では、該当のサブセクションで説明する内容を示しています)。
4. DDR2 メモリ・コントローラのクロックを再度イネーブルします。DDR2 メモリ・コントローラのクロックを有効にする手順については、『TMS320DM643x DMP DDR2 Memory Controller User's Guide』(資料番号 [SPRU986](#)) を参照してください。

### 5.3.2.2 PLL パワー・ダウン・モードから PLL モードへの初期化

PLL がパワー・ダウン状態の場合 (PLLCTL の PLLPWRDN ビットが 1 にセットされている) PLL2 の周波数を変更するためには下記に示す手順に従う必要があります。

1. PLLCTL の CLKMODE ビットをプログラムすることにより、クロック・モードを選択します。
2. PLL の周波数を変更する前に、PLL バイパス・モードに切り替えます。
  - a. PLLCTL.PLEN を有効にするために、PLLCTL の PLENSRC ビットを 0 にクリアします。
  - b. PLLCTL の PLEN ビットを 0 にクリアします (PLL バイパス・モードを選択します)。
  - c. PLLC が適切にバイパス・モードへ切り替わるために、4 MXI サイクル待ちます。
3. PLLCTL の PLLRST ビットを 0 にクリアします (PLL をリセットします)。
4. PLLCTL の PLLDIS ビットを 1 にセットします (PLL 出力をディスエーブルします)。
5. パワー・ダウン・モードから PLL モードへ切り替えるために、PLLCTL の PLLPWRDN ビットを 0 にクリアします。
6. PLL 出力が切り替わって動作を開始できるようにするために、PLLCTL の PLLDIS ビットを 0 にクリアします (PLL をイネーブルします)。PLLC は PLL バイパス・モードで動作したままであることを注意してください。そのため、PLL 出力を切り替えたという情報は他のデバイスへ伝播されていません。
7. PLL が安定するまで待ちます。PLL が安定するまでの時間については、各デバイスのデータ・マニュアルを参照してください。
8. PLLM に要求する逡倍器の値をプログラムします。
9. 必要に応じて、SYSCLK1、SYSCLK2 の各分周比の値を変更するために、PLLDIV1、PLLDIV2 の各レジスタをプログラムします。
  - a. GO 動作が現在進行中ではないことを示すために、PLLSTAT の GOSTAT ビットが 0 にクリアされているかチェックします。

- b. PLLDIV1、PLLDIV2 の各 RATIO フィールドを指定の分周値にプログラムします。PLL2 の場合、SYSCLK1 と SYSCLK2 間には特定の周波数比に関する要件はありません。このステップでは、PLLDIV1.D1EN、PLLDIV2.D2EN の各ビットを（デフォルトに）セットしたままにしておいてください。
  - c. 新たな分周比への遷移を開始するために、PLLCMD の GOSET ビットを 1 にセットします。この遷移中に、SYSCLK1、SYSCLK2 は一時的に動作を停止します。
  - d. 分周比の変更を確実に完了させるために、PLLDIV<sub>n</sub> ソース・クロックの N サイクル分待ちます。サイクル数 N の計算式については、5.3.2.4 項を参照してください。
  - e. PLLSTAT の GOSTAT ビットが 0 にクリアされるまで待ちます。
10. PLL が適切にリセットされるまで待ちます。PLL がリセットされるまでの時間については、各デバイスのデータ・マニュアルを参照してください。
  11. リセット状態から PLL モードへ切り替えるために、PLLCTL の PLLRST ビットを 1 にセットします。
  12. PLL がロックされるまで待ちます。PLL がロックされるまでの時間については、各デバイスのデータ・マニュアルを参照してください。
  13. バイパス・モードから PLL モードへ移行するために、PLLCTL の PLEN ビットを 1 にセットします。

DDR2 メモリ・コントローラの初期化については、『TMS320DM643x DMP DDR2 Memory Controller User's Guide』（資料番号 SPRU986）をご覧ください。

### 5.3.2.3 PLL 逡倍器の変更

PLL がパワー・ダウン状態ではなく（PLLCTL の PLLPWRDN ビットが 0 にクリアされていて）、かつ PLL が安定している状態の（5.3.2.2 項のステップ 7）場合、次の手順に従って PLL2 逡倍器を変更します。

1. PLL の周波数を変更する前に、PLL バイパス・モードに切り替えます。
  - a. PLLCTL.PLEN を有効にするために、PLLCTL の PLENSRC ビットを 0 にクリアします。
  - b. PLLCTL の PLEN ビットを 0 にクリアします（PLL バイパス・モードを選択します）。
  - c. PLLC が適切にバイパス・モードへ切り替わるために、4 MXI サイクル待ちます。
2. PLLCTL の PLLRST ビットを 0 にクリアします（PLL をリセットします）。
3. PLL 出力が切り替わって動作を開始できるようにするために、PLLCTL の PLLDIS ビットを 0 にクリアします（PLL をイネーブルします）。PLLC は PLL バイパス・モードで動作したままであることに注意してください。そのため、PLL 出力を切り替えたという情報は他のデバイスへ伝播されていません。
4. PLLM に要求する逡倍器の値をプログラムします。
5. 必要に応じて、SYSCLK1、SYSCLK2 の各分周比の値を変更するために、PLLDIV1、PLLDIV2 の各レジスタをプログラムします。
  - a. GO 動作が現在進行中ではないことを示すために、PLLSTAT の GOSTAT ビットが 0 にクリアされているかチェックします。
  - b. PLLDIV1、PLLDIV2 の各 RATIO フィールドを指定の分周値にプログラムします。PLL2 の場合、SYSCLK1 と SYSCLK2 間には特定の周波数比に関する要件はありません。このステップでは、PLLDIV1.D1EN、PLLDIV2.D2EN の各ビットを（デフォルトに）セットしたままにしておいてください。
  - c. 新たな分周比への遷移を開始するために、PLLCMD の GOSET ビットを 1 にセットします。この遷移中に、SYSCLK1、SYSCLK2 は一時的に動作を停止します。
  - d. 分周比の変更を確実に完了させるために、PLLDIV<sub>n</sub> ソース・クロックの N サイクル分待ちます。サイクル数 N の計算式については、5.3.2.4 項を参照してください。
  - e. PLLSTAT の GOSTAT ビットが 0 にクリアされるまで待ちます。

6. PLL が適切にリセットされるまで待ちます。PLL がリセットされるまでの時間については、各デバイスのデータ・マニュアルを参照してください。
7. リセット状態から PLL モードへ遷移するために、PLLCTL の PLLRST ビットを 1 にセットします。
8. PLL がロックされるまで待ちます。PLL がロックされるまでの時間については、各デバイスのデータ・マニュアルを参照してください。
9. バイパス・モードから PLL モードへ移行するために、PLLCTL の PLEN ビットを 1 にセットします。

### 5.3.2.4 SYSCLK の分周比の変更

ここでは、SYSCLK の分周比を変更するソフトウェア・シーケンスについて説明します。また、分周比の変更を開始するために、GO ビット (PLLCMD の GOSET ビット) の起動を伴うので、SYSCLK の分周比を変更するシーケンスは GO 動作ともいいます。

1. GO 動作が現在進行中ではないことを示すために、PLLSTAT の GOSTAT ビットが 0 にクリアされているかチェックします。
2. PLLDIV1、PLLDIV2 の各 RATIO フィールドを指定の分周値にプログラムします。PLLC2 の場合、SYSCLK1 と SYSCLK2 間には特定の周波数比に関する要件はありません。このステップでは、PLLDIV1.D1EN、PLLDIV2.D2EN の各ビットを (デフォルトに) セットしたままにしておいてください。
3. 新たな分周比への遷移を開始するために、PLLCMD の GOSET ビットを 1 にセットします。この遷移中に、SYSCLK1、SYSCLK2 は一時的に動作を停止します。
4. 分周比の変更を確実に完了させるために、PLLDIV<sub>n</sub> ソース・クロックの N サイクル分待ちます。サイクル数 N の計算式については、次の式を参照してください。
5. PLLSTAT の GOSTAT ビットが 0 にクリアされるまで待ちます。

PLLDIV<sub>n</sub> ソース・クロック・サイクル数を計算するには、次の式を使用してください。

$$N = (\text{設定前のすべての SYSCLK 分周値の最小公倍数 [LCM] の 2 倍}) + 50 \text{ サイクルのオーバーヘッド}$$

#### 例 5-2. クロック・サイクル数 N の計算

次に、クロック・サイクル数 N を計算する例を示します。

- 分周比変更前の設定値 :
  - PLLDIV1.RATIO = 1 (2 分周)
  - PLLDIV2.RATIO = 9 (10 分周)
- 分周比変更後の新しい設定値 :
  - PLLDIV1.RATIO = 1 (2 分周)
  - PLLDIV2.RATIO = 19 (20 分周)

設定前の古い分周比の値 /2 と /10 の最小公倍数は /10 になるため、サイクル数 N を求める計算式は次のようになります。

$$N = (2 \times 10) + 50 \text{ サイクルのオーバーヘッド} = 70 \text{ PLLDIV}_n \text{ ソース・クロック・サイクル}$$

PLLC2 が PLL モード (PLLCTL.PLEN = 1) の場合、PLLDIV<sub>n</sub> ソース・クロックは PLL2 出力クロックです。PLLC2 が PLL バイパス・モード (PLLCTL.PLEN = 0) の場合、PLLDIV<sub>n</sub> ソース・クロックはデバイス・クロック・ソース MXI/CLKIN です。

## 5.4 PLL コントローラのレジスタ

表 5-3 に、PLL コントローラのベース・アドレスおよびエンド・アドレスを示します。表 5-4 に、PLL およびリセット・コントローラのメモリ・マップド・レジスタを示します。これらのレジスタのメモリ・アドレスについては、各デバイスのデータ・マニュアルを参照してください。

**表 5-3. PLL およびリセット・コントローラ一覧**

PLL およびリセット・コントローラ	ベース・アドレス	エンド・アドレス	サイズ
PLLC1	1C4 0800h	1C4 0BFFh	400h
PLLC2	1C4 0C00h	1C4 0FFFh	400h

**表 5-4. PLL およびリセット・コントローラのレジスタ**

オフセット	略称	レジスタの説明	参照先
00h	PID	ペリフェラル ID レジスタ	5.4.1 項
E4h	RSTYPE <sup>(1)</sup>	リセット・タイプ・ステータス・レジスタ	5.4.2 項
100h	PLLCTL	PLL コントロール・レジスタ	5.4.3 項
110h	PLLM	PLL マルチプライア・コントロール・レジスタ	5.4.4 項
118h	PLLDIV1	PLL コントローラ・ディバイダ 1 レジスタ (SYSCLK1)	5.4.5 項
11Ch	PLLDIV2	PLL コントローラ・ディバイダ 2 レジスタ (SYSCLK2)	5.4.6 項
120h	PLLDIV3 <sup>(1)</sup>	PLL コントローラ・ディバイダ 3 レジスタ (SYSCLK3)	5.4.7 項
124h	OSCDIV1 <sup>(1)</sup>	オシレータ・ディバイダ 1 レジスタ (OBSCLK)	5.4.8 項
12Ch	BPDIV	バイパス・ディバイダ・レジスタ	5.4.9 項
138h	PLLCMD	PLL コントローラ・コマンド・レジスタ	5.4.10 項
13Ch	PLLSTAT	PLL コントローラ・ステータス・レジスタ	5.4.11 項
140h	ALNCTL	PLL コントローラ・クロック・アライン・コントロール・レジスタ	5.4.12 項
144h	DCHANGE	PLLDIV レシオ・チェンジ・ステータス・レジスタ	5.4.13 項
148h	CKEN <sup>(1)</sup>	クロック・イネーブル・コントロール・レジスタ	5.4.14 項
14Ch	CKSTAT	クロック・ステータス・レジスタ	5.4.15 項
150h	SYSTAT	SYSCLK ステータス・レジスタ	5.4.16 項

<sup>(1)</sup> PLL2 ではサポートされていません。

### 5.4.1 ペリフェラル ID レジスタ (PID)

ペリフェラル ID レジスタ (PID) を図 5-3 に示し、表 5-5 で説明します。

図 5-3. ペリフェラル ID レジスタ (PID)

31	24	23	16
Reserved		TYPE	
R-0		R-1h	
15	8	7	0
CLASS		REV	
R-8h		R-Dh	

凡例：R = リード専用。-n = リセット後の値。

表 5-5. ペリフェラル・アイデンティフィケーション・レジスタ (PID) フィールドの説明

ビット	フィールド	値	説明
31-24	Reserved	0	予約。
23-16	TYPE	1h	ペリフェラル・タイプ。 PLL
15-8	CLASS	8h	ペリフェラル・クラス。 現在のクラス
7-0	REV	Dh	ペリフェラル・リビジョン。 現在のバージョン

### 5.4.2 リセット・タイプ・ステータス・レジスタ (RSTYPE)

リセット・タイプ・ステータス・レジスタ (RSTYPE) を図 5-4 に示し、表 5-6 で説明します。これは最後のリセット要因をラッチします。リセット状態が解除されるとすべてのビットのリセット値は 0 ですが、リセットの要因を示すために 1 つのビットが 1 にセットされます。

図 5-4. リセット・タイプ・ステータス・レジスタ (RSTYPE)

31					16		
Reserved							
R-0							
15				3	2	1	0
Reserved			MRST	XWRST	POR		
R-0			R-0	R-0	R-0		

凡例：R = リード専用。-n = リセット後の値。

表 5-6. リセット・タイプ・ステータス・レジスタ (RSTYPE) フィールドの説明

ビット	フィールド	値	説明
31-3	Reserved	0	予約。
2	MRST	0-1	マックス・リセット。このビットが 1 の場合、リセットの原因は優先順位が最上位のマックス・リセット。
1	XWRST	0-1	外部ウォーム・リセット。このビットが 1 の場合、リセットの最後の原因は優先順位が最上位のウォーム・リセット (RESET)。
0	POR	0-1	パワー・オン・リセット。このビットが 1 の場合、リセットの最後の原因は優先順位が最上位のパワー・オン・リセット (POR)。

### 5.4.3 PLL コントロール・レジスタ (PLLCTL)

PLL コントロール・レジスタ (PLLCTL) を図 5-5 に示し、表 5-7 で説明します。

図 5-5. PLL コントロール・レジスタ (PLLCTL)

31	Reserved										16
R-0											
15	9	8	7	6	5	4	3	2	1	0	
	Reserved	CLKMODE	Reserved	PllenSRC	PLLDIS	PLLRS	Rsvd	PLLWRDN	Pllen		
	R-0	R/W-0	R-1h	R/W-1	R/W-1	R/W-0	R-0	R/W-1	R/W-0		

