

# Cortex<sup>™</sup>-A9 MBIST コントローラ

リビジョン : r2p2

テクニカルリファレンス マニュアル

**ARM<sup>®</sup>**

# Cortex-A9 MBIST コントローラ

## テクニカルリファレンス マニュアル

Copyright © 2008-2010 ARM. All rights reserved.

### リリース情報

本書には次の変更が加えられています。

#### 改訂履歴

日付	変更箇所	公開の有無	変更内容
2008年4月2日	A	公開	r0p0用の最初のリリース
2008年7月10日	B	公開版、限定アクセス	r0p0用の2番目のリリース
2008年12月15日	C	公開版、限定アクセス	r1p0用の最初のリリース
2009年9月23日	D	公開版、限定アクセス	r2p0用の最初のリリース
2009年11月27日	E	公開	r2p0用の2番目のリリース
2010年4月29日	F	公開	r2p2用の最初のリリース

### 著作権

® または™ の付いた用語とロゴは、本著作権条項で特に明記されていない限り、EU および他諸国における ARM® の登録商標または商標です。本書に記載されている他の商標その他の名前は、対応する所有者の商標の場合があります。

本書に記載されている情報の全部または一部、ならびに本書で紹介する製品は、著作権所有者の文書による事前の許可を得ない限り、転用・複製することを禁じます。

本書に説明されている製品は、継続的に開発と改良が行われています。本書で言及されている製品とその利用方法に関する記載事項について、ARM は保証しません。したがって当社では、製品の商品性または目的への適合性を含め、黙示的・明示的に関係なく一切の保証を行いません。

本書は、本製品の利用者をサポートすることだけを目的としています。本書に記載されている情報の使用、情報の誤りまたは省略、あるいは本製品の誤使用によって発生したいかなる損失や損害についても、ARM は一切責任を負いません。

本書における ARM という用語は、「ARM、または該当する場合にはその子会社を含む」という意味で使用されています。

### 機密保持ステータス

本書は非機密扱いであり、本書を使用、複製、および開示する権利は、ARM および ARM が本書を提供した当事者との間で締結した契約の条項に基づいたライセンスの制限により異なります。

### 製品ステータス

本書の情報は最終版であり、開発済み製品に対応しています。

## Web アドレス

<http://www.arm.com>



# 目次

## Cortex-A9 MBIST コントローラ テクニカルリ ファレンス マニュアル

	<b>序章</b>	
	本書について .....	xii
	ご意見・ご質問 .....	xvi
<b>第 1 章</b>	<b>はじめに</b>	
	1.1 MBIST コントローラについて .....	1-2
	1.2 MBIST コントローラインタフェース .....	1-3
	1.3 製品リビジョン .....	1-8
<b>第 2 章</b>	<b>機能の説明</b>	
	2.1 機能概要 .....	2-2
	2.2 機能的な動作 .....	2-16
<b>第 3 章</b>	<b>MBIST 命令レジスタ</b>	
	3.1 MBIST 命令レジスタについて .....	3-2
	3.2 フィールドの説明 .....	3-4

## 第 4 章

### MBIST データログレジスタ

4.1	MBIST データログレジスタについて .....	4-2
4.2	フィールドの説明 .....	4-3

## 付録 A

### 信号の説明

A.1	MBIST コントローラのインタフェース信号 .....	A-2
A.2	その他の信号 .....	A-4

## 付録 B

### リビジョン

### 用語集

# 表一覧

## Cortex-A9 MBIST コントローラ テクニカルリファレンス マニュアル

	改訂履歴 .....	ii
表 1-1	Cortex-A9 プロセッサの MBIST インタフェース信号、パリティなしの構成 .....	1-6
表 1-2	Cortex-A9 プロセッサの MBIST インタフェース信号、パリティ付きの構成 .....	1-6
表 2-1	MBIST インタフェース信号 .....	2-3
表 2-2	MBIST 用の Cortex-A9 信号の設定 .....	2-4
表 2-3	パリティなしの RAM アレイと MBIST コントローラとのインタフェース .....	2-4
表 2-4	データキャッシュ データ RAM のバイト書き込みイネーブル制御 .....	2-7
表 2-5	タグ RAM での MBISTARRAY ビットの使用法 .....	2-7
表 2-6	タグ RAM の制御 .....	2-10
表 2-7	MBISTTX 信号 .....	2-13
表 2-8	MBISTRX 信号 .....	2-14
表 2-9	MBIST コントローラのトップレベル I/O .....	2-15
表 2-10	データログの形式 .....	2-19
表 3-1	パターンフィールドのエンコード .....	3-4
表 3-2	Go/No-Go テストパターン .....	3-6
表 3-3	制御フィールドのエンコード（下位 5 ビット） .....	3-7
表 3-4	読み出しレイテンシフィールドのエンコード .....	3-8
表 3-5	書き込みレイテンシフィールドのエンコード .....	3-8
表 3-6	MBIR[39:36] の CPU へのマッピング .....	3-9
表 3-7	MaxXAddr フィールドのエンコード .....	3-10

表 3-8	MaxYAddr フィールドのエンコード .....	3-10
表 3-9	ArrayEnables フィールドのエンコード .....	3-11
表 3-10	ColumnWidth フィールドのエンコード .....	3-13
表 3-11	CacheSize フィールドのエンコード .....	3-13
表 A-1	パリティなし構成での MBIST コントローラのインタフェース信号 .....	A-2
表 A-2	パリティ付き構成での MBIST コントローラのインタフェース信号 .....	A-2
表 A-3	MBISTARRAY のチップイネーブル (1 ビットのみアクティブ) .....	A-3
表 A-4	その他の信号 .....	A-4
表 B-1	A 版と B 版の相違点 .....	B-1
表 B-2	B 版と C 版の相違点 .....	B-1
表 B-3	C 版と D 版の相違点 .....	B-2
表 B-4	D 版と E 版の相違点 .....	B-2
表 B-5	D 版と F 版の相違点 .....	B-3



## 図一覧

# Cortex-A9 MBIST コントローラ テクニカルリファレンス マニュアル

	タイミング図の表記に使用される記号 .....	xiv
図 1-1	Cortex-A9 MBIST の構成 .....	1-2
図 1-2	パリティなしの構成での MBIST コントローラの配線図 .....	1-3
図 1-3	パリティ付きの構成での MBIST コントローラの配線図 .....	1-4
図 1-4	MBIST の従来のインタフェース方式 .....	1-4
図 1-5	Cortex-A9 プロセッサの MBIST インタフェース .....	1-5
図 2-1	命令キャッシュデータ RAM とデータキャッシュ データ RAM についての MBISTINDATA[63:0] の形式 .....	2-6
図 2-2	命令キャッシュデータ RAM とデータキャッシュ データ RAM についての MBISTOUTDATA[255:0] 形式のデータ出力 .....	2-7
図 2-3	命令タグ RAM での MBISTINDATA[63:0] 形式 .....	2-8
図 2-4	命令タグ RAM での MBISTOUTDATA[255:0] 形式 .....	2-8
図 2-5	データタグ RAM での MBISTINDATA[63:0] 形式 .....	2-8
図 2-6	データタグ RAM での MBISTOUTDATA[255:0] 形式 .....	2-8
図 2-7	SCU タグ RAM での MBISTINDATA[63:0] 形式 .....	2-9
図 2-8	SCU タグ RAM での MBISTOUTDATA[255:0] 形式 .....	2-9
図 2-9	GHB タグ RAM での MBISTINDATA[63:0] 形式 .....	2-9
図 2-10	GHB タグ RAM での MBISTOUTDATA[255:0] 形式 .....	2-9
図 2-11	外部 RAM での MBISTINDATA[63:0] 形式 .....	2-10
図 2-12	外部 RAM での MBISTOUTDATA[255:0] 形式 .....	2-10

図 2-13	BTAC RAM での MBISTINDATA[63:0] 形式 .....	2-11
図 2-14	BTAC RAM での MBISTOUTDATA[255:0] 形式 .....	2-11
図 2-15	TLB RAM での MBISTINDATA[63:0] 形式 .....	2-11
図 2-16	TLB RAM でのデータ出力用 MBISTOUTDATA[255:0] 形式 .....	2-12
図 2-17	GHB タグ RAM での MBISTINDATA[63:0] 形式 .....	2-12
図 2-18	GHB タグ RAM での MBISTOUTDATA[255:0] 形式 .....	2-12
図 2-19	MBIST コントローラブロック .....	2-13
図 2-20	MBIST コントローラ命令のロード .....	2-17
図 2-21	MBIST テストの開始 .....	2-17
図 2-22	MBIST エラーの検出 .....	2-17
図 2-23	データログの取得開始 .....	2-18
図 2-24	データログの取得終了 .....	2-19
図 2-25	ビットマップ データログの取得開始 .....	2-20
図 2-26	ビットマップ データログの取得終了 .....	2-20
図 3-1	MBIST 命令レジスタの制御ユニット .....	3-2
図 3-2	MBIST 命令レジスタのディスパッチユニット .....	3-2
図 4-1	MBIST データログレジスタの形式、パリティ付きの構成 .....	4-2
図 4-2	MBIST データログレジスタの形式、パリティなしの構成 .....	4-2

# 序章

本章では、*Cortex-A9 MBIST* コントローラテクニカルリファレンス マニュアルを紹介します。本章は次のセクションから構成されています。

- 「本書について」 (ページ xii)
- 「ご意見・ご質問」 (ページ xvi)

## 本書について

本書は、Cortex-A9 MBIST コントローラのテクニカルリファレンス マニュアルです。

本書では、MBIST コントローラという一般的な用語を使用した場合、Cortex-A9 MBIST コントローラを意味します。また、Cortex-A9 プロセッサという用語は Cortex-A9 プロセッサファミリを意味します。

## 製品リビジョンステータス

*rn* 識別子は、本書に記載されている製品のリビジョンステータスを示しています。各識別子の意味は次のとおりです。

*rn*           製品が大幅に修正されたことを示しています。

*pn*           製品に小さな修正または変更が加えられたことを示しています。

## 対象読者

本書は、ARM テクノロジを熟知しており、MBIST コントローラを使用して、Cortex-A9 プロセッサにより使用される RAM ブロックのテストを行うハードウェアエンジニアを対象としています。AXI プロトコルは指定されていませんが、AXI に対するある程度の知識を前提としています。

## 本書の使用方法

本書は以下の章に分かれています。

### 第 1 章 はじめに

MBIST テクノロジについて紹介します。

### 第 2 章 機能の説明

Cortex-A9 プロセッサから MBIST コントローラへのインタフェースと、データ RAM およびタグ RAM に対する MBIST テストについて説明します。また、MBIST 命令のロード、MBIST テストの開始、障害検出、データログの取得について、タイミングシーケンスの説明を行います。

### 第 3 章 MBIST 命令レジスタ

MBIST 命令レジスタを使用して、MBIST エンジンの動作モードを構成する方法について説明します。

### 第 4 章 MBIST データログレジスタ

MBIST データログレジスタについて説明します。

## 付録 A 信号の説明

MBIST コントローラの入力および出力信号について説明します。

## 付録 B リビジョン

本書の各版における技術的な変更点について説明します。

**用語集** 本書で使用されている用語の定義について説明します。

## 表記規則

本書では次の表記規則が採用されています。

- 「書体の一般的な規則」
- 「タイミング図」 (ページ xiv)
- 「信号」 (ページ xiv)

### 書体の一般的な規則

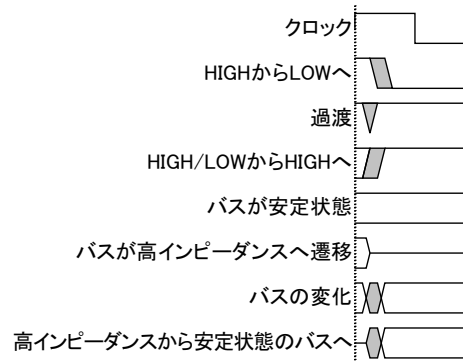
本書で使用されている書体の一般的な規則は次のとおりです。

<b>斜体</b>	重要な注釈の強調、特別な用語の初出時、本書内での相互参照と引用に使用されます。
<b>太字</b>	メニュー名などのインタフェース要素を強調するために太字が使用されます。信号名を示すためにも使用されています。また、必要に応じて説明表の用語にも太字が使用されています。
<code>monospace</code>	コマンド、ファイル名、プログラム名、ソースコードなどの、キーボードから入力可能なテキストを示しています。
<u><code>monospace</code></u>	コマンドまたはオプションに使用可能な略語を示しています。コマンドやオプションの名前を全部入力する代わりに、下線部分のテキストだけを入力してこれらを指定できます。
<i>monospace italic</i>	具体的な値に置き換えられる引数を示しています。
<b>monospace bold</b>	サンプルコード以外で使用されている場合、言語のキーワードを示しています。
<b>&lt; および &gt;</b>	コードまたはコード片の中で不等号の括弧で囲まれている部分は、アセンブラ構文内で置き換え可能なことを示しています。次に例を示します。 <ul style="list-style-type: none"> <li>• MRC p15, 0 &lt;Rd&gt;, &lt;CRn&gt;, &lt;CRm&gt;, &lt;Opcode_2&gt;</li> </ul>

## タイミング図

タイミング図で使用される構成要素を、「タイミング図の表記に使用される記号」に示します。この図と異なる意味で使用されている場合は、その都度明記されています。タイミング図に明示されていないタイミング情報については、推測で判断しないで下さい。

バスと信号で影が付いている部分は定義されていないため、その時点のバスと信号は、影付きの領域内の任意の値を取り得ます。実際のレベルは重要ではなく、通常の動作には影響しません。



### タイミング図の表記に使用される記号

#### 信号

信号の表記規則は次のとおりです。

- 信号レベル**      アサートされる信号のレベルは、その信号がアクティブ HIGH かアクティブ LOW かによって異なります。「アサートされた」とは、次の状態を意味します。
- アクティブ HIGH の信号が HIGH の状態
  - アクティブ LOW の信号が LOW の状態
- 小文字の n**      アクティブ LOW 信号の信号名の最初または最後に付加されます。

#### 参照資料

このセクションでは、ARM Limited やサードパーティが発行している出版物を紹介します。

ARM の出版物は Infocenter, <http://infocenter.arm.com> で参照できます。

## ARM の刊行物

本書には、この製品に固有の情報が記載されています。他の関連情報については、以下の出版物を参照して下さい。

- *Cortex™-A9* テクニカルリファレンス マニュアル (ARM DDI 0388)
- *Cortex-A9 MPCore* テクニカルリファレンス マニュアル (ARM DDI 0407)
- *Cortex-A9* 構成およびサインオフ ガイド (ARM DII 0146)
- *AMBA AXI* プロトコル仕様 (ARM IHI 0022)
- *ARM* アーキテクチャ リファレンスマニュアル、*ARMv7-A* および *ARMv7-R* エディション (ARM DDI 0406)

## ご意見・ご質問

ARM では、MBIST コントローラと本書に関するご意見をお待ちしております。

### 製品に関するご意見

本製品に関するご意見・ご質問がございましたら、次の情報とともに製品購入元までご連絡下さい。

- 製品名
- 製品のリビジョンまたはバージョン
- できるだけ詳細な説明。該当する場合には、現象もご記載下さい。

### 本書の内容に関するご意見

本書の内容に関するご意見がございましたら、電子メールに次の情報をご記入の上、[errata@arm.com](mailto:errata@arm.com) までお寄せ下さい。

- 題名
- 資料番号 ARM DDI 0414FJ
- ご意見のあるページ番号
- ご意見についての簡潔な説明

補足または改善すべき点についての一般的なご意見もお待ちしております。



# 第 1 章

## はじめに

本章では、MBIST コントローラの目的について説明します。本章は次のセクションから構成されています。

- 「MBIST コントローラについて」 (ページ 1-2)
- 「MBIST コントローラインタフェース」 (ページ 1-3)
- 「製品リビジョン」 (ページ 1-8)

## 1.1 MBIST コントローラについて

MBIST は、組み込みメモリをテストするための業界標準の方式です。MBIST は、テストアルゴリズムに従ってメモリへの一連の読み書きを行うことで動作します。業界標準のテストアルゴリズムは数多く存在します。

MBIST コントローラは、セルが正しく動作していることを保証するため、RAM のすべての位置へ読み書きを行う正しいシーケンスを生成します。このために、MBIST が使用するアドレスとデータパスについて、いくつかの追加テストカバレッジが達成されます。この MBIST コントローラは、Cortex-A9 RAM のメモリテストを実行する目的でのみ、Cortex-A9 プロセッサで使用する必要があります。

MBIST モードは他のすべてのモード、例えば SCAN よりも優先されるため、MBIST モードがアクティブなときは、Cortex-A9 RAM は MBIST コントローラからのみアクセス可能です。

MBIST コントローラは、Cortex-A9 プロセッサの MBIST ポートを経由して、Cortex-A9 RAM の MBIST テストを制御します。Cortex-A9 プロセッサの MBIST 構成を、図 1-1 に示します。

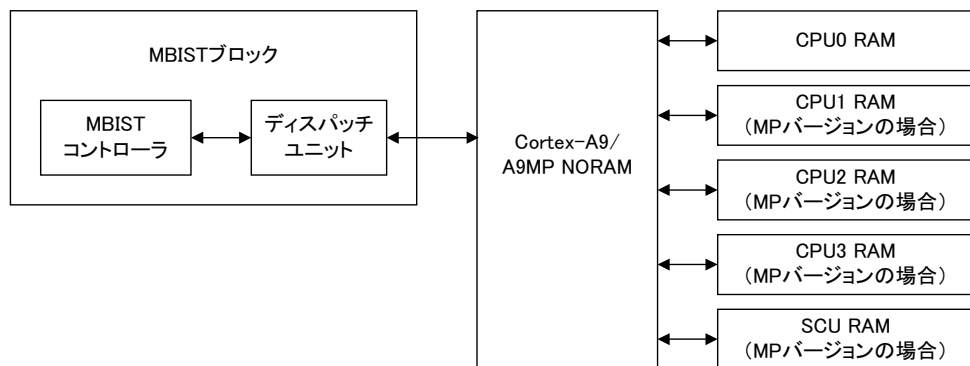


図 1-1 Cortex-A9 MBIST の構成

Cortex-A9 プロセッサのリセット信号は、MBIST テストモードでは HIGH の必要があります。MBIST テストモードは、RAM にアクセスするためにいくつかの機能パスを使用するため、これらのパスのレジスタはリセット状態を終了している必要があります。

ユニプロセッサのリセットシーケンスの説明については、『Cortex-A9 テクニカルリファレンス マニュアル』を参照して下さい。

マルチプロセッサのリセットシーケンスの説明については、『Cortex-A9 MPCore テクニカルリファレンス マニュアル』を参照して下さい。

## 1.2 MBIST コントローラインタフェース

MBIST コントローラから自動テスト設備 (ATE) および Cortex-A9 プロセッサの MBIST インタフェースへの、パリティなしのインタフェース構成を、図 1-2 に示します。

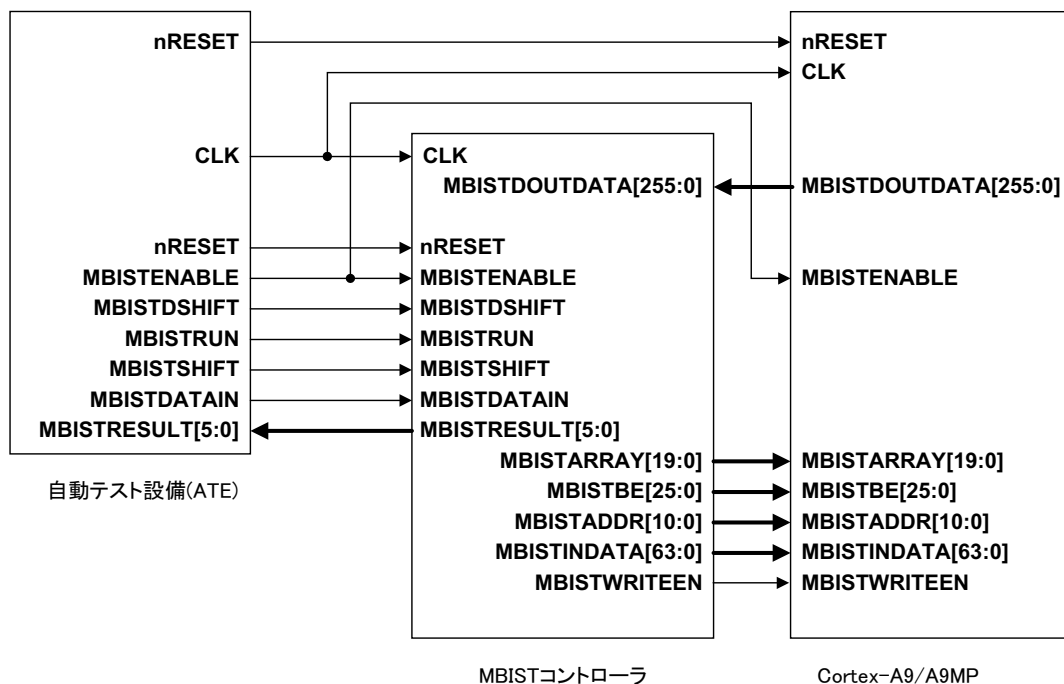


図 1-2 パリティなしの構成での MBIST コントローラの配線図

MBIST コントローラから自動テスト設備 (ATE) および Cortex-A9 プロセッサの MBIST インタフェースへの、パリティ付きのインタフェース構成を、表 1-2 (ページ 1-6) に示します。

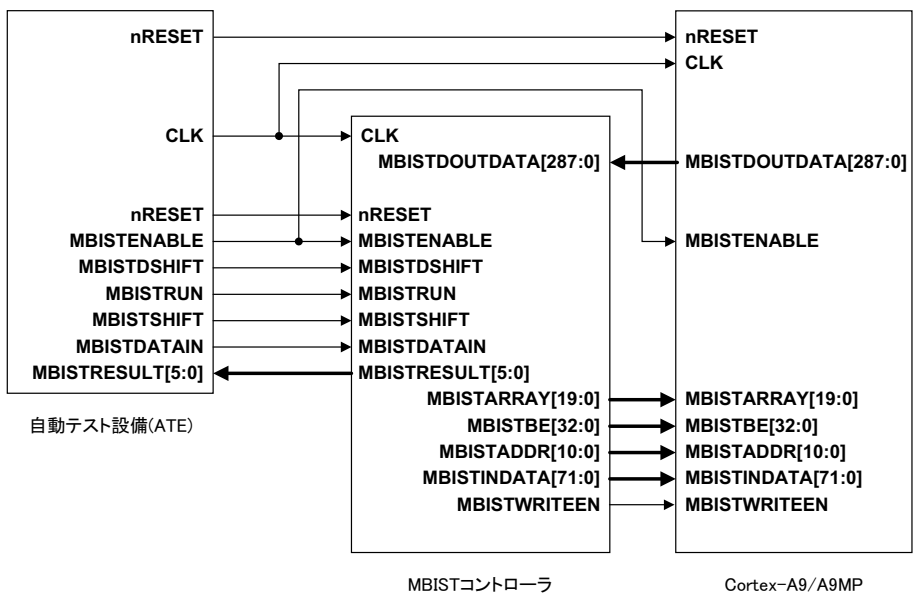


図 1-3 パリティ付きの構成での MBIST コントローラの配線図

MBIST が RAM にアクセスする従来の方法を、図 1-4 に示します。

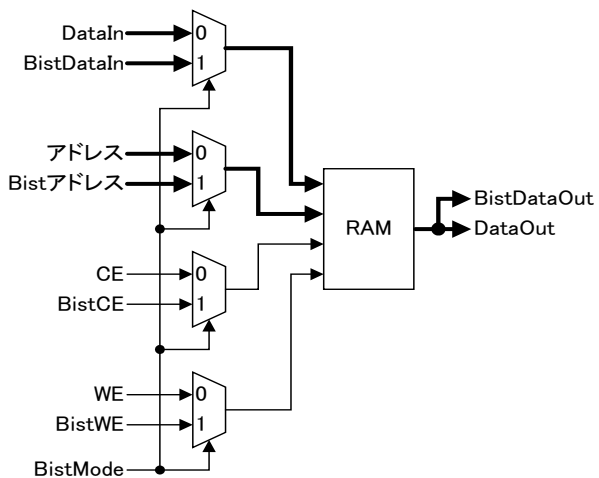


図 1-4 MBIST の従来のインタフェース方式

この方式では最大動作周波数が大幅に低下するため、高パフォーマンスの設計には適していません。その代わりに、MBIST コントローラは既存の機能マルチプレクサへの新たな入力を使用することで、最大動作周波数の低下を回避しています。