凡例：R/W = リード / ライト。R = リード専用。-n = リセット後の値。

表 5-7. PLL コントロール・レジスタ (PLLCTL) フィールドの説明

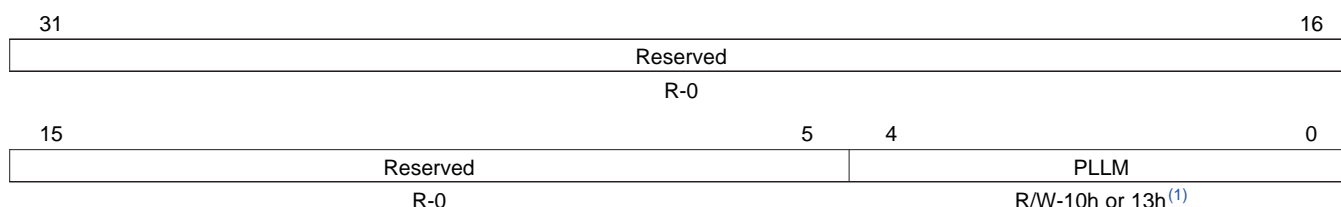
ビット	フィールド	値	説明
31-9	Reserved	0	予約。
8	CLKMODE	0	リファレンス・クロックの選択。 内部発振器。デバイス・リファレンス・クロック・ソースが MXI/CLKIN ピンの発振子の場合、クロック・ソースとして内部発振器を選択する必要があります。
		1	CLKIN 矩形波。このモードが適用されるのは、デバイス・リファレンス・クロック・ソースが MXI/CLKIN ピンで矩形波の場合です。このモードが選択されると、PLL は節電するために内部発振器のバイアス抵抗をオフにします。
7-6	Reserved	1	予約。
5	PllenSRC	0	Pllen を有効にする前に、このビットを 0 にクリアする必要があります。
4	PLLDIS	0	PLL ディスエーブルのアサート。
		1	PLL ディスエーブルのデアサート。PLL 出力はディスエーブルされ、切り換わりません。
3	PLLRS	0	サポートされている場合、PLL リセットをアサートします。
		1	PLL リセットのデアサート。
2	Reserved	0	予約。
1	PLLWRDN	0	PLL のパワー・ダウン。PLL のパワー・アップ (PLLWRDN ビットが 1 から 0 への遷移) 後、PLL が安定するまで待つ必要があります。PLL が安定するまでの時間については、各デバイスのデータ・マニュアルを参照してください。
		1	PLL は動作中。
		1	PLL はパワー・ダウン。
0	Pllen	0	PLL モードのイネーブル。
		0	バイパス・モード
		1	PLL モードです。バイパスされません。



#### 5.4.4 PLL マルチプライア・コントロール・レジスタ (PLLM)

PLL マルチプライア・コントロール・レジスタ (PLLM) を図 5-6 に示し、表 5-8 で説明します。

図 5-6. PLL マルチプライア・コントロール・レジスタ (PLLM)



凡例：R/W = リード / ライト。R = リード専用。-n = リセット後の値。

<sup>(1)</sup> PLLC1 の場合、PLLM のデフォルトは 10h (PLL1 × 17)。PLLC2 の場合、PLLM のデフォルトは 13h (PLL2 × 20)。

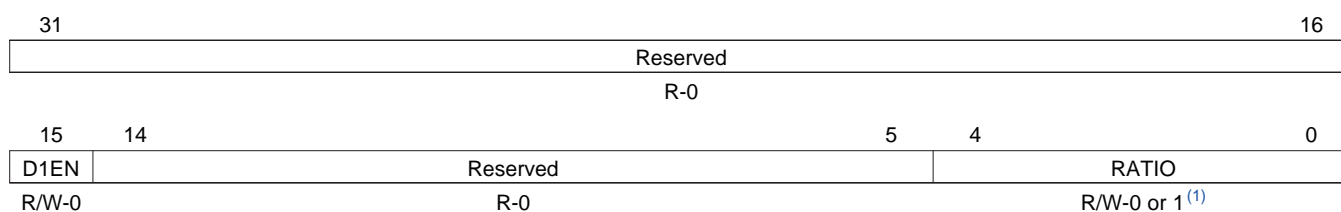
表 5-8. PLL マルチプライア・コントロール・レジスタ (PLLM) フィールドの説明

ビット	フィールド	値	説明
31-5	Reserved	0	予約。
4-0	PLLM	0 ~ 1Fh	PLL マルチプライアの選択。通倍値 = PLLM + 1。たとえば、PLLM = 16 (10h) は 17 の通倍を表します。PLL ごとの有効な通倍値については、各デバイスのデータ・マニュアルを参照してください。

#### 5.4.5 PLL コントローラ・ディバイダ 1 レジスタ (PLLDIV1)

PLL コントローラ・ディバイダ 1 レジスタ (PLLDIV1) を図 5-7 に示し、表 5-9 で説明します。分周器 1 は SYSCLK1 の分周を制御します。

図 5-7. PLL コントローラ・ディバイダ 1 レジスタ (PLLDIV1)



凡例：R/W = リード / ライト。R = リード専用。-n = リセット後の値。

<sup>(1)</sup> PLLC1 の場合、RATIO のデフォルトは 0 (PLL1 の 1 分周)。PLLC2 の場合、RATIO のデフォルトは 1 (PLL2 の 2 分周)。

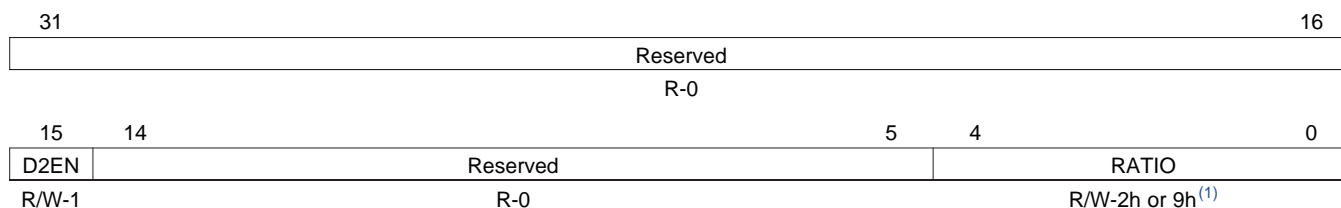
表 5-9. PLL コントローラ・ディバイダ 1 レジスタ (PLLDIV1) フィールドの説明

ビット	フィールド	値	説明
31-16	Reserved	0	予約。
15	DIEN	0 1	分周器 1 のイネーブル。 0 分周器 1 はディスエーブル。 1 分周器 1 はイネーブル。
14-5	Reserved	0	予約。
4-0	RATIO	0 ~ 1Fh	分周比。分周比の値 = RATIO + 1。たとえば、RATIO = 0 は 1 分周を表します。

### 5.4.6 PLL コントローラ・ディバイダ 2 レジスタ (PLLDIV2)

PLL コントローラ・ディバイダ 2 レジスタ (PLLDIV2) を図 5-8 に示し、表 5-10 で説明します。分周器 2 は SYSCLK2 の分周を制御します。

図 5-8. PLL コントローラ・ディバイダ 2 レジスタ (PLLDIV2)



凡例：R/W = リード / ライト。R = リード専用。-n = リセット後の値。

<sup>(1)</sup> PLLC1 の場合、RATIO のデフォルトは 2h (PLL1 の 3 分周)。PLLC2 の場合、RATIO のデフォルトは 9h (PLL2 の 10 分周)。

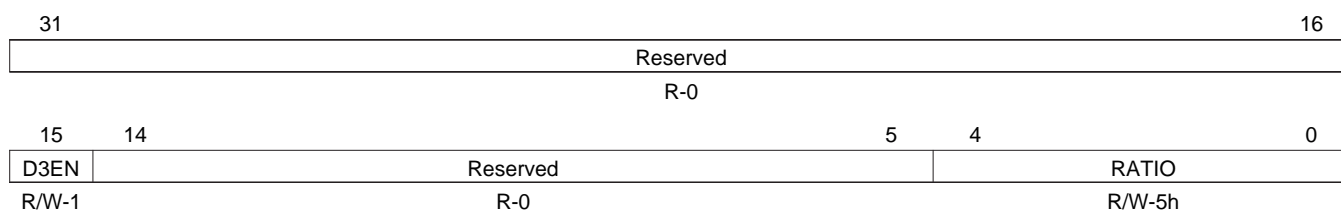
表 5-10. PLL コントローラ・ディバイダ 2 レジスタ (PLLDIV2) フィールドの説明

ビット	フィールド	値	説明
31-16	Reserved	0	予約。
15	D2EN	0	分周器 2 のイネーブル。 分周器 2 はディスエーブル。
		1	分周器 2 はイネーブル。
14-5	Reserved	0	予約。
4-0	RATIO	0 ~ 1Fh	分周比。分周比の値 = RATIO + 1。たとえば、RATIO = 0 は 1 分周を表します。

### 5.4.7 PLL コントローラ・ディバイダ 3 レジスタ (PLLDIV3)

PLL コントローラ・ディバイダ 3 レジスタ (PLLDIV3) を図 5-9 に示し、表 5-11 で説明します。分周器 3 は SYSCLK3 の分周を制御します。PLLDIV3 は、PLLC2 では使われません。

図 5-9. PLL コントローラ・ディバイダ 3 レジスタ (PLLDIV3)



凡例：R/W = リード / ライト。R = リード専用。-n = リセット後の値。

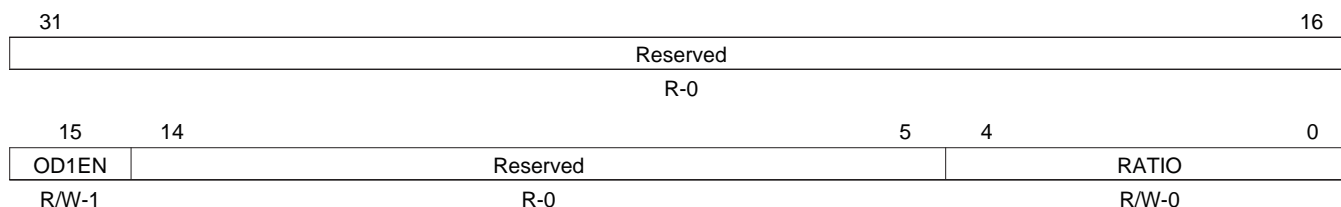
表 5-11. PLL コントローラ・ディバイダ 3 レジスタ (PLLDIV3) フィールドの説明

ビット	フィールド	値	説明
31-16	Reserved	0	予約。
15	D3EN	0	分周器 3 のイネーブル。 分周器 3 はディスエーブル。
		1	分周器 3 はイネーブル。
14-5	Reserved	0	予約。
4-0	RATIO	0 ~ 1Fh	分周比。分周比の値 = RATIO + 1。たとえば、RATIO = 0 は 1 分周を表します。

### 5.4.8 オシレータ・ディバイダ1レジスタ (OSCDIV1)

オシレータ・ディバイダ1レジスタ(OSCDIV1)を図 5-10 に示し、表 5-12 で説明します。オシレータ分周器1はOBSCLKの分周を制御し、MXI/CLKIN クロックから分周を行います。PLL1の場合、OBSCLKはCLKOUT0ピンに接続されます。OSCDIV1はPLL1にのみ適用され、PLL2には適用されません。

図 5-10. オシレータ・ディバイダ1レジスタ (OSCDIV1)



凡例：R/W = リード/ライト。R = リード専用。-n = リセット後の値。

表 5-12. オシレータ・ディバイダ1レジスタ (OSCDIV1) フィールドの説明

ビット	フィールド	値	説明
31-16	Reserved	0	予約。
15	OD1EN	0 1	オシレータ分周器1のイネーブル。 0 オシレータ分周器1はディスエーブル。 1 オシレータ分周器1はイネーブル。OBSCLKを切り替えるためには、クロック・イネーブル・コントロール・レジスタ (CKEN) の OD1EN ビットと OBSEN ビットの両方を1にセットする必要があります。
14-5	Reserved	0	予約。
4-0	RATIO	0 ~ 1Fh	分周比。分周比の値 = RATIO + 1。たとえば、RATIO = 0 は1分周を表します。

### 5.4.9 バイパス・ディバイダ・レジスタ (BPDIV)

バイパス・ディバイダ・レジスタ (BPDIV) を図 5-11 に示し、表 5-13 で説明します。バイパス・ディバイダは SYSCLKBP の分周比を制御し、MXI/CLKIN クロックから分周を行います。

図 5-11. バイパス・ディバイダ・レジスタ (BPDIV)

31	Reserved										16	
R-0												
15	14						5	4				0
BPDEN	Reserved						RATIO					
R/W-1	R-0						R/W-0 or 1 <sup>(1)</sup>					

凡例：R/W = リード / ライト。R = リード専用。-n = リセット後の値。

<sup>(1)</sup> PLLC1 の場合、RATIO のデフォルトは 0 (MXI/CLKIN の 1 分周)。PLLC2 の場合、RATIO のデフォルトは 1 (MXI/CLKIN の 2 分周)。

表 5-13. バイパス・ディバイダ・レジスタ (BPDIV) フィールドの説明

ビット	フィールド	値	説明
31-16	Reserved	0	予約。
15	BPDEN	0	バイパス分周器のイネーブル。
		1	バイパス分周器はディスエーブル。
14-5	Reserved	0	予約。
4-0	RATIO	0 ~ 1Fh	分周比。分周比の値 = RATIO + 1。たとえば、RATIO = 0 は 1 分周を表します。

### 5.4.10 PLL コントローラ・コマンド・レジスタ (PLLCMD)

PLL コントローラ・コマンド・レジスタ (PLLCMD) を図 5-12 に示し、表 5-14 で説明します。PLLCMD には、GO 動作用のコマンド・ビットが含まれています。このビットに 1 をライトすると、コマンドの実行を開始します。このビットに 0 をライトすると、ビットがクリアされますが、影響はありません。

図 5-12. PLL コントローラ・コマンド・レジスタ (PLLCMD)

31	Reserved										16
R-0											
15										1	0
Reserved										GOSET	
R-0										R/W-0	

凡例：R/W = リード / ライト。R = リード専用。-n = リセット後の値。

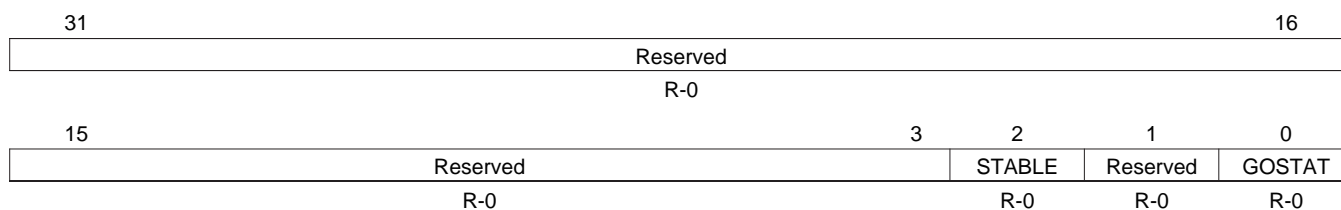
表 5-14. PLL コントローラ・コマンド・レジスタ (PLLCMD) フィールドの説明

ビット	フィールド	値	説明
31-1	Reserved	0	予約。
0	GOSET	0	SYSCLKx による新たな分周および位相アライメントのロードを行う GO ビット。
		1	ビットをクリア (影響なし)。 SYSCLKx 位相アライメントを開始します。

### 5.4.11 PLL コントローラ・ステータス・レジスタ (PLLSTAT)

PLL コントローラ・ステータス・レジスタ (PLLSTAT) を図 5-13 に示し、表 5-15 で説明します。

図 5-13. PLL コントローラ・ステータス・レジスタ (PLLSTAT)