RAM アレイへのアクセスに使用される 6 パイプラインステージを、図 1-5 に示します。

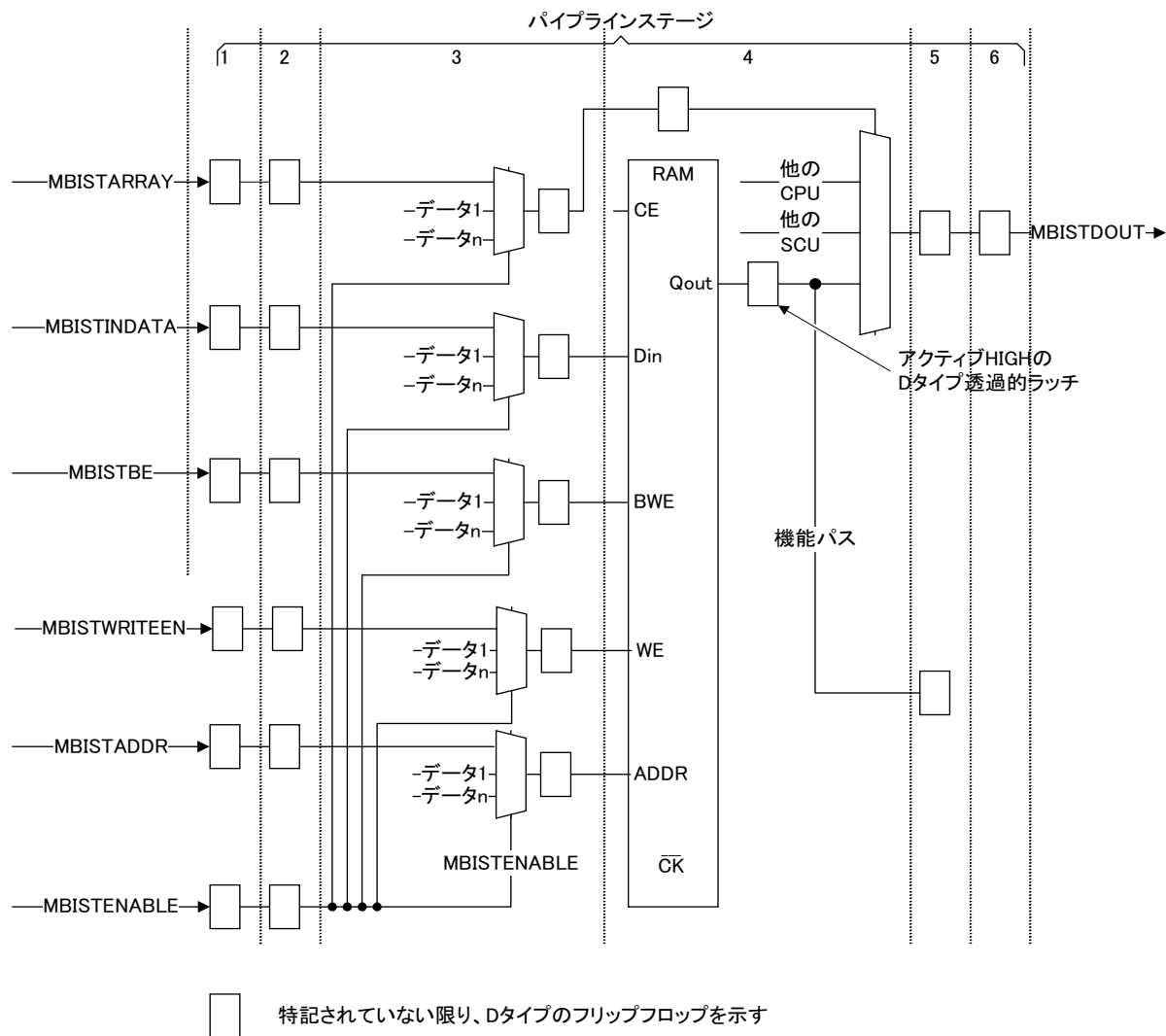


図 1-5 Cortex-A9 プロセッサの MBIST インタフェース

MBIST コントローラは、Cortex-A9 プロセッサの MBIST インタフェースを経由してメモリにアクセスします。パリティなしの構成について、Cortex-A9 プロセッサの MBIST インタフェース信号を、表 1-1 に示します。

表 1-1 Cortex-A9 プロセッサの MBIST インタフェース信号、パリティなしの構成

名前	タイプ	説明
nRESET	入力	グローバルのアクティブ LOW リセット信号
CLK	入力	アクティブ HIGH のクロック信号。このクロックは、Cortex-A9 プロセッサのロジックを駆動します。
MBISTOUTDATA[255:0]	出力	すべてのキャッシュ RAM ブロックからのデータ出力バス
MBISTENABLE	入力	キャッシュ RAM アレイ用の選択信号。この信号は、テスト用にキャッシュ RAM アレイへアクセスするマルチプレクサへの選択入力です。MBISTENABLE がアサートされているとき、マルチプレクサへの他のすべての選択入力に優先します。
MBISTARRAY[19:0]	入力	テスト用の RAM アレイを選択するチップイネーブルで、1 ビットのみがアクティブになります。
MBISTBE[25:0]	入力	すべての RAM アレイ用のグローバル書き込みイネーブル信号
MBISTWRITEEN	入力	グローバル書き込みイネーブル
MBISTADDR[10:0]	入力	キャッシュ RAM アレイ用のアドレス信号
MBISTINDATA[63:0]	入力	RAM アレイへのデータバス。すべての RAM アレイがデータ幅の全体を使用するわけではありません。

パリティ付きの構成について、Cortex-A9 プロセッサの MBIST インタフェース信号を、表 1-2 に示します。

表 1-2 Cortex-A9 プロセッサの MBIST インタフェース信号、パリティ付きの構成

名前	タイプ	説明
nRESET	入力	グローバルのアクティブ LOW リセット信号
CLK	入力	アクティブ HIGH のクロック信号。このクロックは、Cortex-A9 プロセッサのロジックを駆動します。
MBISTOUTDATA[287:0]	出力	すべてのキャッシュ RAM ブロックからのデータ出力バス
MBISTENABLE	入力	キャッシュ RAM アレイ用の選択信号。この信号は、テスト用にキャッシュ RAM アレイへアクセスするマルチプレクサへの選択入力です。MBISTENABLE がアサートされているとき、マルチプレクサへの他のすべての選択入力に優先します。

表 1-2 Cortex-A9 プロセッサの MBIST インタフェース信号、パリティ付きの構成 (続き)

名前	タイプ	説明
MBISTARRAY[19:0]	入力	テスト用の RAM アレイを選択するチップイネーブルで、1 ビットのみがアクティブになります。
MBISTBE[32:0]	入力	すべての RAM アレイ用のグローバル書き込みイネーブル信号
MBISTWRITEEN	入力	グローバル書き込みイネーブル
MBISTADDR[10:0]	入力	キャッシュ RAM アレイ用のアドレス信号
MBISTINDATA[71:0]	入力	RAM アレイへのデータバス。すべての RAM アレイがデータ幅の全体を使用するわけではありません。

—— 注 ——

MBIST コントローラのインタフェースは、Cortex-A9 プロセッサの ATE および MBIST インタフェースの両方と通信します。MBIST コントローラインタフェース信号の説明については、付録 A 信号の説明を参照して下さい。MBIST インタフェースの詳細については、『Cortex-A9 プロセッサ テクニカルリファレンス マニュアル』を参照して下さい。

## 1.3 製品リビジョン

ここでは、この MBIST コントローラのリリース間の機能の違いについて簡単に説明します。

**r0p0 - r1p0** 機能の変更はありません。MBIST コントローラは、プロセッサのリビジョンに対応する正しいリビジョンを使用する必要があります。例えば、**r1p0** プロセッサでは **r1p0** の MBIST コントローラを使用します。

**r1p0 - r2p0** MBIST コントローラは、プロセッサのリビジョンに対応する正しいリビジョンを使用する必要があります。**r2p0** プロセッサでは **r2p0** の MBIST コントローラを使用します。

**r2p0 - r2p2** 説明書の更新のみ。



## 第 2 章 機能の説明

本章では、MBIST コントローラ的设计例の機能概要と、その機能的な動作について説明します。ロード命令、MBIST エンジンの開始、障害検出、データログの取得について、タイミングシーケンスが含まれています。本章は次のセクションから構成されています。

- 「機能概要」 (ページ 2-2)
- 「機能的な動作」 (ページ 2-16)

## 2.1 機能概要

ここでは、MBIST コントローラと、テスト対象となる RAM との間のインタフェースについて説明します。

- 「MBIST コントローラインタフェース」
- 「MBISTINDATA と MBISTOUTDATA のマッピング」 (ページ 2-6)
- 「MBIST コントローラの実装」 (ページ 2-12)

### 2.1.1 MBIST コントローラインタフェース

MBIST コントローラには、MBIST ポートが 1 つ存在します。付録 A 信号の説明を参照して下さい。MBIST コントローラから同時にアクセス可能な RAM は 1 セットのみです。

Cortex-A9 MPCore 実装では、MBIST モードで次に示すピンを LOW に固定する必要があります。

- **DBGEN**
- **NIDEN**
- **SPIDEN**
- **SPNIDEN**
- **EDBGRQ**
- **DBGRESTART**
- **PSELDBG**
- **CPUCLKOFF**
- **PERIPHCLKEN**
- **ACLKENS**
- **INCLKENM0**
- **INCLKENM1**
- **OUTCLKENM0**

**MAXCLKLATENCY** は、いずれかの値に駆動する必要があります。言い換えれば、非駆動にすることはできません。

Cortex-A9 ユニプロセッサ実装では、MBIST モードで次に示すピンを LOW に固定する必要があります。

- **DBGEN**
- **NIDEN**
- **SPIDEN**
- **SPNIDEN**
- **EDBGRQ**
- **DBGRESTART**

- **PSELDBG**
- **ACLKENM0**
- **ACLKENM1**
- **AWREADYM0**
- **WREADYM0**
- **BVALIDM0**
- **RVALIDM0**

**MAXCLKLATENCY** は、いずれかの値に駆動する必要があります。言い換えれば、非駆動にすることはできません。

MBIST インタフェース信号を、表 2-1 に示します。

表 2-1 MBIST インタフェース信号

名前	タイプ	説明
<b>nRESET</b>	入力	グローバルのアクティブ LOW リセット信号
<b>CLK</b>	入力	アクティブ HIGH のクロック信号。このクロックは、Cortex-A9 プロセッサのロジックを駆動します。
<b>MBISTOUTDATA[255:0]</b>	出力	すべてのキャッシュ RAM ブロックからのデータ出力バス
<b>MBISTENABLE</b>	入力	キャッシュ RAM アレイ用の選択信号。この信号は、テスト用にキャッシュ RAM アレイへアクセスするマルチプレクサへの選択入力です。 <b>MBISTENABLE</b> がアサートされているとき、マルチプレクサへの他のすべての選択入りに優先します。
<b>MBISTARRAY[19:0]</b>	入力	テスト用の RAM アレイを選択するチップイネーブルで、1 ビットのみがアクティブになります。
<b>MBISTBE[25:0]</b>	入力	すべての RAM アレイ用のグローバル書き込みイネーブル信号
<b>MBISTWRITEEN</b>	入力	グローバル書き込みイネーブル
<b>MBISTADDR[10:0]</b>	入力	キャッシュ RAM アレイ用のアドレス信号
<b>MBISTINDATA[63:0]</b>	入力	RAM アレイへのデータバス。すべての RAM アレイがデータ幅の全体を使用するわけではありません。
<b>MBISTRUN</b>	入力	メモリテストを開始します。
<b>MBISTDSHIFT</b>	入力	選択されたディスパッチユニット データログレジスタのシフトイネーブル
<b>MBISTSHIFT</b>	入力	<b>MBIST</b> 命令レジスタ (MBIR) の連続ロードを行います。