凡例：R = リード専用。-n = リセット後の値。

表 5-15. PLL コントローラ・ステータス・レジスタ (PLLSTAT) フィールドの説明

ビット	フィールド	値	説明
31-3	Reserved	0	予約。
2	STABLE	0 1	OSC カウンタの終了。発振器が安定していることを想定。デバイスがリセット状態から解除される前に、このビットを 1 にセットしてください。 いいえ はい
1	Reserved	0	予約。
0	GOSTAT	0 1	GO 動作のステータス。 GO 動作は進行していません。 GO 動作は進行中。

### 5.4.12 PLL コントローラ・クロック・アライン・コントロール・レジスタ (ALNCTL)

PLL コントローラ・クロック・アライン・コントロール・レジスタ (ALNCTL) を図 5-14 に示し、表 5-16 で説明します。ALNCTL は適切にデバイスを動作させるためにアラインが必要な SYSCLK を示します。ALNCTL のデフォルト設定値を変更してはいけません。

図 5-14. PLL コントローラ・クロック・アライン・コントロール・レジスタ (ALNCTL)

31	Reserved				16		
R-0							
15	Reserved			3	2	1	0
R/W-0h or 3h <sup>(1)</sup>				ALN3	ALN2	ALN1	
				R/W-0 or 1 <sup>(2)</sup>	R/W-0 or 1 <sup>(3)</sup>	R/W-0 or 1 <sup>(4)</sup>	

凡例：R/W = リード / ライト。R = リード専用。-n = リセット後の値。

- (1) PLLC1 の場合、この予約フィールドのデフォルトは 3h です。PLLC2 の場合、この予約フィールドのデフォルトは 0h です。このデフォルト値を変更してはいけません。
- (2) PLLC1 の場合、ALN3 のデフォルトは 1 です。PLLC2 の場合、ALN3 は予約されており、そのデフォルトは 0 です。
- (3) PLLC1 の場合、ALN2 のデフォルトは 1 です。PLLC2 の場合、ALN2 のデフォルトは 0 です。
- (4) PLLC1 の場合、ALN1 のデフォルトは 1 です。PLLC2 の場合、ALN1 のデフォルトは 0 です。

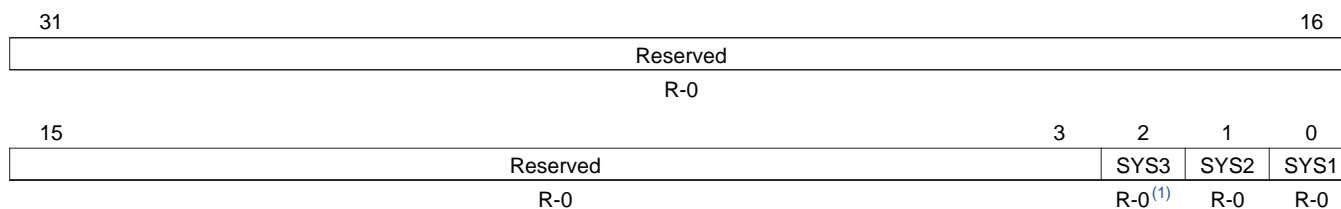
表 5-16. PLL コントローラ・クロック・アライン・コントロール・レジスタ (ALNCTL) フィールドの説明

ビット	フィールド	値	説明
31-3	Reserved	0 または 3	予約。このデフォルト値を変更してはいけません。
2	ALN3	0	SYSCLK3 は、このレジスタで選択された他の SYSCLK とアラインする必要があります。ただし、これは PLLC2 には適用されません (このビットは予約されています)。
		1	SYSCLK3 をアラインする必要はありません。
1	ALN2	0	SYSCLK2 は、このレジスタで選択された他の SYSCLK とアラインする必要があります。
		1	SYSCLK2 をアラインする必要はありません。
0	ALN1	0	SYSCLK1 は、このレジスタで選択された他の SYSCLK とアラインする必要があります。
		1	SYSCLK1 をアラインする必要はありません。

### 5.4.13 PLLDIV レシオ・チェンジ・ステータス・レジスタ (DCHANGE)

PLLDIV レシオ・チェンジ・ステータス・レジスタ (DCHANGE) を図 5-15 に示し、表 5-17 で説明します。DCHANGE は、SYSCLK の分周比が変更されたかどうかを示します。

図 5-15. PLLDIV レシオ・チェンジ・ステータス・レジスタ (DCHANGE)



凡例：R = リード専用。-n = リセット後の値。

<sup>(1)</sup> PLLC2 の場合、SYS3 は予約されており、そのデフォルトは 0 です。

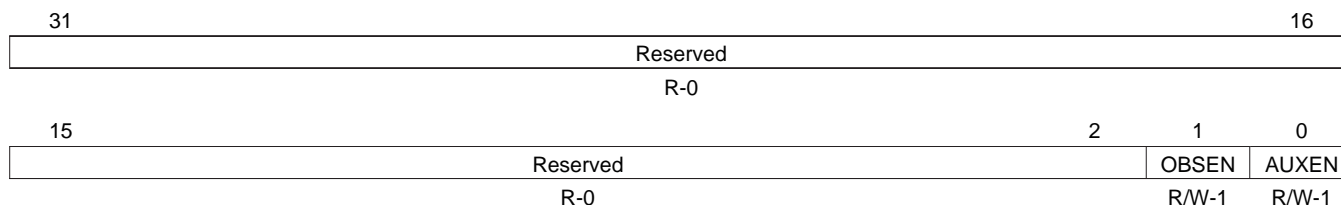
表 5-17. PLLDIV レシオ・チェンジ・ステータス・レジスタ (DCHANGE) フィールドの説明

ビット	フィールド	値	説明
31-3	Reserved	0	予約。
2	SYS3	0	SYSCLK3 の分周比の変更。ただし、これは PLLC2 には適用されません (このビットは予約されています)。
		1	SYSCLK3 の分周比は変更されています。
1	SYS2	0	SYSCLK2 の分周比は変更されていません。
		1	SYSCLK2 の分周比は変更されています。
0	SYS1	0	SYSCLK1 の分周比は変更されていません。
		1	SYSCLK1 の分周比は変更されています。

#### 5.4.14 クロック・イネーブル・コントロール・レジスタ (CKEN)

クロック・イネーブル・コントロール・レジスタ (CKEN) を図 5-16 に示し、表 5-18 で説明します。CKEN は、さまざまな出力クロックに対するクロック・イネーブル制御を行います。CKEN は PLLC1 にのみ適用され、PLLC2 には適用されません。

図 5-16. クロック・イネーブル・コントロール・レジスタ (CKEN)



凡例：R/W = リード / ライト。R = リード専用。-n = リセット後の値。

表 5-18. クロック・イネーブル・コントロール・レジスタ (CKEN) の説明

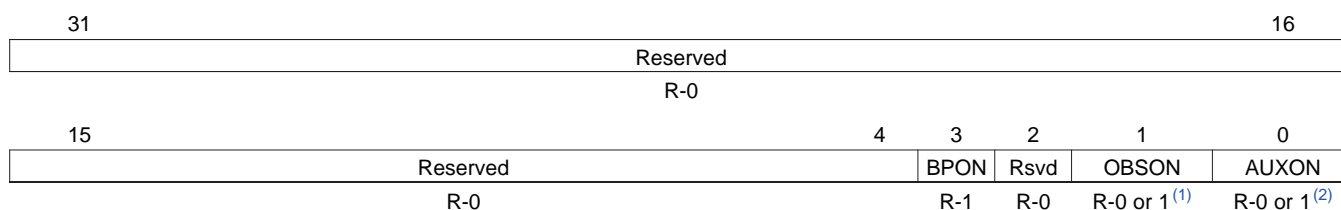
ビット	フィールド	値	説明
31-2	Reserved	0	予約。
1	OBSEN	0	OBSCCLK はディスエーブルです。
		1	OBSCCLK はイネーブルです。OBSCCLK を切り替えるためには、オシレータ・ディバイダ 1 レジスタ (OSCDIV1) の OBSEN ビットと OD1EN ビットの両方を 1 にセットする必要があります。
0	AUXEN	0	AUXCLK はディスエーブルです。
		1	AUXCLK はイネーブルです。



### 5.4.15 クロック・ステータス・レジスタ (CKSTAT)

クロック・ステータス・レジスタ (CKSTAT) を図 5-17 に示し、表 5-19 で説明します。CKSTAT は、SYSCLK $n$  を除くすべてのクロックのクロック・ステータスを示します。

図 5-17. クロック・ステータス・レジスタ (CKSTAT)



凡例：R = リード専用。-n = リセット後の値。

(1) PLLC1 の場合、OBSON のデフォルトは 1 です。PLLC2 の場合、OBSON は予約されており、そのデフォルトは 0 です。

(2) PLLC1 の場合、AUXON のデフォルトは 1 です。PLLC2 の場合、AUXON は予約されており、そのデフォルトは 0 です。

表 5-19. クロック・ステータス・レジスタ (CKSTAT) フィールドの説明

ビット	フィールド	値	説明
31-4	Reserved	0	予約。
3	BPON	0	SYSCLKBP がオンであることを示すステータス。SYSCLKBP は、バイパス・ディバイダ・レジスタ (BPDIV) で制御されます。
		1	SYSCLKBP はオフ。
2	Reserved	0	予約。
1	OBSON	0	OBSCCLK がオンであることを示すステータス。OBSCCLK は、オシレータ・ディバイダ 1 レジスタ (OSCDIV1) のビットおよびクロック・イネーブル・コントロール・レジスタ (CKEN) の OBSEN ビットにより制御されます。ただし、これは PLLC2 には適用されません (このビットは予約されています)。
		1	OBSCCLK はオン。
0	AUXON	0	AUXCLK がオンであることを示すステータス。AUXCLK は、クロック・イネーブル・コントロール・レジスタ (CKEN) の AUXEN ビットにより制御されます。ただし、これは PLLC2 には適用されません (このビットは予約されています)。
		1	AUXCLK はオン。

### 5.4.16 SYSCLK ステータス・レジスタ (SYSTAT)

SYSCLK ステータス・レジスタ (SYSTAT) の内容と説明をそれぞれ図 5-18 と表 5-20 に示します。SYSCLK のオン / オフのステータスを示します。実際のデフォルトは、実際のクロックのオン / オフのステータスにより決定され、デフォルトの PLLDIV[n] の D[n]EN ビットにより異なります。

図 5-18. SYSCLK ステータス・レジスタ (SYSTAT)

31	Reserved	16
	R-0	
15	Reserved	3      2      1      0
	R-0	R-0 or 1 <sup>(1)</sup> R-1      R-1
		SYS3ON      SYS2ON      SYS1ON

凡例：R = リード専用。-n = リセット後の値。

<sup>(1)</sup> PLLC1 の場合、SYS3ON のデフォルトは 1 です。PLLC2 の場合、SYS3ON は予約されており、そのデフォルトは 0 です。

表 5-20. SYSCLK ステータス・レジスタ (SYSTAT) フィールドの説明

ビット	フィールド	値	説明
31-3	Reserved	0	予約。
2	SYS3ON	0 1	SYSCLK3 がオンであるか示すステータス。SYSCLK3 は、PLL コントローラ・ディバイダ 3 レジスタ (PLLDIV3) のビットで制御されます。ただし、これは PLLC2 には適用されません (このビットは予約されています)。 SYSCLK3 はオフ。 SYSCLK3 はオン。
1	SYS2ON	0 1	SYSCLK2 がオンであるか示すステータス。SYSCLK2 は、PLL コントローラ・ディバイダ 2 レジスタ (PLLDIV2) のビットで制御されます。 SYSCLK2 はオフ。 SYSCLK2 はオン。
0	SYS1ON	0 1	SYSCLK1 がオンであるか示すステータス。SYSCLK1 は、PLL コントローラ・ディバイダ 1 レジスタ (PLLDIV1) のビットで制御されます。 SYSCLK1 はオフ。 SYSCLK1 はオン。

## パワー・スリープ・コントローラ

項目	ページ
6.1 はじめに .....	60
6.2 パワー・ドメインおよびモジュール・トポロジ .....	61
6.3 パワー・ドメインおよびモジュール・ステート .....	62
6.4 ステート遷移の実行 .....	63
6.5 PSC での PSC エミュレーション・サポート .....	64
6.6 PSC 割り込み .....	64
6.7 PSC レジスタ .....	66

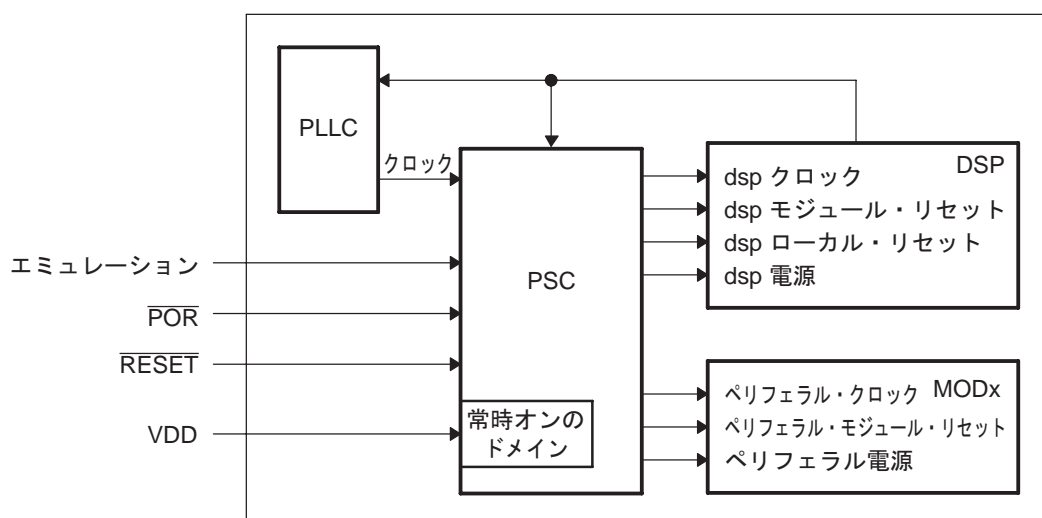
## 6.1 はじめに

パワー・スリープ・コントローラ (PSC) は、システム・パワーのオン/オフ、クロックのオン/オフ、およびリセットの遷移を管理します。DM643x DMP は、消費電力を抑制するために PSC のクロック・ゲート制御機能のみを利用します。PSC は、グローバル PSC (GPSC) および一連のローカル PSC (LPSC) から構成されています。GPSC には、ペリフェラルまたはモジュールごとにメモリ・マップド・レジスタ、PSC 割り込みコントロール、およびステート・マシンが含まれています。LPSC には、ペリフェラルまたはモジュール個々に関連付けられており、クロックおよびリセットを制御します。図 6-1 に、デバイスに内蔵されている PSC の様子を示します。パワー・オン・リセットやハードウェア・リセットなど PSC の動作は、ソフトウェアからは見えないものが多いです。ただし、PSC にはいくつかの重要な電源、クロック、リセットの動作を制御するインターフェイスが用意されています。本章では、電源、クロック、リセットの動作を中心に説明します。

PSC に組み込まれている機能は次のとおりです。

- チップでのパワー・オン/オフおよびリセットの管理
- 次の機能を実現するソフトウェア・インターフェイスの提供
  - モジュール・クロックのオン/オフ
  - モジュール・リセットの制御
  - DSP によるローカル・リセットの制御 (CPU リセット)
- IcePick エミュレーション機能のサポート (電源、クロック、リセット)