MBIST コントローラを使用して、Cortex-A9 プロセッサのコンパイル済み RAM をテストできます。また、独自の MBIST コントローラを設計することもできます。

休眠 / 電力オフラップが実装されている場合、Cortex-A9 プロセッサで MBIST が正しく実行されるようにするには、Cortex-A9 プロセッサインタフェースの信号を、表 2-2 に示されているように設定する必要があります。

対称 CPU 構成の RAM は並列にテストします。例えば、対称設計のすべてのタグ RAM を並列にテストできます。各 CPU に対して、**BISTINDATA** を使用して同じデータを送信し、結果は **BISTOUTDATA** で並列に読み出します。

非統一構成では、各 CPU の RAM を別々にテストする必要があります。

表 2-2 MBIST 用の Cortex-A9 信号の設定

信号名	設定
CPURAMCLAMP[3:0]	4'b0000
CPUCLAMP[3:0]	4'b0000
RVALIDM1	1'b0
SCURAMCLAMP	1'b0

パリティなしの構成について、MBIST コントローラと、テスト対象の RAM との間のインタフェースを、表 2-3 に示します。

表 2-3 パリティなしの RAM アレイと MBIST コントローラとのインタフェース

RAM 名	MBISTARRAY ビット	MBISTINDATA ビット	MBISTBE ビット	MBISTOUTDATA ビット	最大アドレスビット
SCU タグ RAM ウェイ 3	[19]	[22:0]	[22:0]	[54:32]	[8:0]
SCU タグ RAM ウェイ 2	[19]	[22:0]	[22:0]	[22:0]	[8:0]
SCU タグ RAM ウェイ 1	[18]	[22:0]	[22:0]	[54:32]	[8:0]
SCU タグ RAM ウェイ 0	[18]	[22:0]	[22:0]	[22:0]	[8:0]
ドータ RAM	[17]	[11:0]	[11:0]	[11:0]	[8:0]
データキャッシュデータ RAM ウェイ 3 (アレイ 3、7)	[16]	[63:0]	[7:0]	[63:0]	[10:0]
データキャッシュデータ RAM ウェイ 2 (アレイ 2、6)	[15]	[63:0]	[7:0]	[63:0]	[10:0]
データキャッシュデータ RAM ウェイ 1 (アレイ 1、5)	[14]	[63:0]	[7:0]	[63:0]	[10:0]
データキャッシュデータ RAM ウェイ 0 (アレイ 0、4)	[13]	[63:0]	[7:0]	[63:0]	[10:0]

表 2-3 パリティなしの RAM アレイと MBIST コントローラとのインタフェース (続き)

RAM 名	MBISTARRAY ビット	MBISTINDATA ビット	MBISTBE ビット	MBISTOUTDATA ビット	最大アドレ スビット
データキャッシュ タグ RAM アレイ 3	[12]	[25:0]	[25:0]	[57:32]	[8:0]
データキャッシュ タグ RAM アレイ 2	[12]	[25:0]	[25:0]	[25:0]	[8:0]
データキャッシュ タグ RAM アレイ 1	[11]	[25:0]	[25:0]	[57:32]	[8:0]
データキャッシュ タグ RAM アレイ 0	[11]	[25:0]	[25:0]	[25:0]	[8:0]
TLB RAM アレイ 1	[10]	[60:0]	-	[60:0]	[5:0]
TLB RAM アレイ 0	[9]	[60:0]	-	[60:0]	[5:0]
グローバル履歴バッファアレ イ 0、1、2、3	[8]	[15:0]	[15:0]	[15:0]	[8:0]
命令キャッシュデータ RAM アレイ 7 (ウエイ 3 HIGH)	[7]	[63:32]	-	[63:32]	[10:0]
命令キャッシュデータ RAM アレイ 6 (ウエイ 3 LOW)	[7]	[31:0]	-	[31:0]	[10:0]
命令キャッシュデータ RAM アレイ 5 (ウエイ 2 HIGH)	[6]	[63:32]	-	[63:32]	[10:0]
命令キャッシュデータ RAM アレイ 4 (ウエイ 2 LOW)	[6]	[31:0]	-	[31:0]	[10:0]
命令キャッシュデータ RAM アレイ 3 (ウエイ 1 HIGH)	[5]	[63:32]	-	[63:32]	[10:0]
命令キャッシュデータ RAM アレイ 2 (ウエイ 1 LOW)	[5]	[31:0]	-	[31:0]	[10:0]
命令キャッシュデータ RAM アレイ 1 (ウエイ 0 HIGH)	[4]	[63:32]	-	[63:32]	[10:0]
命令キャッシュデータ RAM アレイ 0 (ウエイ 0 LOW)	[4]	[31:0]	-	[31:0]	[10:0]
命令キャッシュタグ RAM アレイ 3	[3]	[21:0]	-	[53:32]	[8:0]
命令キャッシュタグ RAM アレイ 2	[3]	[21:0]	-	[21:0]	[8:0]
命令キャッシュタグ RAM アレイ 1	[2]	[21:0]	-	[53:32]	[8:0]

表 2-3 パリティなしの RAM アレイと MBIST コントローラとのインタフェース (続き)

RAM 名	MBISTARRAY ビット	MBISTINDATA ビット	MBISTBE ビット	MBISTOUTDATA ビット	最大アドレ スビット
命令キャッシュタグ RAM アレイ 0	[2]	[21:0]	-	[21:0]	[8:0]
BTAC RAM ターゲットアレイ 1	[1]	[63:32]	-	[63:32]	[7:0]
BTAC RAM 制御アレイ 1	[1]	[27:0]	-	[27:0]	[7:0]
BTAC RAM ターゲットアレイ 0	[0]	[63:32]	-	[63:32]	[7:0]
BTAC RAM 制御アレイ 0	[0]	[27:0]	-	[27:0]	[7:0]

### 2.1.2 MBISTINDATA と MBISTOUTDATA のマッピング

ここでは、各種の RAM アレイが **MBISTINDATA** と **MBISTOUTDATA** にどのようにマッピングされるかについて説明します。

- 「命令キャッシュデータ RAM およびデータキャッシュ データ RAM」
- 「命令タグ、データタグ、SCU タグの RAM」 (ページ 2-7)
- 「外部 RAM」 (ページ 2-10)
- 「分岐先アドレスキャッシュ RAM」 (ページ 2-10)
- 「TLB RAM」 (ページ 2-11)
- 「グローバル履歴バッファ RAM」 (ページ 2-12)

#### 命令キャッシュデータ RAM およびデータキャッシュ データ RAM

命令キャッシュデータ RAM は、ビット **MBISTARRAY[7:4]** を使用して選択されます。データキャッシュ データ RAM は、ビット **MBISTARRAY[16:13]** を使用して選択されます。このキャッシュはバイト書き込み可能です。すべての書き込みイネーブルビットは、別々に制御可能な必要があります。

両方のキャッシュは、関連するデータを格納するための 8 つの RAM アレイで構成されます。詳細については、「パリティなしの RAM アレイと MBIST コントローラとのインタフェース」 (ページ 2-4) を参照して下さい。両方のデータキャッシュについて、2 つのアレイが並列にテストされ、それぞれの CPU に同じデータが送信されます。n = 0 から n = 3 までについて、データ入力およびデータ出力パスは、図 2-1 および図 2-2 (ページ 2-7) に示されているようにマッピングされます。

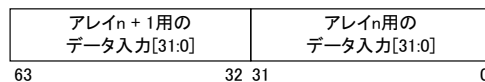


図 2-1 命令キャッシュデータ RAM とデータキャッシュ データ RAM についての MBISTINDATA[63:0] の形式

パリティ付きの構成では、**MBISTINDATA** の幅は **MBISTINDATA[71:0]** です。

CPU3用のデータ出力 [63:0]		CPU2用のデータ出力 [63:0]		CPU1用のデータ出力 [63:0]		CPU0用のデータ出力 [63:0]		
アレイン+1 用のデータ 出力[31:0]	アレイン 用のデータ 出力[31:0]	アレイン+1 用のデータ 出力[31:0]	アレイン 用のデータ 出力[31:0]	アレイン+1 用のデータ 出力[31:0]	アレイン 用のデータ 出力[31:0]	アレイン+1 用のデータ 出力[31:0]	アレイン 用のデータ 出力[31:0]	
255	224 223	192	191 160 159	128	127 96 95	64	63 32 31	0

**図 2-2 命令キャッシュデータ RAM とデータキャッシュ データ RAM についての  
MBISTOUTDATA[255:0] 形式のデータ出力**

パリティ付きの構成では、**MBISTOUTDATA** の幅は **MBISTOUTDATA[287:0]** です。

命令キャッシュデータ RAM はワード書き込み可能で、BIST モードでは **MBISTWRITEEN** により制御されます。データキャッシュ データ RAM はバイト書き込み可能で、表 2-4 に示されているように **MBISTBE[3:0]** により制御されます。

**表 2-4 データキャッシュ データ RAM のバイト書き込みイネーブル制御**

MBISTBE ビット	説明
0	バイト 0、ビット [7:0]
1	バイト 1、ビット [15:8]
2	バイト 2、ビット [23:16]
3	バイト 3、ビット [31:24]

### 命令タグ、データタグ、SCU タグの RAM

命令タグ RAM、データタグ RAM、SCU タグ RAM は、いずれも CPU ごとに 4 つのアレイで構成されます。データタグ RAM と SCU タグ RAM の構造は同じです。各 CPU について、2 つのアレイが並列にテストされます。

各タグ RAM の選択に使用される **MBISTARRAY** ビットを、表 2-5 に示します。

**表 2-5 タグ RAM での MBISTARRAY ビットの使用方法**

MBISTARRAY ビット	説明
[3:2]	命令タグアレイを選択
[12:11]	データタグアレイを選択
[19:18]	SCU タグアレイを選択

命令タグ RAM での **MBISTINDATA** 上のデータマッピングを、図 2-3 に示します。

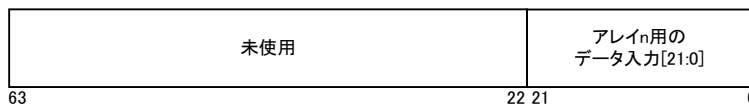


図 2-3 命令タグ RAM での **MBISTINDATA**[63:0] 形式

命令タグ RAM での **MBISTOUTDATA** 上のデータマッピングを、図 2-4 に示します。

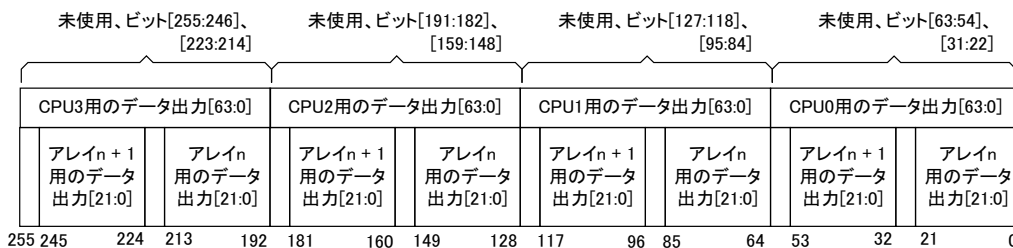


図 2-4 命令タグ RAM での **MBISTOUTDATA**[255:0] 形式

データタグ RAM での、**MBISTINDATA** および **MBISTOUTDATA** バスへのデータマッピングを、図 2-5 および図 2-6 に示します。