図 6-1. パワー・スリープ・コントローラ (PSC) 集積回路



**注:** DSP ローカル・リセットおよび DSP モジュール・リセットの影響は、十分な検証を行っていません。したがって、これらのリセットはサポートされていないので使用してはいけません。その代わりに、POR または RESET ピンを使用して DSP 全体をリセットしてください。

## 6.2 パワー・ドメインおよびモジュール・トポロジ

DM643x DMP には、AlwaysOn パワー・ドメインというパワー・ドメインが 1 つ組み込まれています。AlwaysOn パワー・ドメインは、チップへの電源供給がオンの場合常にオンの状態になります。AlwaysOn ドメインは、DM643x DMP の  $V_{DD}$  ピンから電源が供給されます (各デバイスのデータ・マニュアルを参照)。DM643x DMP モジュールはすべて、AlwaysOn パワー・ドメイン内にあります。表 6-1 に、DM643x DMP に搭載されているペリフェラル、LPSC 割り当て、およびデフォルトのモジュール・ステートについて示します。指定デバイスで使用可能なペリフェラルについては、各デバイスのデータ・マニュアルを参照してください。デフォルトのモジュール・ステートについては、6.3.2 項で定義します。

表 6-1. DM643x DMP モジュールのデフォルトのコンフィギュレーション

LPSC 番号	モジュール名	デフォルトのモジュール・ステート (MDSTAT.STATE)
0	VPSS (マスタ)	SwRstDisable
1	VPSS (スレーブ)	SwRstDisable
2	EDMACC	SwRstDisable
3	EDMATC0	SwRstDisable
4	EDMATC1	SwRstDisable
5	EDMATC2	SwRstDisable
6	EMAC メモリ・コントローラ	SwRstDisable
7	MDIO	SwRstDisable
8	EMAC	SwRstDisable
9	McASP0	SwRstDisable
10	予約	-
11	VLYNQ	SwRstDisable
12	HPI	SwRstDisable
13	DDR2 メモリ・コントローラ	SwRstDisable
14	EMIFA	SwRstDisable (コンフィギュレーション・ピン AEM[2:0] = 000b の場合) イネーブル (コンフィギュレーション・ピン AEM[2:0] = それ以外の場合)
15	PCI	SwRstDisable
16	McBSP0	SwRstDisable
17	McBSP1	SwRstDisable
18	I2C	SwRstDisable
19	UART0	SwRstDisable
20	UART1	SwRstDisable
21	予約	SwRstDisable <sup>(1)</sup>
22	HECC	SwRstDisable
23	PWM0	SwRstDisable
24	PWM1	SwRstDisable
25	PWM2	SwRstDisable
26	GPIO	SwRstDisable
27	TIMER0	SwRstDisable
28	TIMER1	SwRstDisable
29-38	予約	-
39	C64x+ CPU	イネーブル
40	予約	-

<sup>(1)</sup> この予約ドメインの場合、モジュール・ステータス  $n$  レジスタ (MDSTAT0 ~ MDSTAT39) の対応する STATE ビットをディスエーブルにセットしないことが重要です。MDSTAT $n$  および STATE ビットの詳細については、6.7.9 項を参照してください。

## 6.3 パワー・ドメインおよびモジュール・ステート

**注：** DSP ローカル・リセットおよび DSP モジュール・リセットの影響は、十分な検証を行っていません。したがって、これらのリセットはサポートされていないので使用してはいけません。その代わりに、POR または RESET ピンを使用して DSP 全体をリセットしてください。

表 6-1 に、チップのパワー・オン・リセット ( $\overline{\text{POR}}$ )、ウォーム・リセット ( $\overline{\text{RESET}}$ )、マックス・リセットのいずれかを実行後の各モジュールのステートを示します。これらのステートについては、これ以降で定義します。

### 6.3.1 パワー・ドメインのステート

パワー・ドメインのステートは、オンかオフのいずれかの状態にのみなります。その定義は次のとおりです。

- ・ オン：パワー・ドメインへの電源供給がオン。
- ・ オフ：パワー・ドメインへの電源供給がオフ。

DM643x DMP では、AlwaysOn パワー・ドメインは、チップへの電源供給がオンの場合、常にオンの状態になります。

### 6.3.2 モジュール・ステート

モジュールのステートは、ディスエーブル、イネーブル、SyncReset、SwRstDisable の 4 つのいずれかになります。これらの 4 つのステートで、アサートされたまたはディアサートされたモジュール・リセットとモジュール・クロックのオン/オフの組み合わせに対応します (表 6-2 を参照)。

表 6-2. モジュール・ステート

モジュール・ステート	モジュール・リセット	モジュール・クロック	モジュール・ステートの定義
イネーブル	ディアサート	オン	イネーブル・ステートのモジュールではモジュール・リセットがディアサートされ、クロックはオンになっています。これは、特定のモジュールに対する通常の実行時のステートです。
ディスエーブル	ディアサート	オフ	ディスエーブル・ステートのモジュールではモジュール・リセットがディアサートされ、クロックはオフに設定されています。このステートは通常、節電を実現するモジュール・クロックをディスエーブルするために使われます。DM643x DMP は完全なスタティック CMOS で設計されているため、モジュール・クロックを停止した場合、モジュールのステートを保持しています。クロックが再起動されると、モジュールは停止した時点から動作を再開します。
SyncReset	アサート	オン	SyncReset ステートのモジュールではモジュール・リセットがアサートされ、クロックはオンに設定されています。一般に、ソフトウェアでこのステートを開始することは想定されていません。
SwRstDisable	アサート	オフ	SwResetDisable ステートのモジュールではモジュール・リセットがアサートされ、クロックはオフに設定されます。初期段階での電源投入後に、ほとんどのモジュールはデフォルトで SyncRst ステートになります (表 6-1 を参照)。一般に、ソフトウェアでこのステートを開始することは想定されていません。

**注：** モジュール・リセットは、特定のモジュールを完全にリセットすることと規定されているため、すべてのハードウェアはデフォルト・ステートに復帰します。モジュール・リセットの詳細については、第 10 章を参照してください。

パワー・マネージメントの詳細については、第 7 章を参照してください。

### 6.3.3 ローカル・リセット

モジュール・リセット (6.3.2 項を参照) に加え、特別なローカル・リセットを使用することで、DSP CPU をリセット状態にすることができます。DSP ローカル・リセットがアサートされた場合でも、DSP の内部メモリ (L1P、L1D、L2) にはアクセスできます。ローカル・リセットによってリセットされるのは、DSP CPU コアのみで、他の DSP サブシステムはリセットされません。これは DSP モジュールによるリセットと同様です。

モジュール・リセットは、ローカル・リセットより優先されます。したがって、ローカル・リセットは DSP が SyncReset または SwRstDisable ステートのときには有効とはなりません。

ローカル・リセットおよびこのローカル・リセットを使用する状況の詳細については、第 10 章を参照してください。

DSP ローカル・リセットをアサートまたはディアサートする手順は、次のとおりです。

1. MDCTL39 の LRST ビットを 0 にクリアします (DSP ローカル・リセットをアサートします)。
2. MDCTL39 の LRST ビットを 1 にセットします (DSP ローカル・リセットをディアサートします)。DSP がイネーブ  
ル・ステートの場合、リセットがディアサートされた直後にプログラム命令を実行します。

## 6.4 ステート遷移の実行

ここでは、デバイス・モジュールのステート遷移を実行する方法について説明します。

### 6.4.1 パワー・ダウンのステート遷移

DM643x DMP には、AlwaysOn パワー・ドメインというパワー・ドメインが 1 つ組み込まれています。AlwaysOn パワー・ドメインは、チップへの電源供給がオンの場合、常にオンの状態になります。このパワー・ドメインのステートをオフに変更することはできません。

### 6.4.2 モジュールのステート遷移

ここでは、モジュールのステート遷移手順について説明します。すべての DM643x DMP モジュールは、AlwaysOn ドメインにあります (パワー・ドメイン 0)。

一部のペリフェラルには特別なプログラミング要件があるため、PSC モジュールのステート遷移を行う前に処理すべき手順があることに注意してください。詳細については、各ペリフェラルのリファレンス・ガイドを参照してください。たとえば、DDR2 メモリ・コントローラにおいて、メモリ内容を保持する場合、PSC モジュールのステート遷移を行う前に、最初に DDR メモリをセルフ・リフレッシュ・モードにしておくことを要求します。

---

**注：** 次の手順は、DM643x DMP に搭載されている DSP を除くすべてのモジュールに直接適用できません。DSP モジュールのステートを遷移するためには、システム上の考慮事項をいくつか確認する必要があります。

---

モジュールのステート遷移を行う手順は、次のとおりです ( $n$  はモジュールに対応)。

1. PTSTAT の GOSTAT[0] ビットが 0 にクリアされるまで待ちます。新たな遷移を開始する前に、以前開始した遷移が終了するまで待つ必要があります。
2. MDCTL $n$  の NEXT ビットを SwRstDisable (0)、SyncReset (1)、ディスエーブル (2h)、イネーブル (3h) のいずれかにセットします。

---

**注：** このステップでは、MDCTL $n$  の複数の NEXT ビットに遷移機能をセットできます。後半のステップで、PTCMD の GO[0] ビットをセットするまで、実際には遷移は発生しません。

---

3. 遷移を開始するために、PTCMD の GO[0] ビットを 1 にセットします。
4. PTSTAT の GOSTAT[0] ビットが 0 にクリアされるまで待ちます。モジュールは PTSTAT の GOSTAT[0] ビットが 0 にクリアされた後でのみ、新たなステートで安全に動作します。

## 6.5 PSC での PSC エミュレーション・サポート

PSC は IcePick コマンドをサポートしていて、IcePick が認識したエミュレーション・ツールでパワー・ドメインおよびモジュールのステートを制御できます。DM643x DMP 上では、この IcePick のサポートは、C64x+ CPU ( AlwaysOn パワー・ドメイン 0 のモジュール番号 39 ) にのみ適用されます。

表 6-3 に、PSC によって認識される IcePick エミュレーション・コマンドを具体的に示します。これらは、DM643x DMP の C64x+ CPU に適用されるコマンドです。

表 6-3. IcePick エミュレーション・コマンド

パワー・オンおよびイネーブル機能	パワー・オンおよびイネーブルの説明	リセット機能	リセットの説明
スリープ抑制	ソフトウェアでモジュールをイネーブ ル・ステートから遷移させないようにエ ミュレーションできます。DM643x DMP に適用可能。	アサート・リセット	モジュールのローカル・リセット をアサートするようにエミュレー ションできます。DM643x DMP に 適用可能。
電源供給の強制	パワー・ドメインをオン状態へ強制的に 設定するようにエミュレーションできま す。AlwaysOn パワー・ドメインが常にオ ンのため、DM643x DMP には適用不可。	ウェイト・リセット	ソフトウェアでローカル・リセッ トのディアサートを開始した後、 長時間にわたりローカル・リセッ トをアサートしておくようにエ ミュレーションできます。 DM643x DMP に適用可能。
アクティブ状態の強制	モジュールをイネーブル・ステートへ強 制的に設定するようにエミュレーショ ンできます。DM643x DMP に適用可能。	ブロック・リセット	ソフトウェアで開始されたローカ ル・リセットおよびモジュール・ リセットをブロックするようにエ ミュレーションできます。 DM643x DMP に適用可能。

注： エミュレーション・ツールによって、上記のコマンドが取り除かれた場合、PSC はソフトウェアでセットされる PDCTL0 の NEXT ビット、MDCTL<sub>n</sub> の NEXT ビットに基づいてただちにステート遷移を実行します。

## 6.6 PSC 割り込み

PSC には、C64x+ 割り込みコントローラ ( INTC ) に接続される割り込みがあります。この割り込みは、割り込みマップでは PSCINT という名前が付いています。PSC 割り込みは、特定の IcePick エミュレーション・イベントが発生したときに生成されます。

### 6.6.1 割り込みイベント

PSC 割り込みは、次のイベントのいずれかが発生したときに生成されます。

- モジュールのステートをエミュレーションするイベント
- モジュールのローカル・リセットをエミュレーションするイベント

これらの割り込みイベントを表 6-4 に示し、詳細に説明します。

表 6-4. PSC 割り込みイベント

割り込みイネーブル・ビット		割り込み条件
コントロール・レジスタ	ステータス・ビット	
MDCTL <sub>n</sub>	EMUIHB	エミュレーションによってモジュール・ステートが変更されたときに、割り込みが発生します。
MDCTL <sub>n</sub>	EMURST	エミュレーションによってモジュールのローカル・リセットが変更されたときに、割り込みが発生します。



PSC 割り込みイベントが適用されるのは、モジュールのステートが IcePick エミュレーションによって MDCTL $n$  の NEXT ビットに示された、ユーザがプログラムした状態から変更されたときのみです。6.5 節で説明したように、DM643x DMP 上では、IcePick のサポートは C64x+ CPU (モジュール 39) にのみ適用されます。したがって、PSC 割り込み条件はモジュール 39 にのみ適用されます。

DM643x DMP は、シングル・プロセッサ・デバイスです。C64x+ CPU は、そのモジュールのステートをプログラムする必要はありません。C64x+ CPU モジュールのステートは、外部ホスト (PCI や HPI など) によってのみプログラムできます。その結果、表 6-4 に示す割り込みイベントは、外部ホストが C64x+ CPU モジュールのステートをプログラムする場合にのみ発生しますが、エミュレータはその指定されたステートを変更します。

### 6.6.1.1 モジュール・ステートをエミュレートするイベント

モジュール・ステートをエミュレートするイベントは、エミュレーションによってモジュールのステートが変更されるときに発生します。状態は、MDSTAT $n$  の EMUIHB ビットに示されます。モジュール・ステートをエミュレートするイベントが発生する条件は、具体的には次のとおりです。

- スリープ抑制がエミュレーションによってアサートされ、かつソフトウェアでイネーブル・ステートからモジュールを遷移する場合。
- アクティブ状態の強制がエミュレーションによってアサートされ、かつモジュールがまだイネーブル・ステートになっていない場合。

### 6.6.1.2 ローカル・リセットをエミュレートするイベント

エミュレーションによってモジュールのローカル・リセットが変更されると、ローカル・リセットをエミュレートするイベントが発生します。状態は、MDSTAT $n$  の EMURST ビットに示されます。モジュールのローカル・リセットをエミュレートするイベントが発生する条件は、具体的には次のとおりです。

- ソフトウェアでローカル・リセットはディアサートされるが、エミュレーションによってアサート・リセットがアサートされる場合。
- ウェイト・リセットがエミュレーションによってアサートされる場合。
- ブロック・リセットがエミュレーションによってアサートされ、かつソフトウェアでローカル・リセットのステートを変更する場合。

## 6.6.2 インタラプト・レジスタ

PSC 割り込みイネーブル・ビットは、MDCTL39 の EMUIHBIE ビットおよび MDCTL39 の EMURSTIE ビットです。

---

**注：** DSP への割り込みは、パワー・スリープ・コントローラ割り込み (PSCINT) も DSP 割り込みコントローラでイネーブルする必要があります。割り込みコントローラの詳細については、2.4.1 項を参照してください。

---

PSC 割り込みステータス・ビットは MERRPR1 の M[39] ビット、MDSTAT39 の EMUIHB ビット、および MDSTAT39 の EMURST ビットです。MERRPR1 のステータス・ビットは、エミュレーション割り込みを生成したモジュールを判別するためにソフトウェアでリードされます。その後、割り込みを発生したイベントを判別するために、ソフトウェアで MDSTAT39 の対応するステータス・ビットをリードすることができます。

PSC 割り込みクリア・ビットは、MERRCR1 の M[39] ビットです。

PSC 割り込み評価ビットは、INTEVAL の ALLEV ビットです。このビットがセットされると、PSC 割り込みロジックはイベント状態を強制的に再度評価します。いずれかのイベントがアクティブのままの場合 (いずれかのステータス・ビットがセットされる場合)、INTEVAL の ALLEV ビットが 1 にセットされると、PSCINT は DSP 割り込みコントローラへ再度アサートされます。PSC 割り込みを確実に取得するために、PSCINT 割り込みサービス・ルーチンを終了する前に、INTEVAL の ALLEV ビットをセットします。

すべての PSC レジスタをまとめた説明については、6.7 節を参照してください。

### 6.6.3 割り込み処理

次の手順で説明されているように、PSC 割り込みを処理します。まず、割り込みをイネーブルします。

1. 必要な割り込みイベントをイネーブルするように、MDCTL39 の EMUIHBIE ビットおよび EMURSTIE ビットをセットします。

**注：** イネーブルされたイベントが最低でも 1 つアクティブになると、PSC 割り込み PSCINT を DSP 割り込みコントローラへ送ります。

2. DSP 割り込みコントローラのパワー・スリープ・コントローラ割り込み (PSCINT) をイネーブルします。DSP への割り込みは、DSP 割り込みコントローラの PSCINT をイネーブルする必要があります。詳細については、2.4.1 項を参照してください。

DSP は割り込みを受け取ると、割り込みサービス・ルーチン (ISR) に入ります。

1. 割り込み要因を判別するために、MERRPR1 の  $M_n$  ビットをリードします。DM643x DMP では、M[39] のみが割り込み要因となることに注意してください。
2. 処理するアクティブなイベントごとに、次のことを行います。
  - 割り込みの要因となったイベントを判別するために、前のステップでリードされたステータス・ビットに応じて MDSTAT39 のイベント・ステータス・ビットをリードします。
  - アプリケーションで必要に応じた割り込みを処理します。
  - 対応するステータスをクリアするために、MERRCR1 の M[39] ビットをライトします。
  - INTEVAL の ALLEV ビットを 1 にセットします。アクティブな割り込みイベントが残っている場合、このビットをセットすると、PSCINT が DSP 割り込みコントローラへ再度アサートされます。

## 6.7 PSC レジスタ

表 6-5 に、PSC のメモリ・マップド・レジスタを示します。これらのレジスタのメモリ・アドレスについては、各デバイスのデータ・マニュアルを参照してください。

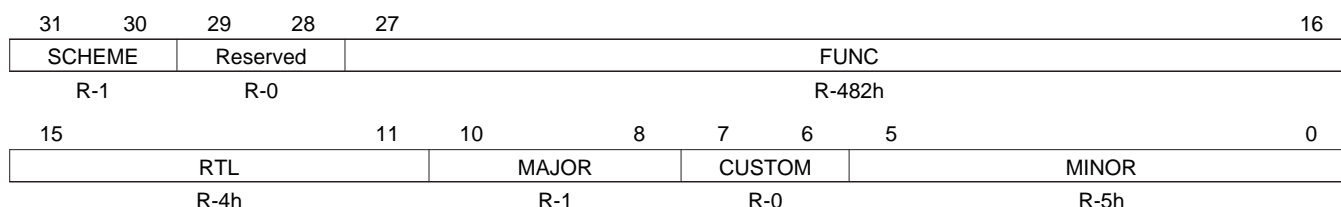
表 6-5. パワー・スリープ・コントローラ (PSC) のレジスタ

オフセット	レジスタ	説明	参照先
0h	PID	ペリフェラル・リビジョン・クラス・インフォメーション・レジスタ	6.7.1 項
18h	INTEVAL	インタラプト・エバリュエーション・レジスタ	6.7.2 項
44h	MERRPR1	モジュール・エラー・ペンディング・レジスタ 1	6.7.3 項
54h	MERRCR1	モジュール・エラー・クリア・レジスタ 1	6.7.4 項
120h	PTCMD	パワー・ドメイン・トランジション・コマンド・レジスタ	6.7.5 項
128h	PTSTAT	パワー・ドメイン・トランジション・ステータス・レジスタ	6.7.6 項
200h	PDSTAT[0]	パワー・ドメイン・ステータス 0 レジスタ	6.7.7 項
300h	PDCTL[0]	パワー・ドメイン・コントロール 0 レジスタ	6.7.8 項
800h-89Ch	MDSTAT0-39	モジュール・ステータス $n$ レジスタ	6.7.9 項
A00h-A9Ch	MDCTL0-39	モジュール・コントロール $n$ レジスタ	6.7.10 項

### 6.7.1 ペリフェラル・リビジョン・クラス・インフォメーション・レジスタ (PID)

ペリフェラル・リビジョン・クラス・インフォメーション・レジスタ (PID) を図 6-2 に示し、表 6-6 で説明します。

図 6-2. ペリフェラル・リビジョン・クラス・インフォメーション・レジスタ (PID)



凡例：R = リード専用。-n = リセット後の値。

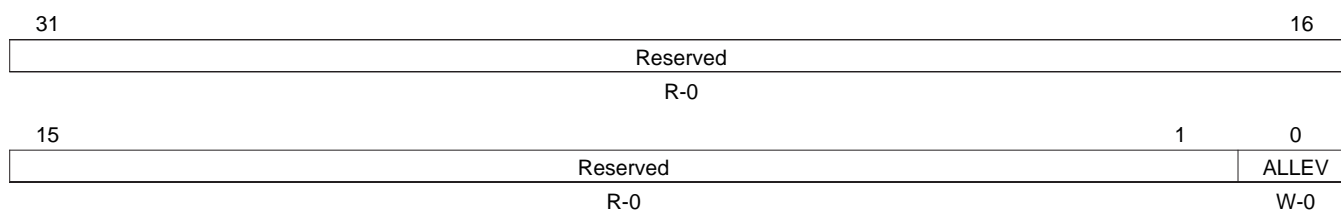
表 6-6. ペリフェラル・リビジョン・クラス・インフォメーション・レジスタ (PID) フィールドの説明

ビット	フィールド	値	説明
31-30	SCHEME	0 ~ 3h	新旧の構成を区別します。今後の構成をエンコードするための予約ビットです。
29-28	Reserved	0	予約。
27-16	FUNC	0 ~ FFFh	ソフトウェア互換モジュール・ファミリーを示します。
15-11	RTL	4h	RTL バージョン。 現在の RTL バージョン。
10-8	MAJOR	1h	メジャー・リビジョン。 現在のメジャー・バージョン。
7-6	CUSTOM	0 ~ 3h	特定のデバイス用の特別なバージョンを示します。
5-0	MINOR	5h	マイナー・リビジョン。 現在のマイナー・リビジョン。

### 6.7.2 インタラプト・エバリュエーション・レジスタ (INTEVAL)

インタラプト・エバリュエーション・レジスタ (INTEVAL) を図 6-3 に示し、表 6-7 で説明します。

図 6-3. インタラプト・エバリュエーション・レジスタ (INTEVAL)



凡例：R = リード専用。W = ライト専用。-n = リセット後の値。

表 6-7. インタラプト・エバリュエーション・レジスタ (INTEVAL) フィールドの説明

ビット	フィールド	値	説明
31-1	Reserved	0	予約。
0	ALLEV	0	PSC 割り込みを評価します。 このビットに 0 をライトしても影響はありません。
		1	このビットに 1 をライトすると、割り込み条件を再度評価します。

### 6.7.3 モジュール・エラー・ペンディング・レジスタ 1 (MERRPR1)

モジュール・エラー・ペンディング・レジスタ 1 (MERRPR1) を図 6-4 に示し、表 6-8 で説明します。C64x+ CPU (モジュール 39) だけが IcePick サポート機能を備えているため、エラー・コンディションを持つことができます。詳細については、6.5 節を参照してください。

図 6-4. モジュール・エラー・ペンディング・レジスタ 1 (MERRPR1)

31	Reserved			16
R-0				
15	8	7	6	0
Reserved		M[39]	Reserved	
R-0		R-0	R-0	

凡例：R = リード専用。-n = リセット後の値。

表 6-8. モジュール・エラー・ペンディング・レジスタ 1 (MERRPR1) フィールドの説明

ビット	フィールド	値	説明
31-8	Reserved	0	予約。
7	M[39]	0	モジュール 39 (C64x+ CPU) のモジュール割り込みステータス・ビット。 モジュール 39 には、エラー・コンディションがありません。
		1	モジュール 39 には、エラー・コンディションがあります。正確なエラー・コンディションについては、モジュール・ステータス 39 レジスタ (MDSTAT39) を参照してください。
6-0	Reserved	0	予約。

### 6.7.4 モジュール・エラー・クリア・レジスタ 1 (MERRCR1)

モジュール・エラー・クリア・レジスタ 1 (MERRCR1) を図 6-5 に示し、表 6-9 で説明します。C64x+ CPU (モジュール 39) だけが IcePick サポート機能を備えているため、エラー・コンディションを持つことができます。

図 6-5. モジュール・エラー・クリア・レジスタ 1 (MERRCR1)

31	Reserved			16
R-0				
15	8	7	6	0
Reserved		M[39]	Reserved	
R-0		W-0	R-0	

凡例：R = リード専用。W = ライト専用。-n = リセット後の値。

表 6-9. モジュール・エラー・クリア・レジスタ 1 (MERRCR1) フィールドの説明

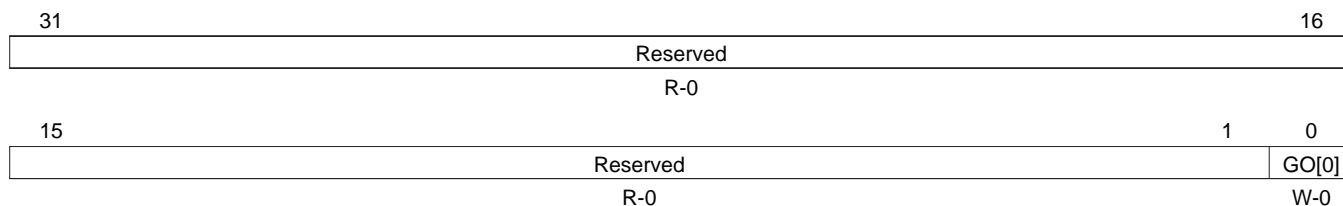
ビット	フィールド	値	説明
31-8	Reserved	0	予約。
7	M[39]	0	対応するモジュール・エラー・ペンディング・レジスタ 1 (MERRPR1) およびモジュール・ステータス 39 レジスタ (MDSTAT39) でセットされた割り込みステータス・ビットをクリアします。これは、モジュール 39 に関するものです。
		0	このビットに 0 をライトしても影響はありません。
		1	モジュール割り込みステータス・ビットをクリアします (MERRPR1 の M[39] ビット、MDSTAT39 の EMURST ビットおよび EMUIHB ビット)。

**表 6-9. モジュール・エラー・クリア・レジスタ 1 (MERRCR1) フィールドの説明**

ビット	フィールド	値	説明
6-0	Reserved	0	予約。

### 6.7.5 パワー・ドメイン・トランジション・コマンド・レジスタ (PTCMD)

パワー・ドメイン・トランジション・コマンド・レジスタ (PTCMD) を図 6-6 に示し、表 6-10 で説明します。

**図 6-6. パワー・ドメイン・トランジション・コマンド・レジスタ (PTCMD)**


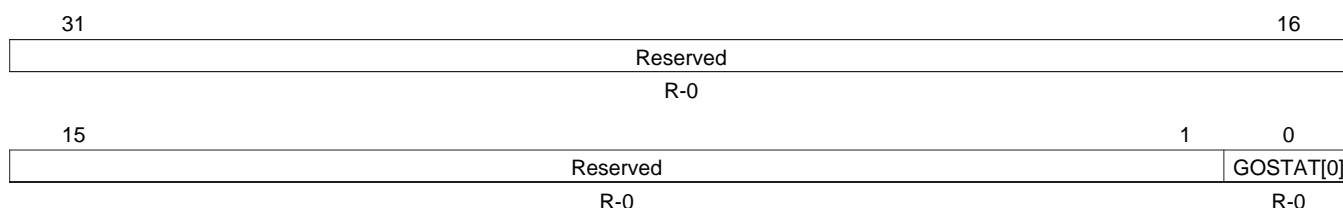
凡例：R = リード専用。W = ライト専用。-n = リセット後の値。

**表 6-10. パワー・ドメイン・トランジション・コマンド・レジスタ (PTCMD) フィールドの説明**

ビット	フィールド	値	説明
31-1	Reserved	0	予約。
0	GO[0]	0	AlwaysOn パワー・ドメインの GO 遷移コマンド。 このビットに 0 をライトしても影響はありません。
		1	このビットに 1 をライトすると、PSC はこのパワー・ドメインに関連するすべての NEXT フィールドを評価します (このドメインのすべてのモジュールに対する MDCTL <sub>n</sub> の NEXT ビットを含む)。NEXT フィールドのいずれも対応する現在のステート (MDSTAT <sub>n</sub> の STATE ビット) と一致しない場合、PSC はこれらのそれぞれのドメイン / モジュールを新しい NEXT ステートへ遷移します。

### 6.7.6 パワー・ドメイン・トランジション・ステータス・レジスタ (PTSTAT)

パワー・ドメイン・トランジション・ステータス・レジスタ (PTSTAT) を図 6-7 に示し、表 6-11 で説明します。

**図 6-7. パワー・ドメイン・トランジション・ステータス・レジスタ (PTSTAT)**


凡例：R = リード専用。-n = リセット後の値。

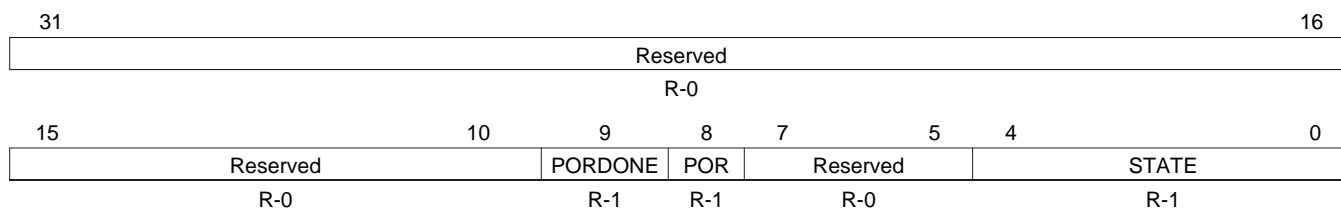
**表 6-11. パワー・ドメイン・トランジション・ステータス・レジスタ (PTSTAT) フィールドの説明**