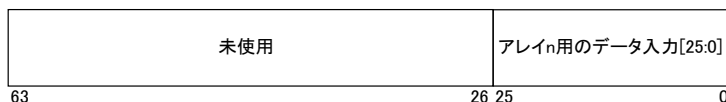


図 2-5 データタグ RAM での **MBISTINDATA**[63:0] 形式

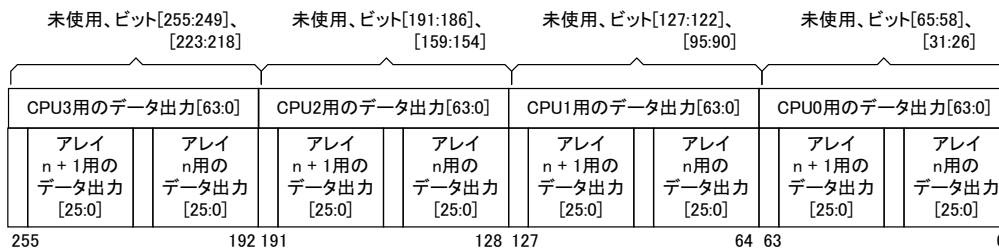


図 2-6 データタグ RAM での **MBISTOUTDATA**[255:0] 形式



SCU タグ RAM での MBISTINDATA[63:0] 上のデータマッピングを、図 2-7 に示します。

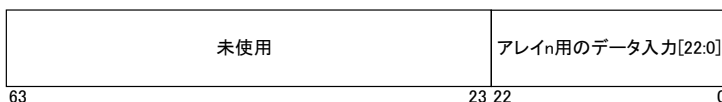


図 2-7 SCU タグ RAM での MBISTINDATA[63:0] 形式

SCU タグ RAM での MBISTOUTDATA[255:0] 上のデータマッピングを、図 2-8 に示します。

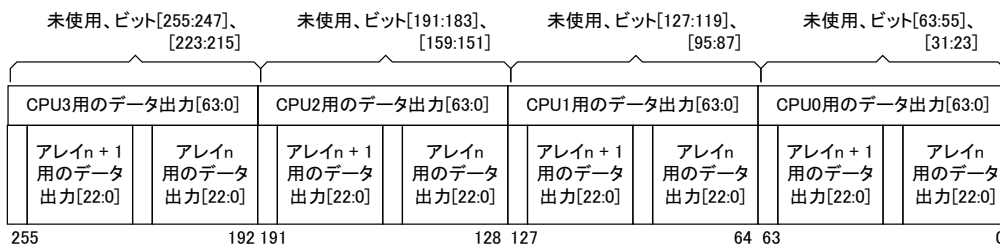


図 2-8 SCU タグ RAM での MBISTOUTDATA[255:0] 形式

GHB タグ RAM での MBISTINDATA[63:0] 上のデータマッピングを、図 2-9 に示します。

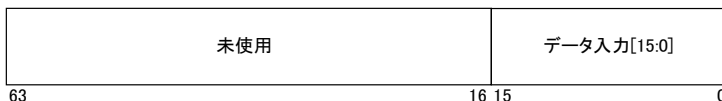


図 2-9 GHB タグ RAM での MBISTINDATA[63:0] 形式

GHB タグ RAM での MBISTOUTDATA[255:0] 上のデータマッピングを、図 2-10 に示します。

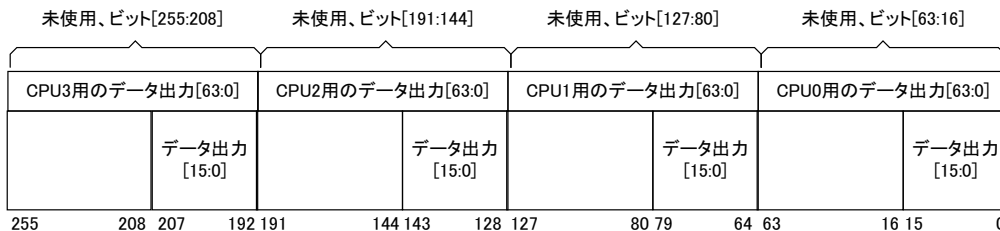


図 2-10 GHB タグ RAM での MBISTOUTDATA[255:0] 形式

タグ RAM の制御に使用される **MBISTBE** ビットを、表 2-6 に示します。

表 2-6 タグ RAM の制御

RAM タイプ	書き込み可能	MBISTBE ビット
SCU タグ RAM	ビット書き込み可能	[22:0]
データタグ RAM	ビット書き込み可能	[25:0]

## 外部 RAM

外部 RAM は、CPU ごとに 1 つのアレイで構成されます。この RAM はビット書き込み可能です。ビット書き込み可能は、別々に制御できる必要があります。

**MBISTARRAY[17]** は、外部 RAM アレイを選択します。

外部 RAM での **MBISTINDATA** 上のデータマッピングを、図 2-11 に示します。

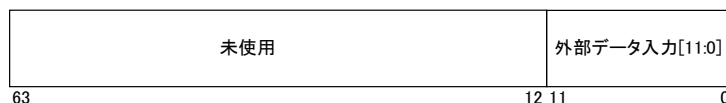


図 2-11 外部 RAM での **MBISTINDATA[63:0]** 形式

外部 RAM での **MBISTOUTDATA[255:0]** 上のデータマッピングを、図 2-12 に示します。



図 2-12 外部 RAM での **MBISTOUTDATA[255:0]** 形式

## 分岐先アドレスキャッシュ RAM

分岐先アドレスキャッシュ (BTAC) RAM は 2 つのアレイで構成され、1 つは制御用、もう 1 つはターゲットです。ターゲットアレイは常に 32 ビット幅です。

**MBISTARRAY[1:0]** は BTAC アレイを選択します。これらのアレイはワード書き込み可能で、BIST モードでは **MBISTWRITEEN** により制御されます。

BTAC RAM のデータマッピングを、図 2-13 に示します。

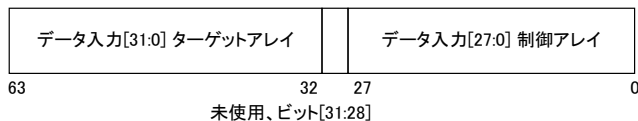


図 2-13 BTAC RAM での MBISTINDATA[63:0] 形式

BTAC RAM での MBISTOUTDATA[255:0] 上のデータマッピングを、図 2-14 に示します。

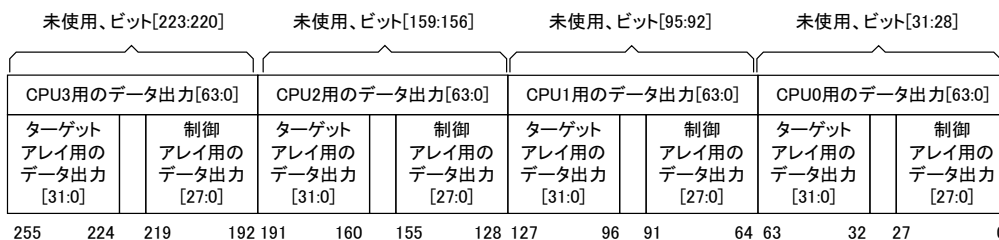


図 2-14 BTAC RAM での MBISTOUTDATA[255:0] 形式

## TLB RAM

TLB RAM は、2つのアレイで構成されます。**MBISTARRAY[10:9]** は、これらのアレイを選択します。TLB アレイはワード書き込み可能で、BIST モードでは **MBISTWRITEEN** により制御されます。

TLB RAM での **MBISTINDATA[63:0]** 上のデータマッピングを、図 2-15 に示します。

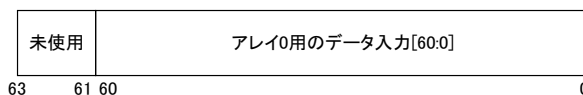


図 2-15 TLB RAM での MBISTINDATA[63:0] 形式

TLB RAM の **MBISTOUTDATA** データマッピングを、図 2-16 (ページ 2-12) に示します。

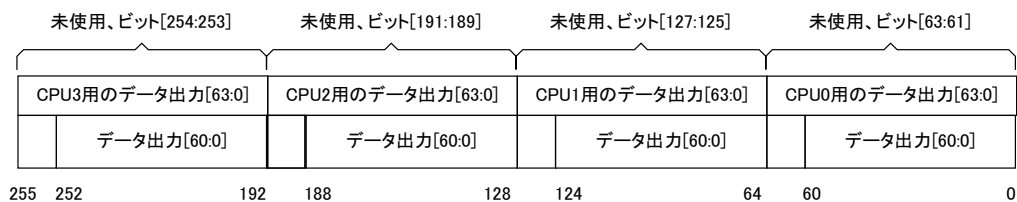


図 2-16 TLB RAM でのデータ出力用 MBISTOUTDATA[255:0] 形式

### グローバル履歴バッファ RAM

グローバル履歴バッファ (GHB) RAM は 4 つのアレイで構成され、各アレイは 4 ビット幅です。アドレス空間は 512 ワードです。MBISTARRAY[8] は、GHB アレイを選択します。

GHB アレイはビット書き込み可能で、BIST モードでは MBISTBE[11:0] により制御されます。

GHB タグ RAM での MBISTINDATA 上のデータマッピングを、図 2-17 に示します。

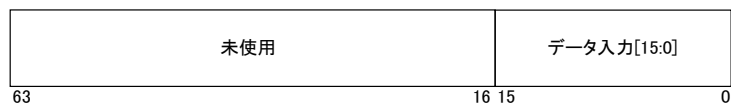


図 2-17 GHB タグ RAM での MBISTINDATA[63:0] 形式

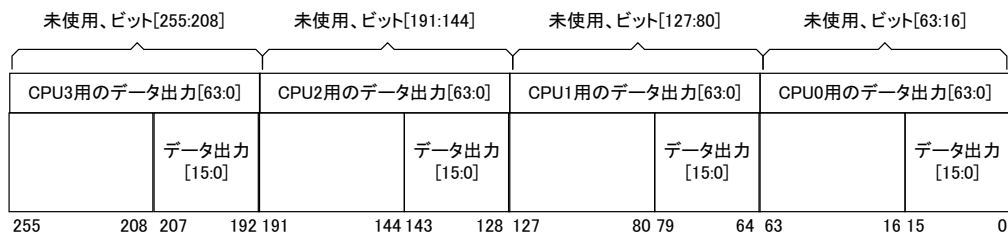


図 2-18 GHB タグ RAM での MBISTOUTDATA[255:0] 形式

### 2.1.3 MBIST コントローラの実装

図 2-19 (ページ 2-13) に示されている MBIST コントローラブロックには、2 つの主要なブロックが含まれます。

- MBIST コントローラ

- ディスパッチユニット

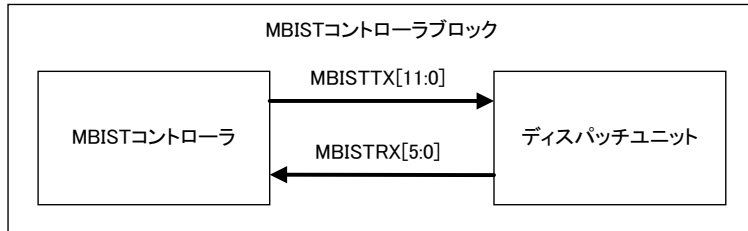


図 2-19 MBIST コントローラブロック

ここでは、次のトピックについて説明します。

- 「MBIST コントローラとディスパッチユニットのインターフェース」
- 「MBIST コントローラブロックのトップレベル I/O」 (ページ 2-15)

### MBIST コントローラとディスパッチユニットのインターフェース

MBIST コントローラとディスパッチユニットは、以下の信号を使用して通信を行います。

#### MBISTTX[11:0]

この信号は MBIST コントローラの出力で、ディスパッチユニットに送信されます。この信号を表 2-7 に示します。

表 2-7 MBISTTX 信号

MBISTTX ビット	説明
0	アドレスのリセット
1	アドレスのインクリメント
2	アクセス犠牲行。Bang パターンで使用されます。
3	データ / 命令データ入力反転
4	チェッカーボードデータ
5	データ書き込み
6	データ読み出し
7	Yfast/nXfast
8	方向

表 2-7 MBISTTX 信号 (続き)

MBISTTX ビット	説明
9	ビットマップモードのイネーブル
10	GO/NOGO データワード選択のインクリメント
11	レイテンシストール制御

命令シフトが可能なとき、BIST エンジンの 2 つのパート間のデータシフトはビット 3 に存在します。実行テストモードでは、このビットは反転データ情報として使用されます。