ビット	フィールド	値	説明
31-1	Reserved	0	予約。
0	GOSTAT[0]	0	AlwaysOn パワー・ドメインの遷移状態。 遷移は進行していません。
		1	AlwaysOn パワー・ドメインのモジュールの遷移が進行中です。

### 6.7.7 パワー・ドメイン・ステータス 0 レジスタ (PDSTAT0)

パワー・ドメイン・ステータス  $n$  レジスタ (PDSTAT0) を図 6-8 に示し、表 6-12 で説明します。PDSTAT0 は AlwaysOn パワー・ドメインに適用されます。

図 6-8. パワー・ドメイン・ステータス 0 レジスタ (PDSTAT0)



凡例：R = リード専用。-n = リセット後の値。

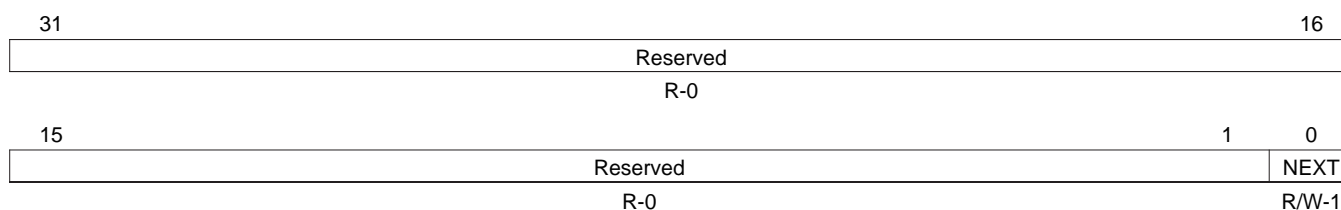
表 6-12. パワー・ドメイン・ステータス 0 レジスタ (PDSTAT0) フィールドの説明

ビット	フィールド	値	説明
31-10	Reserved	0	予約。
9	PORDONE	0	パワー・オン・リセット (POR) 完了状態。 パワー・ドメイン POR は完了していません。
		1	パワー・ドメイン POR は完了しています。
8	POR	0	パワー・ドメインのパワー・オン・リセット (POR) 状態。このビットは、ドメインのすべてのモジュールを含むこのパワー・ドメインの POR 状態を示します。 パワー・ドメイン POR はアサートされます。
		1	パワー・ドメイン POR はディアサートされます。
7-5	Reserved	0	予約。
4-0	STATE	0	パワー・ドメインの状態。 パワー・ドメインはオフ。
		1	パワー・ドメインはオン。

### 6.7.8 パワー・ドメイン・コントロール 0 レジスタ (PDCTL0)

パワー・ドメイン・コントロール  $n$  レジスタ (PDCTL0) を図 6-9 に示し、表 6-13 で説明します。PDCTL0 は AlwaysOn パワー・ドメインに適用されます。

図 6-9. パワー・ドメイン・コントロール 0 レジスタ (PDCTL0)



凡例：R/W = リード / ライト。R = リード専用。- $n$  = リセット後の値。

表 6-13. パワー・ドメイン・コントロール 0 レジスタ (PDCTL0) フィールドの説明

ビット	フィールド	値	説明
31-1	Reserved	0	予約。
0	NEXT	0	パワー・ドメインの次のステート。
		1	パワー・ドメインはオフ。 パワー・ドメインはオン。AlwaysOn ドメインは、常にこの値にプログラムする必要があります。

### 6.7.9 モジュール・ステータス $n$ レジスタ (MDSTAT $n$ )

モジュール・ステータス  $n$  レジスタ (MDSTAT0 ~ MDSTAT39) を図 6-10 に示し、表 6-14 で説明します。

図 6-10. モジュール・ステータス  $n$  レジスタ (MDSTAT $n$ )

31												18	17	16
Reserved													EMUIHB	EMURST
R-0													R-0	R-0
15	13	12	11	10	9	8	7	6	5			0		
Reserved		MCKOUT	Reserved	MRST	LRSTDONE	LRST	Reserved		STATE					
R-0		R-0	R-1	R-0	R-1	R-1	R-0		R-0					

凡例：R = リード専用。- $n$  = リセット後の値。

表 6-14. モジュール・ステータス  $n$  レジスタ (MDSTAT $n$ ) フィールドの説明

ビット	フィールド	値	説明
31-18	予約。	0	予約。
17	EMUIHB	0	エミュレーションで、モジュール・ステートを変更します。このビットは、DSP モジュール (モジュール 39) にのみ適用されます。このフィールドは、それ以外のすべてのモジュールで 0 となります。
		1	MDCTL39 の NEXT ビットでプログラムされたユーザ指定モジュール・ステートを変更するエミュレーションはありませぬ。 MDCTL39 の NEXT ビットでプログラムされたユーザ指定モジュール・ステートを変更するエミュレーション。 このイベントが発生するたびに、PSCINT を生成する場合、MDCTL39 の EMUIHBIE をセットする必要があります。
16	EMURST	0	エミュレーションで、モジュール・リセットを変更します。このビットは、DSP モジュール (モジュール 39) にのみ適用されます。このフィールドは、それ以外のすべてのモジュールで 0 となります。
		1	ユーザ指定モジュールのリセット・ステートを変更するエミュレーションはありませぬ。 ユーザ指定モジュールのリセット・ステートを変更するエミュレーション。 このイベントが発生するたびに、PSCINT を生成する場合、MDCTL39 の EMURSTIE をセットする必要があります。
15-13	Reserved	0	予約。
12	MCKOUT	1	モジュール・クロックの出力状態。モジュール・クロックの実際の状態を示します。
		0	モジュール・クロックはオフ。 モジュール・クロックはオン。
11	Reserved	1	予約。
10	MRST	0	モジュール・リセットの状態。モジュール・リセットの実際のステートを示します。
		1	モジュール・リセットはアサートされます。 モジュール・リセットはディアサートされます。
9	LRSTDONE	0	ローカル・リセットの完了。ソフトウェアで、このモジュールにアクセスする前にローカル・リセットが完了したかどうかをチェックします。このビットは、DSP モジュール (モジュール 39) にのみ適用されます。このフィールドは、それ以外のすべてのモジュールで 1 となります。
		1	ローカル・リセットは完了していません。 ローカル・リセットは完了。



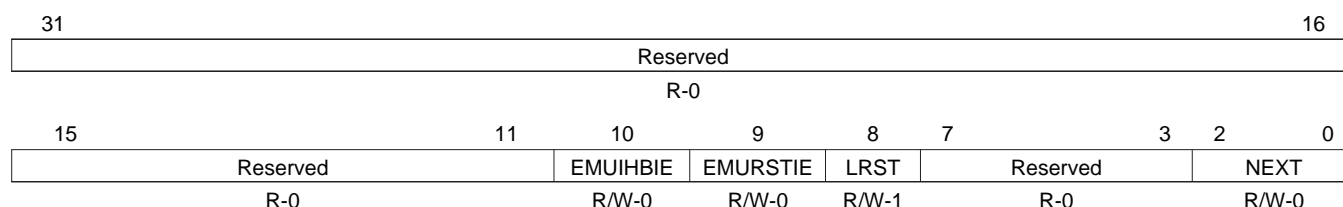
**表 6-14. モジュール・ステータス  $n$  レジスタ (MDSTAT $n$ ) フィールドの説明 (続き)**

ビット	フィールド	値	説明
8	LRST	0 1	モジュール・ローカル・リセットの状態。このビットは、DSP モジュール (モジュール 39) にのみ適用されます。 ローカル・リセットはアサートされます。 ローカル・リセットはディアサートされます。
7-6	Reserved	0	予約。
5-0	STATE	0 ~ 3Fh 0 1h 2h 3h 4h ~ 3Fh	モジュール・ステータスの状態。現在のモジュールの状態を示します。 SwRstDisable ステータス SyncReset ステータス ディスエーブル・ステータス イネーブル・ステータス 遷移を示します。

### 6.7.10 モジュール・コントロール $n$ レジスタ (MDCTL $n$ )

モジュール・コントロール  $n$  レジスタ (MDCTL0 ~ MDCTL39) を図 6-11 に示し、表 6-15 で説明します。

図 6-11. モジュール・コントロール  $n$  レジスタ (MDCTL $n$ )



凡例：R/W = リード / ライト。R = リード専用。- $n$  = リセット後の値。

表 6-15. モジュール・コントロール  $n$  レジスタ (MDCTL $n$ ) フィールドの説明

ビット	フィールド	値	説明
31-11	Reserved	0	予約。
10	EMUIHBIE	0	エミュレーションでモジュール・ステートを変更するための割り込みイネーブル。このビットは、DSP モジュール (モジュール 39) にのみ適用されます。このフィールドは、それ以外のすべてのモジュールに対して 0 にプログラムします。
		1	割り込みをディスエーブルします。
9	EMURSTIE	0	エミュレーションでモジュール・リセットを変更するための割り込みイネーブル。このビットは、DSP モジュール (モジュール 39) にのみ適用されます。このフィールドは、それ以外のすべてのモジュールに対して 0 にプログラムします。
		1	割り込みをイネーブルします。
8	LRST	0	モジュール・ローカル・リセットの制御。このビットは、DSP モジュール (モジュール 39) にのみ適用されます。このフィールドは、それ以外のすべてのモジュールに対して 1 にプログラムします。
		1	ローカル・リセットをアサートします。
7-3	Reserved	0	予約。
2-0	NEXT	0 ~ 7h	モジュールの次のステート。
		0	SwRstDisable ステート
		1h	SyncReset ステート
		2h	ディスエーブル・ステート
		3h	イネーブル・ステート
		4h ~ 7h	予約

## パワー・マネージメント

項目	ページ
7.1 概要.....	76
7.2 PSC および PLLC の概要 .....	76
7.3 クロック管理 .....	77
7.4 DSP によるスリープ・モード管理 .....	78
7.5 3.3 V 入出力 (I/O) パワー・ダウン .....	80
7.6 ビデオ DAC のパワー・ダウン .....	80

## 7.1 概要

多くのアプリケーションでは、電源（またはバッテリー）と熱考察の両方について、消費電力を最小限にする明確な要求事項がある場合があります。消費電力に対応するための2つのコンポーネントとして、アクティブ電力と漏れ電力があります。アクティブ電力は処理を実行するために消費される電力で、クロック周波数や実行される計算量におおよそ比例します。アクティブ電力を削減するには、必要な動作を限られた時間内で十分に完了できるクロック設定値で動作させるか、作業が完了するまで一定のクロック設定値で動作させてから、他の作業の実行が必要になるまで大幅にクロック・スピードを下げる（つまり、PLL バイパス・モードに設定する）などの方法でクロックを制御します。漏れ電力は静的な漏れ電流によるもので、クロック・レートに関係なく発生します。漏れ電力またはスタンバイ電力は、電源が供給され、動作ジャンクション温度におおよそ比例する限り、避けることができません。漏れ電力を回避するには、デバイスやサブシステムから完全に電源を取り除くことが唯一の保方法です。

TMS320DM643x DMP には、消費電力を管理するさまざまな手段が備えられています。これについては、これから説明します。非動作時にステートを基本的にフリーズすることにより、クロック・ツリー電源を削減するだけでなく、モジュール電源も削減する、ソフトウェア制御によるモジュール・クロックのゲート制御を行う方法と同様に、設計上クロックのゲート制御を自動化するさまざまな方法を使用できます。スイッチング電源を削減するために、クロック管理を行うことにより、チップ上のクロック・スピードを低速にすることができます。具体的にこの管理を行うために、DM643x DMP にはすべてのパワー・マネージメント機能が組み込まれています（表 7-1 を参照）。

**表 7-1. パワー・マネージメント機能**

パワー・マネージメント機能	説明
<b>クロック管理</b>	
PLL のパワー・ダウン	スイッチング電源を削減するために、PLL 非使用時にパワー・ダウンすることができます。
モジュール・クロックのオン / オフ	スイッチング電源を削減するために、モジュール・クロックのオン / オフを切り替えることができます。
モジュール・クロックの周波数スケーリング	スイッチング電源を削減するために、モジュール・クロックの周波数をスケーリングすることができます。
<b>DSP スリープ管理</b>	
DSP スリープ・モード	スイッチング電源を削減するために、DSP をスリープ・モードにすることができます。
<b>入出力 (I/O) 管理</b>	
3.3 ボルト入出力 (I/O) パワー・ダウン	入出力 (I/O) セル電源を削減するために、3.3 V 入出力 (I/O) をパワー・ダウンすることができます。
DAC のパワー・ダウン	DAC 電源を削減するために、DAC をパワー・ダウンすることができます。

## 7.2 PSC および PLLC の概要

パワー・スリープ・コントローラ (PSC) は、システム・パワーのオン / オフ、クロックのオン / オフ、およびリセットを管理する重要な機能を実行しています。同様に、PLL コントローラ (PLLC) はデバイス・クロックを生成する上で重要な機能を実行しています。PSC および PLLC については、本章全体で説明します。PSC の詳細については第 6 章を、PLLC の詳細については第 4 章および第 5 章をそれぞれ参照してください。

## 7.3 クロック管理

### 7.3.1 モジュール・クロックのオン/オフ

モジュールのアクティブな消費電力を 0 に削減するために、モジュール・クロックをオン/オフする機能が組み込まれているので、ソフトウェアでモジュールのクロックを個別にディスエーブルすることができます。クロックが再起動された場合、モジュールは停止した時点から動作を再開します。

---

**注：** モジュールに対するクロックを停止すると、アクティブな消費電力にのみ影響を与え、リーク電流の消費電力には影響を与えません。

---

アクセスが行われているときにモジュールのクロックを停止した場合、そのアクセスは発生せず、デバイスをロックしてしまう可能性があります。ユーザは、クロックを停止する前にモジュールに対するすべてのトランザクションを確実に終了させる必要があります。パワー・スリープ・コントローラ (PSC) は、モジュール・クロックのゲート制御を管理します。PSC には、内部バスのアクティビティを監視することによりシステムのハングを防止する保護機能があります。つまり、内部バスからモジュールに対するアクセスがないことをチェックした後にモジュールに対する内部クロックのゲート制御のみを行います。

PSC を使用してモジュール・クロックのオン/オフを切り替える手順については、第 6 章を参照してください。また、DSP クロックのオン/オフを切り替えるには特別な考慮事項が必要です。DSP クロックのオン/オフを切り替える手順の詳細については、7.4.2 項を参照してください。

一部のペリフェラルには、そのモジュール境界内にクロックのゲート制御を行うコンポーネントを搭載することにより、節電機能も提供しています。これらの節電機能の詳細については、各ペリフェラルのユーザズ・ガイドを参照してください。

### 7.3.2 モジュール・クロックの周波数スケーリング

モジュール・クロックの周波数をスケーリングするには、PLL の通倍および分周パラメータをプログラムします。クロック周波数を下げると、アクティブなスイッチング電源の消費電力を周波数に合わせてリニアに削減できます。漏れ消費電力には影響しません。

PLL 周波数および周波数の制約をプログラムする方法については、第 4 章および第 5 章を参照してください。

### 7.3.3 PLL のバイパスおよびパワー・ダウン

DM643x DMP に搭載されている PLL をバイパスすることができます。PLL をバイパスすると、PLL 出力 (PLLOUT) ではなく PLL リファレンス・クロック (MXI/CLKIN) が PLLC の SYSCLK 分周器 (PLLDIV $n$ ) へ送られます。PLL リファレンス・クロックは通常 27 MHz で動作します。したがって、このモードを使用すれば PLL を使わずに、システムがきわめて低速で動作している期間、コア周波数およびモジュール・クロックの周波数をほとんど保守する必要のないレベルへ引き下げることができます。また、他のアクティブ電力を節電するために PLL バイパスする際、PLL をパワー・ダウンすることができます。

PLL のバイパス設定および PLL のパワー・ダウンについては、第 4 章および第 5 章を参照してください。

## 7.4 DSP によるスリープ・モード管理

C64x+ DSP は、消費電力を削減するためにスリープ・モード管理をサポートしています。

- DSP を完全にシャットダウン可能
- C64x+ メガモジュールをスリープ・モードにすることが可能
  - C64x+ CPU をスリープ・モードにすることが可能

DM643x DMP では、DSP 内部メモリ (L1P、L1D、L2) に対するスリープ・モードはサポートされていません。

### 7.4.1 DSP スリープ・モード

DSP サブシステムの C64x+ メガモジュールには、C64x+ メガモジュールのコンポーネントをパワー・ダウンする機能を制御するパワー・ダウン・コントローラ (PDC) が組み込まれています。PDC の詳細については、『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) を参照してください。

### 7.4.2 DSP モジュール・クロックのオン/オフ

7.4.1 項で説明したように、C64x+ メガモジュールは節電するために搭載されているコンポーネントのクロックをゲート制御できます。節電を行う方法として他にも、パワー・スリープ・コントローラ (PSC) をプログラムして、C64x+ メガモジュールをディスエーブル・ステートに遷移させることにより、C64x+ メガモジュールへ供給するクロック・ソースを停止する方法もあります。C64x+ DSP では、自身に対してこのプログラミング方法を実行することができません。これは、C64x+ DSP のクロック・ソースが処理の途中でゲート制御されると、C64x+ DSP が PSC プログラミング・シーケンスを完了できないためです。さらなる節電のために、C64x+ DSP へのクロック・ソース供給停止が要求される場合には、外部ホストから C64x+ メガモジュールをディスエーブルするために PSC を (HPI や PCI インターフェイスを使用して) プログラムします。同様に、このようなケースでは、外部ホストで C64x+ メガモジュールをイネーブルするために PSC をプログラムします。

#### 7.4.2.1 DSP モジュール・クロックのオン

クロックがイネーブル・ステートの場合、DSP モジュール・クロックは DSP モジュール・リセットがディアサートされている間イネーブルされます。これは、通常の DSP 実行時のステートです。DSP のデフォルトはイネーブル・ステートのため、この DSP モジュール・クロックをオンにするプロセスは通常必要ではありません。このプロセスは、外部ホストによって DSP がディスエーブル・ステートにされた後、DSP をウェイク・アップするためにのみ必要です (7.4.2.2 項)。

- ホスト : DSP へのクロック供給をイネーブルします。
  - PTSTAT の GOSTAT[0] ビットが 0 にクリアされるまで待ちます。パワー・ドメインが新たな遷移を開始する前に、以前開始された遷移がすべて終了するまで待つ必要があります。
  - DSP モジュールがイネーブルに遷移できるように、MDCTL39 の NEXT ビットを 3h にセットします。
  - ステートの遷移を開始するために、PTCMD の GO[0] ビットを 1 にセットします。
  - PTSTAT の GOSTAT[0] ビットが 0 にクリアされるまで待ちます。ドメインは GOSTAT[0] ビットが 0 にクリアされた後、新たなステートでのみ安全に動作します。
  - MDSTAT39 の STATE ビットが 3h に変更されるまで待ちます。モジュールは MDSTAT39 の STATE ビットに新しいステートが反映された後、新たなステートでのみ安全に動作します。
- ホスト : DSP をウェイクします。
  - ディスエーブルから遷移が行われると、ウェイク・アップ割り込みとしてあらかじめ設定されていた DSP 割り込みがトリガされます。

---

**注：** このステップは、ディスエーブル・ステートから遷移する場合にのみ適用されます。前の状態がディスエーブル・ステートだった場合、DSP をウェイクするためにウェイク・アップ割り込みをトリガする必要があります。この例では、アイドル・ステートに入る前に DSP によってこの割り込みがイネーブルされたということを前提にしています。前の状態がソフトウェア・リセット・ディスエーブル・ステートまたは同期リセット・ステートだった場合、これらのステートが DSP モジュール・リセットをアサートするので、DSP をウェイクする必要はありません。ソフトウェア・リセット・ディスエーブル・ステートおよび同期リセット・ステートについては、第 10 章を参照してください。DSP 割り込みの詳細については、『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) を参照してください。

---

### 7.4.2.2 DSP モジュール・クロックのオフ

クロックがディスエーブル・ステートの場合、DSP のモジュール・クロックは DSP リセットがディアサートされている間ディスエーブルです。このステートは通常、節電を実現する DSP クロックをディスエーブルするために使われます。7.4.2 項で説明したように、DSP では自身をディスエーブル・ステートにすることはできません。外部ホストにてこの処理を行います。たとえば、HPI または PCI ペリフェラルを介して外部ホストとの接続を行います。

- ホスト：パワー・ダウンの準備をするように DSP に通知します。
- DSP：DSP がパワー・ダウンさせるときは、事前に現在処理中の動作をすべて終了して、C64x+ メガモジュールへのアクセスがないことを確認します。
  - C64x+ メガモジュールへの転送機能を有するすべてのマスタ・ペリフェラル（ホストを除く）をディスエーブルするように PSC をプログラムします。
  - C64x+ メガモジュールにアクセスできる未処理の EDMA 転送がないことを確認するために、EDMA の転送状態をチェックします。
- DSP：パワー・ダウンを準備します。
  - PDCCMD を 0001 5555h にセットします。PDCCMD は、DSP パワー・ダウン・コントローラ・モジュールのコントロール・レジスタです。

---

**注：** DSP がスーパーバイザ・モードで動作している場合、このレジスタヘライトのみできます。

---

- ホストが DSP クロック・オン・シーケンスで DSP をウェイクするために使用する割り込みの 1 つをイネーブルします。
- IDLE 命令を実行します。IDLE 命令は、C64x+ CPU 命令セットのプログラム命令です。CPU が IDLE 命令を実行すると、CPU から PDC に対して、PDCCMD (0181 0000h) レジスタでセットされたビットに従って、DSP のパワー・ダウンを通知し、その実行を開始します。PDC および IDLE 命令の詳細については、『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) を参照してください。
- ホスト：DSP クロックをディスエーブルします。
  - PTSTAT の GOSTAT[0] ビットが 0 にクリアされるまで待ちます。パワー・ドメインが新たな遷移を開始する前に、以前開始された遷移がすべて終了するまで待つ必要があります。
  - DSP モジュールがディスエーブルに遷移できるように、MDCTL39 の NEXT ビットを 2h にセットします。
  - ステートの遷移を開始するために、PTCMD の GO[0] ビットを 1 にセットします。
  - PTSTAT の GOSTAT[0] ビットが 0 にクリアされるまで待ちます。ドメインは GOSTAT[0] ビットが 0 にクリアされた後、新たなステートでのみ安全に動作します。
  - MDSTAT39 の STATE ビットが 2h に変更されるまで待ちます。モジュールは MDSTAT39 の STATE ビットに新しいステートが反映された後、新たなステートでのみ安全に動作します。

### 3.3 V 入出力 (I/O) パワー・ダウン

---

#### 7.5 3.3 V 入出力 (I/O) パワー・ダウン

3.3 V 入出力 (I/O) ドライバは、DC バイアス電流を要求する設計手法で 1.8 V のトランジスタから加工されます。これらの入出力 (I/O) セルは、DC 電流をオフにするパワー・ダウン・モードを備えています。システム・モジュールの VDD3P3V\_PWDN レジスタで、このスタンバイ・モードを制御します。VDD3P3V\_PWDN レジスタの詳細については、各デバイスのデータ・マニュアルを参照してください。

#### 7.6 ビデオ DAC のパワー・ダウン

DM643x DMP ビデオ・プロセッシング・バック・エンド (VPBE) には、アナログ・テレビ・ディスプレイをドライブするために 4 つのビデオ DA コンバータ (DAC) が組み込まれています。VPBE のビデオ・エンコーダ (VENC) モジュールには、DAC をイネーブル/ディスエーブルするレジスタが組み込まれています。VMOD の VIE ビットを使用すると、4 つの DAC の出力はビデオ信号に関係なく強制的に低電圧レベルに変更することができます。また、DACTST の DAPD[3:0] ビットを使用すると、それぞれの DAC を個別にディスエーブルすることもできます。レジスタの説明および DAC のパワー・ダウンの詳細については、『TMS320DM643x DMP Video Processing Back End (VPBE) User's Guide』(資料番号 [SPRU952](#)) を参照してください。



## 割り込みコントローラ

---

---

---

C64x+ メガモジュールには、CPU 割り込みを管理するために割り込みコントローラ (INTC) が組み込まれています。割り込みコントローラは、システム・イベントを CPU の割り込みおよび例外入力に対して関連付けを行います。割り込みコントローラは、最大で 128 個のシステム・イベントをサポートし、これらのシステム・イベントを 12 個の CPU 割り込みにマップします。システム・イベントのリストについては、各デバイスのデータ・マニュアルを参照してください。『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) の「割り込みコントローラ」では、INTC および DSP デバイス・イベントを 12 個の CPU 割り込みへマップする方法について完全に説明しています。



## システム・モジュール

項目	ページ
9.1 概要.....	84
9.2 デバイス識別.....	84
9.3 デバイス設定.....	84
9.4 3.3 V 入出力 (I/O) パワー・ダウン.....	85
9.5 ペリフェラルの状態および制御.....	85
9.6 帯域管理.....	86
9.7 ブート制御.....	87

## 9.1 概要

TMS320DM643x DMP システム・モジュールは、デバイスに要求される状態とトップ・レベルの制御ロジックを含むシステム・レベル・モジュールです。システム・モジュールは、DSP からアクセスできる一連のステータス・レジスタおよびコントロール・レジスタで構成され、すべてのシステム機能および動作をサポートします。

- デバイス識別
- デバイス設定
  - ピン・マルチプレクス制御
  - デバイス・ブート時の設定状態
- パワー・マネージメント
  - $V_{DD}$  3.3 V 入出力 (I/O) パワー・ダウン制御
- 特別なペリフェラルの状態および制御
  - タイマ制御
  - VPSS クロックおよび DAC の制御
  - DDR2 VTP 制御
  - HPI 制御
- 帯域管理
  - バス・マスタの DMA プライオリティ制御
  - EDMA 転送チャネルの設定
- ブート制御

本章では、システム・モジュールについて説明します。

## 9.2 デバイス識別

システム・モジュールの DEVICE\_ID レジスタには、ソフトウェアでリード可能な JTAG ID デバイスのバージョンが含まれています。ソフトウェアでは、このレジスタを使用して実行中のデバイスのバージョンを判別することができます。レジスタのフォーマットおよび説明については、各デバイスのデータ・マニュアルに記述されています。

## 9.3 デバイス設定

システム・モジュールには、ピン・マルチプレクスを制御するレジスタおよびブート時の設定状態を示すレジスタが含まれています。

### 9.3.1 ピン・マルチプレクス制御

DM643x DMP では、最小のパッケージで大量のペリフェラル機能に対応するためにピン・マルチプレクス機能が多く使用されています。この作業を実行するために、ハードウェア設定 (デバイス・リセット時にラッチされるコンフィギュレーション・ピン) とレジスタのプログラム制御 (PINMUX0 および PINMUX1 レジスタ) を組み合わせてピン・マルチプレクスを制御します。ハードウェアでは、ペリフェラルに対して適切なピン・マルチプレクスが選択されていることもインターフェイス・モードが使用されていることも確認されません。

ピン・マルチプレクスおよび制御の詳細については、各デバイスのデータ・マニュアルに記述されています。

### 9.3.2 デバイス・ブート時の設定状態

デバイス・ブートおよびコンフィギュレーション設定値は、デバイス・リセット (POR または RESET) 時にラッチされ、BOOTCFG レジスタにキャプチャされます。このレジスタおよびブート時のコンフィギュレーション設定値については、各デバイスのデータ・マニュアルを参照してください。

## 9.4 3.3 V 入出力 (I/O) パワー・ダウン

VDD3P3V\_PWDN レジスタは、3.3 V 入出力 (I/O) セルに対する電源を制御します。一部の 3.3 V 入出力 (I/O) のデフォルトは、節電のためパワー・ダウンに設定されています。VDD3P3V\_PWDN レジスタについては、各デバイスのデータ・マニュアルを参照してください。

## 9.5 ペリフェラルの状態および制御

いくつかの DM643x DMP ペリフェラル・モジュールは、システム・レベルの制御ロジックをさらに必要とします。それらのレジスタについては、ここで説明します。

### 9.5.1 タイマ制御

タイマ・コントロール・レジスタ (TIMERCTL) は、タイマ 0 およびタイマ 2 (ウォッチドッグ・タイマ) に対してさらに制御を行います。このレジスタの詳細については、各デバイスのデータ・マニュアルを参照してください。

### 9.5.2 VPSS クロックおよび DAC の制御

ビデオ・プロセッシング・サブシステム (VPSS) のクロックは、VPSS クロック・コントロール・レジスタ (VPSS\_CLKCTL) を介して制御されます。このレジスタの詳細については、各デバイスのデータ・マニュアルを参照してください。

### 9.5.3 DDR2 VTP 制御

DDR2 VTP イネーブル・レジスタ (DDRVTPER) は、DDR2 メモリ・コントローラの VTP IO バッファのキャリブレーション・プロセスで他のレジスタとともに使われます。このレジスタのロケーションについては、各デバイスのデータ・マニュアルを参照してください。VTP IO バッファのキャリブレーション・プロセスの詳細については、『TMS320DM643x DMP DDR2 Memory Controller User's Guide』(資料番号 [SPRU986](#)) を参照してください。

### 9.5.4 HPI 制御

HPI コントロール・レジスタ (HPICTL) は、HPI 動作のホストでのバースト・ライト時のタイム・アウト値を制御します。このレジスタの詳細については、各デバイスのデータ・マニュアルを参照してください。

## 9.6 帯域管理

### 9.6.1 バス・マスタ DMA のプライオリティ制御

マスタとスレーブ間で許可されている接続を判別するために、それぞれのマスタは関連付けられた固有のマスタ ID (mstid) を備えている必要があります。DM643x DMP マスタごとのマスタ ID を表 9-1 に示します。

**表 9-1. TMS320DM643x DMP のマスタ ID**

MSTID	マスタ
0-1	予約
2	DSP プログラム / データ
3	DSP CFG
4-7	予約
8	VPSS
9	予約
10	EDMA チャンネル・コントローラ
11-15	予約
16	EDMA チャンネル 0 リード
17	EDMA チャンネル 0 ライト
18	EDMA チャンネル 1 リード
19	EDMA チャンネル 1 ライト
20	EDMA チャンネル 2 リード
21	EDMA チャンネル 2 ライト
22-31	予約
32	EMAC
33-35	予約
36	VLYNQ
37	HPI
38	PCI
39-63	予約