**MBISTTX[11:0]** インタフェースは ARM 固有で、MBIST コントローラでのみ使用されることを意図しています。

### MBISTRX[5:0]

この信号はディスパッチユニットの出力で、MBIST コントローラに送信されます。**MBISTRX[5:0]** の動作は ARM 固有で、MBIST コントローラでのみ使用されることを意図しています。アドレス失効信号は、行および列のアドレスカウンタが両方とも失効したときにセットされます。この信号を表 2-8 に示します。

表 2-8 MBISTRX 信号

MBISTRX ビット	説明
0	リアルタイム エラーフラグ
1	シャドウパイプラインが空です。
2	CPU0 アドレス / 命令データ出力 / エラーデータ出力
3	CPU1 アドレス / 命令データ出力 / エラーデータ出力
4	CPU2 アドレス / 命令データ出力 / エラーデータ出力
5	CPU3 アドレス / 命令データ出力 / エラーデータ出力

## MBIST コントローラブロックのトップレベル I/O

MBIST コントローラのトップレベル I/O には、Cortex-A9 プロセッサインタフェースが含まれています。付録 A 信号の説明と、表 2-9 に示されている入出力を参照して下さい。

表 2-9 MBIST コントローラのトップレベル I/O

信号	方向	機能	値、MBIST モード	値、機能モード
MBISTDATAIN	入力	シリアルデータ入力	トグル	0
MBISTDSHIFT	入力	データログシフト	トグル	0
nRESET	入力	MBIST リセット	トグル	0 <sup>a</sup>
MBISTRESULT[5:0]	出力	出力ステータスバス	ストロブ	-
MBISTRUN	入力	MBIST テストの実行	トグル	0
MBISTSHIFT	入力	命令シフト	トグル	0
MBISTENABLE	入力	MBIST パスイネーブル	トグル	0
SE	入力	ATPG 信号	0	0

a. 機能モードでは、nRESET および MBISTENABLE は LOW の必要があります。

以下の信号については、追加情報が存在します。

**SE** 複数ロードの自動テストパターンジェネレータ (ATPG) を実行するとき、または IDDQ テストを実行するときは、アレイの状態を保持する必要があります。MBIST テストを実行して、アレイを必要なバックグラウンドに初期化した後で、ATPG テスト手順では、ロード / アンロードに加えて、すべてのテストセットアップサイクル中に SE をアサートする必要があります。IDDQ キャプチャサイクル中のクロックでは、アレイチップ選択信号が制限されている必要があります。

### MBISTRESULT[5:0]

テスト中に、MBISTRESULT[1] 信号は障害を示します。命令レジスタのエンジン制御選択のビット [5] を構成することで、2つのモードで動作可能です。ビット [5] がセットされている場合、比較が失敗するごとに MBISTRESULT[1] 信号が 1 サイクルアサートされます。ビット [5] がセットされていない場合、MBISTRESULT[1] 信号はスティッキーで、最初に障害が発生してから、テスト完了までアサートされます。

テストの完了時に、MBISTRESULT[0] 信号は HIGH になります。MBISTRESULT[5:2] 信号は、テストでプロセッサのアドレス失効が起きたことを示し、テストのアルゴリズム内のシーケンシャルな進行を測定するために使用できます。

## 2.2 機能的な動作

次に示すサブセクションでは、機能的な動作について説明します。

- 「タイミング」
- 「ビットマップモード」 (ページ 2-19)

### 2.2.1 タイミング

MBIST コントローラの動作は、各テストの開始時に連続的にロードされる 58 ビットの命令により制御されます。この命令を書き込む方法を、第 3 章 *MBIST 命令レジスタ* に示します。

このセクションのタイミング図は、2 つの異なる速度で実行されるクロックを示しています。

- 低速のクロックは、ATE によって駆動されるクロックに関連します。
- 高速のクロックは、オンチップのフェーズロックループ (PLL) によって駆動されるクロックに関連します。

オンチップ PLL が存在しない場合、両方のクロックが、ATE により駆動されるクロックに関連します。

プロセッサのタイミング情報については、『*Cortex-A9 テクニカルリファレンスマニュアル*』および『*Cortex-A9 MPCore テクニカルリファレンスマニュアル*』を参照して下さい。

以下のサブセクションのタイミング図は、MBIST コントローラを動作させる手順を示しています。

- 「命令のロード」
- 「MBIST の開始」 (ページ 2-17)
- 「エラー検出」 (ページ 2-17)
- 「データログの取得」 (ページ 2-18)

#### 命令のロード

MBIST 命令をロードするには、**MBISTSHIFT** を HIGH に駆動します。次の立ち上がりクロックエッジで、図 2-20 (ページ 2-17) に示すように、58 ビットのシフトシーケンスが開始されます。ATE からのデータ入力を可能にするため、PLL はバイパスモードで、クロックはテスト周波数では実行されません。



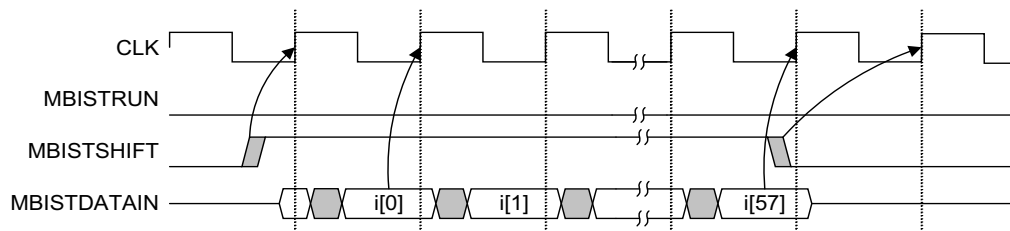


図 2-20 MBIST コントローラ命令のロード

### MBIST の開始

MBIST 命令をロードした後で、**MBISTSHIFT** を LOW に駆動し、**CLK** を非稼働にします。**CLK** が非稼働な状態で、**MBISTRUN** を HIGH に駆動し、**MBISTRUN** セットアップ時間の後で、図 2-21 に示すように、PLL をテスト周波数で開始します。

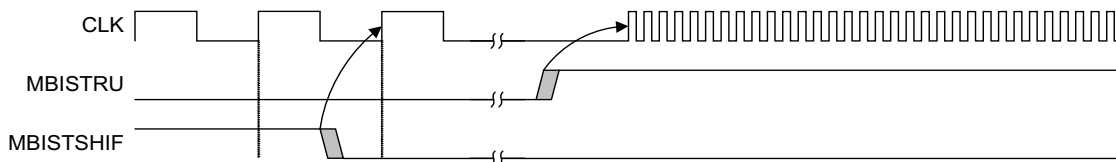


図 2-21 MBIST テストの開始

### エラー検出

**MBISTRESULT[1]** フラグは、図 2-22 に示すように、コントローラがエラーを検出してから 2 **CLK** サイクル後に HIGH に移行します。スティッキーエラーが可能な場合、このフラグは HIGH のまま維持されます。エラー時停止が可能な場合、**MBISTRESULT[0]** フラグが 2 サイクル後に HIGH に移行します。

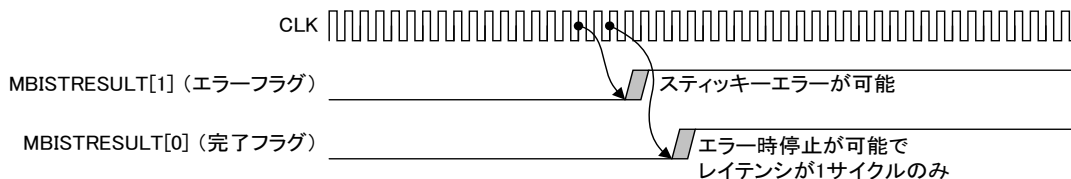


図 2-22 MBIST エラーの検出

### 注

ATE がテスト速度でエラーを観測できることを保証するには、MBIST 命令にスティッキーエラーを指定します。「制御フィールド、**MBIR[51:46]**」(ページ 3-7) を参照して下さい。

## データログの取得

テスト時に、MBIST コントローラは最初に検出されたエラーを自動的にログに記録します。必要なら、テストの終了時にデータログを取得し、エラー統計を生成できます。データログを取得する方法を、図 2-23 と図 2-24（ページ 2-19）に示します。

### 注

**MBISTRESULT[2]** は、CPU0 の命令およびデータログのシリアルデータ出力です。

**MBISTRESULT[3]** は、CPU1 の命令およびデータログのシリアルデータ出力です。

**MBISTRESULT[4]** は、CPU2 の命令およびデータログのシリアルデータ出力です。

**MBISTRESULT[5]** は、CPU3 の命令およびデータログのシリアルデータ出力です。

図 2-23 に示すように、**MBISTRESULT[0]** フラグが HIGH に移行した後で、PLL をバイパスモードに変更し、**MBISTRUN** を LOW に駆動してテストを停止します。**MBISTRESULT[5:2]** でデータログのシフトアウトを開始するには、**MBISTDSHIFT** を HIGH に駆動します。**MBISTRESULT[0]** フラグは、**MBISTRUN** が LOW に移行した 2 サイクル後に LOW に移行します。**MBISTRESULT[5:2]** でのデータのシフトアウトは、**MBISTDSHIFT** が HIGH に移行した 2 サイクル後に開始されます。

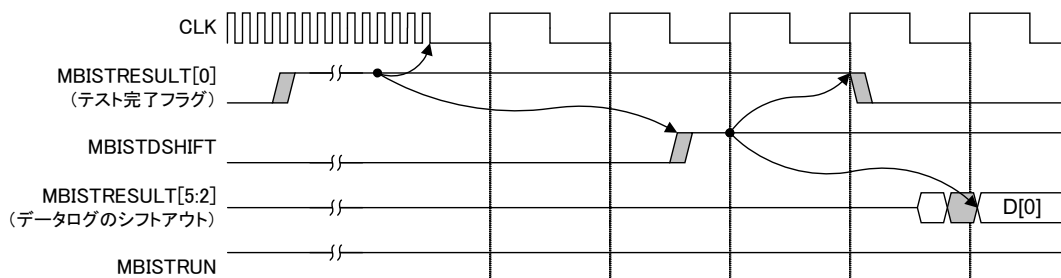


図 2-23 データログの取得開始

最後のデータログビットがシフトアウトされたら、図 2-24（ページ 2-19）に示すように、**MBISTDSHIFT** を LOW に駆動します。

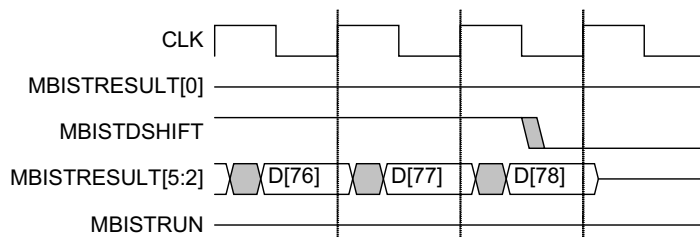


図 2-24 データログの取得終了

データログの形式を、表 2-10 に示します。

表 2-10 データログの形式

ビット	説明
[78:68]	エラーの場所のアドレス
[67:4]	エラーのデータビット。これらのビットは、エラーのあるビットについてセットされ、合格したビットについてはクリアされます。
[3:0]	テストで使用されたデータシード。『DataWord フィールド、MBIR[27:24]』(ページ 3-11) を参照。

データログに含まれているアドレスは、Cortex-A9 プロセッサの MBIST インタフェースの **MBISTADDR[10:0]** ポートに出力されるのと同じ、エラーの発生した位置の完全なアドレスを指しています。

第 4 章 *MBIST* データログレジスタも参照して下さい。

## 2.2.2 ビットマップモード

ビットマップモードでは、RAM でエラーの発生したすべての場所を識別できます。図 2-25 (ページ 2-20) に示すように、エラーが発生するごとに、コントローラは現在のテストの実行を停止し、データログのシフトアウト開始を待ちます。