それぞれのスイッチ・セントラル・リソース (SCR) は、コマンドを送るマスタの優先順位に基づいた優先順位付けを行います。各バス・マスタのプライオリティは、チップ・レベルのバス・マスタ・プライオリティ・コントロール・レジスタ (MSTPRI0 または MSTPRI1) でプログラムされます。バス・マスタごとのデフォルトのプライオリティ・レベルを表 9-2 に示します。アプリケーション・ソフトウェアで、これらの値を変更すれば目的のシステム性能を確保できるはず

**表 9-2. TMS320DM643x DMP のマスタごとのデフォルトのプライオリティ**

マスタ	デフォルトのプライオリティ
VPSS	0 <sup>(1)</sup>
EDMA チャンネル 0	0 <sup>(2)</sup>
EDMA チャンネル 1	0 <sup>(2)</sup>
EDMA チャンネル 2	0 <sup>(2)</sup>
DSP (DMA)	7 <sup>(3)</sup>
DSP (CFG)	1
EMAC	4
VLYNQ	4
PCI	4

(1) VPSS PCR レジスタのデフォルト値

(2) EDMA QUEPRI レジスタのデフォルト値

(3) DSP MDMAARBE.PRI フィールドのデフォルト値

## 9.6.2 EDMA 転送コントローラの設定

システム・モジュールの EDMA トランスファ・コントローラ・デフォルト・バースト・サイズ・コンフィギュレーション・レジスタ (EDMATCCFG) で、EDMA 転送コントローラ (EDMATC0、EDMATC1、EDMATC2) に対するデフォルトのバースト・サイズを設定します。このレジスタの詳細については、各デバイスのデータ・マニュアルを参照してください。

## 9.7 ブート制御

システム・モジュールには、次のブート・コントロール・レジスタが含まれています。

- デバイス・ブート・コンフィギュレーション・レジスタ (BOOTCFG)
- ブート・コンプリート・レジスタ (BOOTCMPLT)
- DSP ブート・アドレス・レジスタ (DSPBOOTADDR)

これらのレジスタについては、第 11 章および各デバイスのデータ・マニュアルを参照してください。





## リセット

項目	ページ
10.1 概要.....	90
10.2 リセット・ピン.....	90
10.3 リセット時に行われるデバイス設定.....	90
10.4 DSP リセット.....	91

## 10.1 概要

TMS320DM643x DMP には、さまざまなタイプのリセットがあります。リセットの種類は、開始方法と / もしくは、チップに対する影響で異なります。それぞれのタイプを表 10-1 で簡単に説明します。リセット・タイプごとの詳細については、各デバイスのデータ・マニュアルを参照してください。

表 10-1. リセット・タイプ

タイプ	開始条件	影響
POR (パワー・オン・リセット)	POR ピンが Low	チップの完全なリセット (コールド・リセット)、メモリ、エミュレーション・ロジックをはじめとするすべてのモジュールをリセットします。デバイスの電源投入時、パワー・オン・リセット (POR) ピンを Low にドライブする必要があります。デバイス・ブートおよびコンフィギュレーション・ピンが、ラッチされます。
ウォーム・リセット	RESET ピンが Low	エミュレーション・ロジックを除く、メモリをはじめとするすべてのモジュールをリセットします。エミュレータはウォーム・リセット中、動作したままです。デバイス・ブートおよびコンフィギュレーション・ピンが、ラッチされます。
マックス・リセット	DSP エミュレータまたはウォッチドッグ・タイマ (タイマ 2)	ウォーム・リセットと同様の影響。デバイス・ブートおよびコンフィギュレーション・ピンは再度ラッチされません。
モジュール / ペリフェラルのローカル・リセット	DSP または外部ホスト・ソフトウェア	固有のモジュールを個別にリセットします。モジュール・リセットは、デバッグ・ツールとして想定されています。量産時に使用するツールとして必ずしも想定されているわけではありません。
DSP ローカル・リセット	外部ホスト・ソフトウェア	DSP CPU をリセットします。DSP の内部メモリ (L1P、L1D、L2) はリセットされません。

## 10.2 リセット・ピン

DM6437 DMP には、2 つのデバイス・レベルでグローバルにリセットを行うピンとしてパワー・オン・リセット (POR) およびウォーム・リセット (RESET) があります。これらのリセット・ピンの詳細については、各デバイスのデータ・マニュアルを参照してください。

## 10.3 リセット時に行われるデバイス設定

POR およびウォーム・リセットが発生すると、DM6437 DMP はブートおよびコンフィギュレーション・ピンから値をラッチします。ブートとコンフィギュレーション・ピン、およびデバイスのデフォルト・ステートについては、各デバイスのデータ・マニュアルを参照してください。

## 10.4 DSP リセット

**注：** DSP ローカル・リセットおよび DSP モジュール・リセットの影響は、十分な検証を行っていません。したがって、これらのリセットはサポートされていないので使用してはいけません。その代わりに、POR または RESET ピンを使用して DSP 全体をリセットしてください。

パワー・スリープ・コントローラ (PSC) のレジスタへアクセスすることにより、外部ホスト (PCI または HPI) から DSP ローカル・リセットおよび DSP モジュール・リセットをアサートすることもディアサートすることもできます。DSP ローカル・リセットがアサートされても、DSP の内部メモリ (L1P、L1D、L2) にはアクセスできます。ローカル・リセットは DSP CPU のみをリセットします。DSP モジュールがイネーブルまたはディスエーブル・状態で動作している場合、ローカル・リセットは有効です。これはモジュール・リセットが SyncReset および SwRstDisable ステートでアサートされ、かつモジュール・リセットがローカル・リセットを優先するためです。DSP モジュール・リセットの目的は、外部ホストが DSP を完全にリセットすることです。DSP ローカル・リセットの目的は、ホストがコードを DSP の内部メモリへロードしている間、外部ホストが CPU をリセット状態に保持することです。このステップは、ホストが DSP をモジュール・リセット状態にして、それ以降で DSP をイネーブルした後に有効となります。PSC の詳細については、第 6 章を参照してください。ここでは、DSP ローカル・リセットおよびモジュール・リセットを開始する方法について説明します。

### 10.4.1 DSP ローカル・リセット

次のステップでは、外部ホストで DSP へローカル・リセットをアサート / ディアサートする方法について説明します。

1. DSP リセットをアサートするために、MDCTL39 の LRST ビットを 0 にクリアします。
2. DSP リセットをディアサートするために、MDCTL39 の LRST ビットを 1 にセットします。

### 10.4.2 DSP モジュール・リセット

外部ホストで DSP モジュール・リセットをアサートするために PSC をプログラムするには、DSP をソフトウェア・リセット・ディスエーブル (SwRstDisable) ステートまたは同期リセット (SyncReset) ステートのいずれかに置きます。これらの PSC ステートの説明については、第 6 章を参照してください。

#### 10.4.2.1 ソフトウェア・リセット・ディスエーブル (SwRstDisable)

ソフトウェア・リセット・ディスエーブル (SwRstDisable) ステートでは、DSP のモジュール・リセットがアサートされ、かつそのモジュール・クロックはオフになります。このステートを使用して DSP をリセットできます。次のステップでは、DSP をソフトウェア・リセット・ディスエーブル・ステートにする方法について説明します。

- ホスト：パワー・ダウンの準備をするように DSP に通知します。
- DSP：DSP をアイドル・ステートにします。
  - PDCCMD を 0001 5555h へセットします。PDCCMD は、DSP パワー・ダウン・コントローラ・モジュールのコントロール・レジスタです。

**注：** DSP がスーパーバイザ・モードで動作している場合、このレジスタへはライトのみできます。

- DSP がイネーブル・状態で動作している場合、IDLE 命令を実行します。IDLE 命令は、C64x+ CPU 命令セットのプログラム命令です。CPU が IDLE 命令を実行すると、CPU から PDC に対して、PDCCMD (0181 0000h) レジスタでセットされたビットに従って、DSP のパワー・ダウンを通知し、実行を開始します。PDC および IDLE 命令の詳細については、『TMS320C64x+ DSP Megamodule Reference Guide』(資料番号 [SPRU871](#)) を参照してください。
- ホスト：ソフトウェア・リセットで DSP をディスエーブルします。
  - PTSTAT の GOSTAT[0] ビットが 0 にクリアされるまで待ちます。パワー・ドメインが新たな遷移を開始する前に、以前開始された遷移がすべて終了するまで待つ必要があります。
  - DSP モジュールが SwRstDisable に遷移できるように、MDCTL39 の NEXT ビットを 0 にクリアします。
  - ステートの遷移を開始するために、PTCMD の GO[0] ビットを 1 にセットします。

- PTSTAT の GOSTAT[0] ビットが 0 にクリアされるまで待ちます。モジュールは GOSTAT[0] が 0 にクリアされた後でのみ、新たな状態で安全に動作します。
- ホスト：DSP ローカル・リセットをアサートします（オプション）。
  - MDCTL39 の LRST ビットを 0 にクリアします。このステップはオプションです。このステップは、DSP ローカル・リセットをアサートします。また、このステップがここに組み込まれているため、DSP はこれ以降ホストによりイネーブルされてもただちに動作を開始しません。通常、ホストはコードが適切にロードされていることを確認後、ローカル・リセットのみをディアサートします。

#### 10.4.2.2 同期リセット (SyncReset)

同期リセット (SyncReset) ステートでは、DSP のモジュール・リセットがアサートされ、かつそのモジュール・クロックはイネーブルされます。このステートを使用して DSP をリセットできます。次のステップでは、DSP を同期リセット・ステートにする方法について説明します。

- ホスト：パワー・ダウンの準備をするように DSP に通知します。
- DSP：DSP をアイドル・ステートに置きます。
  - PDCCMD を 0001 5555h へセットします。PDCCMD は、DSP パワー・ダウン・コントローラ・モジュールのコントロール・レジスタです。

---

**注：** DSP がスーパーバイザ・モードで動作している場合、このレジスタへはライトのみできます。

---

- IDLE 命令を実行します。
- ホスト：DSP を同期リセットします。
  - PTSTAT の GOSTAT[0] ビットが 0 にクリアされるまで待ちます。パワー・ドメインが新たな遷移を開始する前に、以前開始された遷移がすべて終了するまで待つ必要があります。
  - DSP モジュールが SyncReset に遷移できるように、MDCTL39 の NEXT ビットを 1 にセットします。
  - ステートの遷移を開始するために、PTCMD の GO[0] ビットを 1 にセットします。
  - PTSTAT の GOSTAT[0] ビットが 0 にクリアされるまで待ちます。モジュールは GOSTAT[0] ビットが 0 にクリアされた後でのみ、新たな状態で安全に動作します。
- ホスト：DSP ローカル・リセットをアサートします（オプション）。
  - MDCTL39 の LRST ビットを 0 にクリアします。このステップはオプションです。このステップは、DSP ローカル・リセットをアサートします。また、このステップがここに組み込まれているため、DSP はこれ以降ホストによりイネーブルされてもただちに動作を開始しません。通常、ソフトウェアでコードが適切にロードされていることを確認後、ローカル・リセットのみをディアサートします。

## ブート・モード

---

---

---

TMS320DM643x DMP は、直接非同期 EMIF/NOR から、または内部のブート ROM からブートできます。これは、デバイス・ブートおよびコンフィギュレーション・ピンの設定値によって決まります。ブートおよびコンフィギュレーション・ピンの入力ステータスはデバイス・リセットがディアサートされると、サンプリングされ、BOOTCFG レジスタへラッチされます。ブートおよびコンフィギュレーション・ピン、および DM643x DMP でサポートされているブート・モードのリストについては、各デバイスのデータ・マニュアルを参照してください。

すべてのブート・モードでは、C64x+ CPU はリセット状態からただちに解放され、次の 2 つのアドレスのいずれかから実行を開始します。

- EMIFA チップ・セレクト空間 2 (4200 0000h)
- 内部ブート ROM (0010 0000h)

内部ブート ROM から開始するブート・モードの場合、ROM ブート・ローダ (RBL) ソフトウェアは、ブート・シーケンスを完了します。ROM ブート・ローダの詳細については、『Using the TMS320DM643x Bootloader』(資料番号 [SPRAAG0](#)) を参照してください。



## 改訂履歴

本書の旧版からの変更点を表 A-1 に示します。

表 A-1. 資料改訂履歴

参照先	追加 / 変更 / 削除
図 6-1	注を追加。
6.3 節	注を追加。
10.4 節	注を追加。





# 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ないしTexas Instruments Incorporatedを総称してTIJ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TIJ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TIJの標準販売契約約款に従って販売されます。

TIJ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TIJ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。

TIJ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。

TIJ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いません。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るとか、保証もしくは承認をすることを意味しません。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。

TIJ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、か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。TIJ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。

TIJの製品もしくはサービスについてTIJにより示された数値、特性、条件その他のパラメータと異なる、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、かつ不正で誤認を生じさせる行為です。TIJ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。

TIJは、TIの製品が、安全でないことが致命的となる用途ないしアプリケーション(例えば、生命維持装置のように、TI製品に不良があった場合に、その不良により相当な確率で死傷等の重篤な事故が発生するようなもの)に使用されることを認めておりません。但し、お客様とTIJの双方の権限有る役員が書面でそのような使用について明確に合意した場合は除きます。たとえTIJがアプリケーションに関連した情報やサポートを提供したとしても、お客様は、そのようなアプリケーションの安全面及び規制面から見た諸問題を解決するために必要とされる専門的知識及び技術を持ち、かつ、お客様の製品について、またTI製品をそのような安全でないことが致命的となる用途に使用することについて、お客様が全ての法的責任、規制を遵守する責任、及び安全に関する要求事項を満足させる責任を負っていることを認め、かつそのことに同意します。さらに、もし万一、TIの製品がそのような安全でないことが致命的となる用途に使用されたことによって損害が発生し、TIないしその代表者がその損害を賠償した場合は、お客様がTIないしその代表者にその全額の補償をするものとします。

TI製品は、軍事的用途もしくは宇宙航空アプリケーションないし軍事的環境、航空宇宙環境にて使用されるようには設計もされていませんし、使用されることを意図されていません。但し、当該TI製品が、軍需対応グレード品、若しくは「強化プラスチック」製品としてTIが特別に指定した製品である場合は除きます。TIが軍需対応グレード品として指定した製品のみが軍需品の仕様書に合致いたします。お客様は、TIが軍需対応グレード品として指定していない製品を、軍事的用途もしくは軍事的環境下で使用することは、もっぱらお客様の危険負担においてなされるということ、及び、お客様がもっぱら責任をもって、そのような使用に関して必要とされる全ての法的要求事項及び規制上の要求事項を満足させなければならないことを認め、かつ同意します。

TI製品は、自動車用アプリケーションないし自動車の環境において使用されるようには設計されていませんし、また使用されることを意図されていません。但し、TIがISO/TS 16949の要求事項を満たしていると特別に指定したTI製品は除きます。お客様は、お客様が当該TI指定品以外のTI製品を自動車用アプリケーションに使用しても、TIは当該要求事項を満たしていなかったことについて、いかなる責任も負わないことを認め、かつ同意します。

Copyright © 2009, 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。

マウンタやはんだ付け設備等、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 °C、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

はんだ付け時は、最低限260 °C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

は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。は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