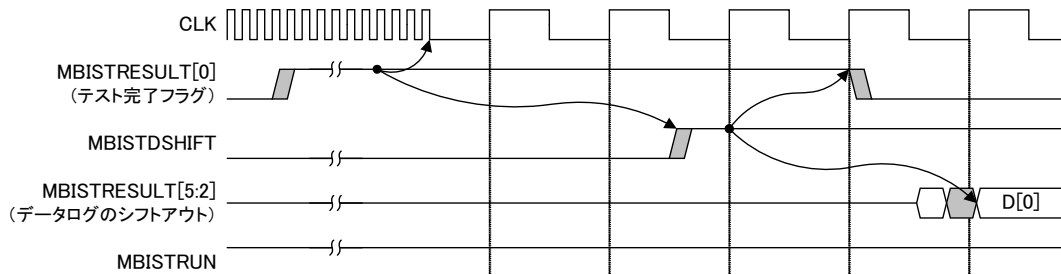


図 2-25 ビットマップ データログの取得開始

シフト処理が完了し、**MBISTDSHIFT** を **LOW** に駆動すると、図 2-26 に示すように、コントローラは停止した場所からテストを再開します。この手順は、テストのアルゴリズムが完了するまで続行されます。テストアルゴリズムの実行中に、1つのフォールトから数回のエラーが発生することもあります。アルゴリズムによって実行される読み出しの回数と、フォールトの正確な性質によっては、フォールトが複数回ログに記録されることがあります。

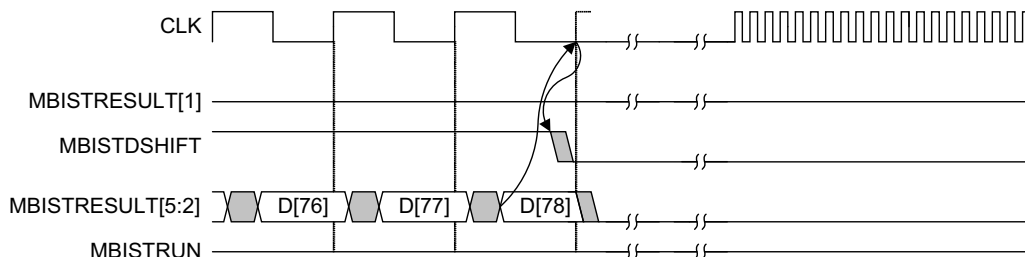


図 2-26 ビットマップ データログの取得終了

新しい命令をロードすると、ビットマップモードはリセットされます。

## 第 3 章

# MBIST 命令レジスタ

本章では、*MBIST* 命令レジスタ(MBIR)を使用してMBISTコントローラの動作モードを構成する方法について説明します。本章は次のセクションから構成されています。

- 「*MBIST* 命令レジスタについて」 (ページ 3-2)
- 「フィールドの説明」 (ページ 3-4)

### 3.1 MBIST 命令レジスタについて

MBIST は、*MBIST 命令レジスタ (MBIR)* にストアされているロード済みの命令を実行します。MBIR は 58 ビット幅で、18 ビットの制御ユニット部と、40 ビットのディスパッチユニット部に分けられます。

MBIR は、**MBISTSHIFT** がアサートされたとき、制御ユニットのシリアルポート **MBISTDATAIN** 経由でロードされます。**MBISTSHIFT** が再度アサートされると、制御ユニットは **MBISTDATAIN** を直列に、**MBISTTX[3]** ポート経由でディスパッチユニットに渡します。

MBIR の制御ユニット部を、図 3-1 に示します。

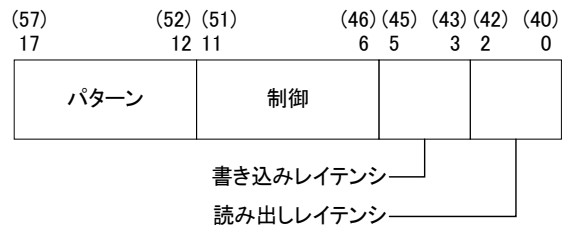


図 3-1 MBIST 命令レジスタの制御ユニット

制御ユニットには、次のフィールドが含まれています。

- パターン**                      テストアルゴリズムを指定します。
- 制御**                              動作の MBIST モードと、エラーフラグがスティッキーかどうかを指定します。
- 書き込みレイテンシ**              RAM 書き込みを可能にするサイクル数を指定します。
- 読み出しレイテンシ**              RAM 読み出しを可能にするサイクル数を指定します。

MBIR のディスパッチユニット部を、図 3-2 に示します。

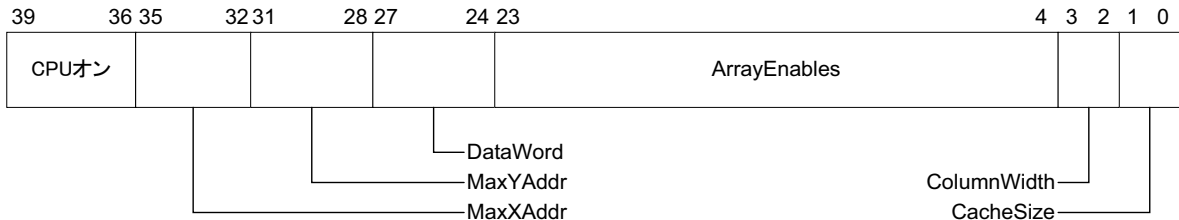


図 3-2 MBIST 命令レジスタのディスパッチユニット

ディスパッチユニットには、次のフィールドが含まれています。

<b>CPU オン</b>	テスト対象の CPU でのデータ比較を制御します。
<b>MaxXAddr</b>	X アドレスカウンタのビット数を指定します。
<b>MaxYAddr</b>	Y アドレスカウンタのビット数を指定します。
<b>DataWord</b>	テスト中に使用されるデータワード。この 4 ビットが 16 回複製され、64 ビットのデータを形成します。
<b>ArrayEnables</b>	テスト対象の RAM を指定します。
<b>ColumnWidth</b>	RAM ブロックごとの列の数を、4、8、16、32 のいずれかに指定します。
<b>CacheSize</b>	キャッシュサイズを、16KB、32KB、64KB のいずれかに指定します。

MBIR フィールドのより詳細な説明については、「フィールドの説明」(ページ 3-4) を参照して下さい。

## 3.2 フィールドの説明

次に示すサブセクションでは、MBIR のフィールドについて説明します。

- 「パターンフィールド、MBIR[57:52]」
- 「制御フィールド、MBIR[51:46]」 (ページ 3-7)
- 「読み出しレイテンシフィールドと書き込みレイテンシフィールド、MBIR[42:40] およびMBIR[45:43]」 (ページ 3-7)
- 「CPU オンフィールド、MBIR[39:36]」 (ページ 3-8)
- 「MaxXAddr フィールドと MaxYAddr フィールド、MBIR[35:32] およびMBIR[31:28]」 (ページ 3-9)
- 「DataWord フィールド、MBIR[27:24]」 (ページ 3-11)
- 「ArrayEnables フィールド、MBIR[23:4]」 (ページ 3-11)
- 「ColumnWidth フィールド、MBIR[3:2]」 (ページ 3-12)
- 「CacheSize フィールド、MBIR[1:0]」 (ページ 3-13)

### 3.2.1 パターンフィールド、MBIR[57:52]

MBIST コントローラには、業界標準のパターンアルゴリズムと、ビットレートのストレスアルゴリズムが組み込まれています。アルゴリズムをグループ化することで、製品に固有のメモリテスト方式を作成できます。

サポートされているアルゴリズムの説明を表 3-1 に、それらの使用法を「パターン仕様」(ページ 3-5) に示します。表の N の値は、アドレス位置ごとの RAM アクセス数を示し、そのアルゴリズムを使用する際のテスト時間の推定に使用できます。

表 3-1 パターンフィールドのエンコード

パターン MBIR[57:52]	アルゴリズム名	N	説明
b000000	ソリッド書き込み	1N	ソリッドパターンをメモリに書き込みます。
b000001	ソリッド読み出し	1N	ソリッドパターンをメモリから読み出します。
b000010	チェッカーボード書き込み	1N	チェッカーボードパターンをメモリに書き込みます。
b000011	チェッカーボード読み出し	1N	チェッカーボードパターンをメモリから読み出します。
b000100	March C+ (x-fast)	14N	March C+ アルゴリズムで、X アドレスを最初にインクリメントします。
b001011	March C+ (y-fast)	14N	March C+ アルゴリズムで、Y アドレスを最初にインクリメントします。
b000101	エラーパターン	6N	メモリのエラー検出機能をテストします。
b000110	読み出し - 書き込み March (x-fast)	6N	読み出し - 書き込み March パターンで、X アドレスを最初にインクリメントします。



表 3-1 パターンフィールドのエンコード (続き)

パターン MBIR[57:52]	アルゴリズム名	N	説明
b000111	読み出し - 書き込み March (y-fast)	6N	読み出し - 書き込み March パターンで、Y アドレスを最初にインクリメントします。
b001000	読み出し - 書き込み - 読み出し March (x-fast)	8N	読み出し - 書き込み - 読み出し March パターンで、X アドレスを最初にインクリメントします。
b001001	読み出し - 書き込み - 読み出し March (y-fast)	8N	読み出し - 書き込み - 読み出し March パターンで、Y アドレスを最初にインクリメントします。
b001010	Bang	18N	ビットライン ストレスパターン
b111111	Go/No-Go	30N	表 3-2 (ページ 3-6) を参照

### パターン仕様

ここでは、MBIST のテストパターンについて説明します。x-fast パターンでは、X アドレスカウンタのインクリメントまたはデクリメントが先に行われます。y-fast パターンでは、Y アドレスカウンタのインクリメントまたはデクリメントが先に行われます。X アドレスと Y アドレスのカウンタについては、「MaxXAddr フィールドと MaxYAddr フィールド、MBIR[35:32] および MBIR[31:28]」(ページ 3-9) を参照して下さい。

最初の 4 つのパターンは、データの保持または I<sub>DDQ</sub> テストに便利です。

#### ソリッド書き込み

指定されたデータシードで RAM を初期化します。

#### ソリッド読み出し

それぞれの RAM 位置を 1 回ずつ読み出し、指定されたデータシードが読み出されることを確認します。

#### チェッカーボード書き込み

指定されたデータシードと、その逆数とを交互に並べて作成される、物理的なチェッカーボードパターンで、RAM を初期化します。

#### チェッカーボード読み出し

指定されたデータシードと、その逆数とを交互に並べて作成される、物理的なチェッカーボードパターンを、RAM から読み出します。

以下の一連のパターンでは、アルゴリズムを記述するために次の表記が使用されます。

- 0 データシードを表します。
- 1 データシードの逆数を表します。

- r** 読み出し操作を表します。
- w** 書き込み操作を表します。
- incr** アドレスを 0 から開始し、addrmax になるまでインクリメントします。
- decr** アドレスを addrmax から開始し、0 になるまでデクリメントします。

**March C+ (x-fast または y-fast)**

業界標準の March C+ アルゴリズムです。

```
(w0) (r0, w1, r1) (r1, w0, r0) decr (r0, w1, r1) decr (r1, w0, r0)
(r0)
```

**読み出し - 書き込み March (x-fast または y-fast)**

```
(w0) (r0, w1) decr (r1, w0) (r0)
```

**読み出し - 書き込み - 読み出し March (x-fast または y-fast)**

```
(w0) (r0, w1, r1) decr (r1, w0, r0) (r0)
```

- Bang** このテストは、常に x-fast で実行されます。複数の連続した書き込みと読み出しを実行し、結果的にビットライン ペアにストレスを与えます。このパターンは特定場所でのフォールトを検出しますが、主な目的はメモリのアナログ特性を判定することです。以下のアルゴリズムの説明で、行 0 は犠牲行へのデータシードの読み出しまたは書き込みを示します。これは常に、アドレス指定される列の最初の行です。

```
(w0) (r0, w0, w0(row 0) × 6) (r0 × 5, w0(row 0), r0) (r0)
```

- Go/No-Go** 独自のメモリテスト手法を実装する必要がない場合、Go/No-Go テストパターンを使用します。このパターンは、表 3-2 に示すアルゴリズムを実行します。

**表 3-2 Go/No-Go テストパターン**

シーケンス	アルゴリズム	データ
1	チェッカーボード書き込み	データシード
2	チェッカーボード読み出し	データシード
3	チェッカーボード書き込み	データシード
4	チェッカーボード読み出し	データシード
5	読み出し - 書き込み - 読み出し March (y-fast)	0x6
6	Bang	0xF

このテストスイートは、アレイの包括的なテストを行います。Go/No-Go の一連のテストは、ARM のメモリテスト エンジニアによるメモリテストの経験を元に作成されたものです。

### 3.2.2 制御フィールド、MBIR[51:46]

この6ビット制御フィールドは、MBISTの機能を指定します。制御フィールドの下位5ビットと、MBISTコントローラの動作との関係を、表3-3に示します。

表 3-3 制御フィールドのエンコード（下位5ビット）

制御 MBIR[50:46]	動作	説明
bx00000	デフォルト	テストを完了まで実行します。MBIR[51]が0の場合、最初の障害の後でスティッキーエラーが存在します。
bx00001	エラー時停止	エラーの発生時点でテストを終了します。
bx00011	ビットマップモード	各障害をログに記録できるようにします。「ビットマップモード」(ページ2-19)を参照。

MBIR[51]は、MBISTRESULT[1]エラーフラグがスティッキーかどうかを選択します。

- MBIR[51]がセットされている場合、エラービットはリアルタイムでトグルされます。エラーが発生した比較についてはHIGH、成功した比較についてはLOWになります。

#### 注

MBIR[51]をセットすると、エラービットはテスト周波数でトグルされます。外部ピンまたはATEがテスト周波数に追従できない場合、この方法は非推奨です。

- MBIR[51]がクリアされているとき、エラービットはスティッキーです。最初のエラーが発生してから、新しいMBIST命令がシフトインされるか、データログがシフトアウトされるまで、HIGHに維持されます。

### 3.2.3 読み出しレイテンシフィールドと書き込みレイテンシフィールド、MBIR[42:40] および MBIR[45:43]

MBIRの読み出しレイテンシフィールドと書き込みレイテンシフィールドは、テスト対象のRAMの読み出し/書き込みレイテンシを指定するために使用されます。読み出し/書き込みレイテンシは、RAMが読み出し/書き込み操作を完了するために必要なサイクル数です。例えば、RAMへの書き込みレイテンシが2サイクルの場合、RAM入力が有効なのは1サイクルの間のみです。次のサイクルはNOPサイクルで、チップイネーブルは否定状態です。同様に、RAMの読み出しレイテンシが3サイクルの場合、RAM入力が有効なのは1サイクルの間のみです。2つのNOPサイクルの後で、読み出しデータがRAM出力で有効になります。

---

**注**


---

テスト対象の RAM で、読み出しと書き込みの両方の操作についてレイテンシが同じであっても、MBIR の読み出しレイテンシと書き込みレイテンシのフィールドは両方プログラムし、それぞれに同じ値を指定する必要があります。

---

読み出し操作のレイテンシ設定を、表 3-4 に示します。

**表 3-4 読み出しレイテンシフィールドのエンコード**

読み出しレイテンシ、MBIR[42:40]	読み出し操作あたりのサイクル数
b000	1
b001	2
b010	3
b011	4
b100	5
b101	6
b110	7
b111	8

書き込み操作のレイテンシ設定を、表 3-5 に示します。

**表 3-5 書き込みレイテンシフィールドのエンコード**

書き込みレイテンシ、MBIR[45:43]	書き込み操作あたりのサイクル数
b000	1
b001	2
b010	3
b011	4
b100	5
b101	6
b110	7
b111	8

### 3.2.4 CPU オンフィールド、MBIR[39:36]

CPU オンフィールドは、テスト対象の CPU でのデータ比較を制御します。

**MBIR[39:36]** ビットのいずれかを 0 にセットすると、対応する CPU でのデータ比較が不可能になります。このフィールドの CPU マッピングを、表 3-6 に示します。

表 3-6 MBIR[39:36] の CPU へのマッピング

MBIR ビット	CPU
39	CPU3
38	CPU2
37	CPU1
36	CPU0

### 3.2.5 MaxXAddr フィールドと MaxYAddr フィールド、MBIR[35:32] および MBIR[31:28]

RAM について指定する必要があるアドレスビットの数は、MBIR フィールドから特定できます。

- 『MaxXAddr』 (ページ 3-10)
- 『MaxYAddr』 (ページ 3-10)

これによって、アドレス範囲を二次元的に指定できます。二次元的な指定は、RAM の物理的な実装のトポロジをより正確に表現しています。2つの次元は、X アドレスカウンタと Y アドレスカウンタという 2つの独立したアドレスカウンタによって制御されます。カウンタの一方は、もう一方のカウンタのすべての値が終了したときにのみインクリメントまたはデクリメントされます。選択したテストアルゴリズムにより、どちらのカウンタが先に変更されるかが決定されます。

X アドレスと Y アドレスのカウンタに割り振るビット数を特定するには、次の手順を使用します。

1. RAM アレイの列幅を判定します。Y アドレスは、最低でも選択した列と同じビット数にする必要があります。
2. RAM が必要とするアドレスビットの数を判定します。詳細については、『Cortex-A9 プロセッサ構成およびサインオフガイド』を参照して下さい。

**MaxXAddr**

MaxXAddr フィールドは、テスト中に使用する X アドレスカウンタのビット数を指定します。MaxXAddr の設定を、表 3-7 に示します。

表 3-7 MaxXAddr フィールドのエンコード

MaxXAddr、MBIR[35:32]	カウンタのビット数
<b0010	非サポート
b0010	2
b0011	3
b0100	4
b0101	5
b0110	6
b0111	7
b1000	8
b1001	9
b1010	10
>b1010	予約

**MaxYAddr**

MaxYAddr フィールドは、テスト中に使用する Y アドレスカウンタのビット数を指定します。MaxYAddr の設定を、表 3-8 に示します。

表 3-8 MaxYAddr フィールドのエンコード

MaxYAddr、MBIR[31:28]	カウンタのビット数
<b0010	非サポート
b0010	2
b0011	3
b0100	4
b0101	5
b0110	6
b0111	7
b1000	8

表 3-8 MaxYAddr フィールドのエンコード (続き)

MaxYAddr、MBIR[31:28]	カウンタのビット数
b1001	9
b1010	10
>b1010	予約

### 3.2.6 DataWord フィールド、MBIR[27:24]

DataWord は 4 ビットのデータシード フィールドで、命令ロード時にテストアルゴリズムのバックグラウンドデータを提供します。

#### 注

Go/No-Go アルゴリズム、読み出し - 書き込み - 読み出し March (y-fast)、および Bang アルゴリズムは、データシードの値を使用しません。Go/No-Go アルゴリズムで使用されるデータを、表 3-2 (ページ 3-6) に示します。

データシードにより、**I<sub>DDQ</sub>** ATPG のアレイに格納する値の選択、または March やビットライン ストレステストで予期しない感知度を検索するためのデータワードの選択を実行できます。MBIST エンジンには、この 4 ビットのデータを 16 回複製し、MBIST インタフェースの **MBISTDIN[63:0]** ポートに必要な 64 ビットのデータ全体を生成します。

### 3.2.7 ArrayEnables フィールド、MBIR[23:4]

ArrayEnables フィールドの各ビットによって、テスト対象のキャッシュ RAM アレイを選択する方法を、表 3-9 に示します。同時に選択できるアレイは 1 つだけです。複数のアレイを選択した場合の動作は予測不能です。

表 3-9 ArrayEnables フィールドのエンコード

ArrayEnables、MBIR[23:4]	RAM 名
b000000000000000000000001	BTAC RAM 制御アレイ 0 およびターゲットアレイ 0
b000000000000000000000010	BTAC RAM 制御アレイ 1 およびターゲットアレイ 1
b000000000000000000000100	命令キャッシュタグ RAM アレイ 0 および 1
b000000000000000000001000	命令キャッシュタグ RAM アレイ 2 および 3
b000000000000000000010000	命令キャッシュデータ RAM ウェイ 0 (ブロック 0 および 1)

表 3-9 ArrayEnables フィールドのエンコード (続き)

ArrayEnables、MBIR[23:4]	RAM 名
b00000000000000100000	命令キャッシュデータ RAM ウェイ 1 (ブロック 2 および 3)
b00000000000001000000	命令キャッシュデータ RAM ウェイ 2 (ブロック 4 および 5)
b00000000000010000000	命令キャッシュデータ RAM ウェイ 3 (ブロック 6 および 7)
b00000000001000000000	グローバル履歴バッファ
b00000000001000000000	TLB RAM アレイ 0
b00000000010000000000	TLB RAM アレイ 1
b00000000100000000000	データキャッシュ タグ RAM アレイ 0 および 1
b00000001000000000000	データキャッシュ タグ RAM アレイ 2 および 3
b00000010000000000000	データキャッシュ データ RAM ウェイ 0 (ブロック 0 および 4)
b00000100000000000000	データキャッシュ データ RAM ウェイ 1 (ブロック 1 および 5)
b00001000000000000000	データキャッシュ データ RAM ウェイ 2 (ブロック 2 および 6)
b00010000000000000000	データキャッシュ データ RAM ウェイ 3 (ブロック 3 および 7)
b00100000000000000000	データ RAM
b01000000000000000000	SCU タグ RAM アレイ 0 および 1
b10000000000000000000	SCU タグ RAM アレイ 2 および 3

### 3.2.8 ColumnWidth フィールド、MBIR[3:2]

ColumnWidth フィールドは、テスト対象のアレイの各 RAM ブロックに含まれる列の数を指定します。列アドレスは常に RAM アドレスの最下位ビットとしてエンコードされるため、列の数により使用されるビット数が決定されます。この情報は、特定の MBIST 操作、例えばビットライン ストレステストや、アレイに実際に物理的なチェッカーボードパターンを書き込む場合、正しい動作のため重要です。



サポートされている列の幅と、それぞれの幅について使用されるアドレス下位ビット数、およびそれらを選択するために必要な MBIR エンコードを、表 3-10 に示します。

表 3-10 ColumnWidth フィールドのエンコード

ColumnWidth、MBIR[3:2]	列数	アドレスのビット数
b00	4	2
b01	8	3
b10	16	4
b11	32	5

### 3.2.9 CacheSize フィールド、MBIR[1:0]

CacheSize フィールドは、モジュールの実装に含まれるキャッシュのサイズを指定します。サポートされているキャッシュサイズを、表 3-11 に示します。

表 3-11 CacheSize フィールドのエンコード

CacheSize、MBIR[1:0]	キャッシュサイズ
b00	16KB
b10	32KB
b11	64KB
b01	予約 <sup>a</sup>

a. 内部的に 16KB にマップされます。



## 第 4 章

# MBIST データログレジスタ

本章では、MBIST データログレジスタについて説明します。本章は次のセクションから構成されています。

- 「MBIST データログレジスタについて」 (ページ 4-2)
- 「フィールドの説明」 (ページ 4-3)

## 4.1 MBIST データログレジスタについて

MBIST データログレジスタは、エラーの発生したアレイについての情報を記録します。パリティ付きの構成について、レジスタの形式を図 4-1 に示します。

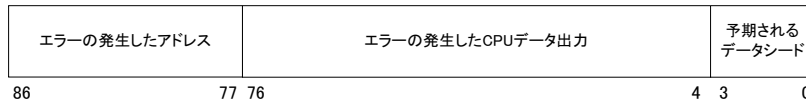


図 4-1 MBIST データログレジスタの形式、パリティ付きの構成

パリティなしの構成について、レジスタの形式を図 4-2 に示します。

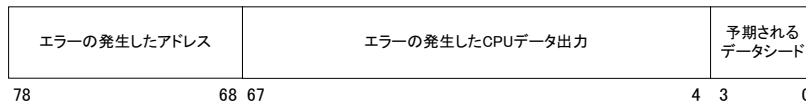


図 4-2 MBIST データログレジスタの形式、パリティなしの構成

レジスタのフィールドの詳細については、「フィールドの説明」(ページ 4-3)を参照して下さい。

すべての CPU のデータログは、**MBISTRESULT[5:2]** 経由で並列にダンプされます。

- **MBISTRESULT[5]** は CPU3 用に使用されます。
- **MBISTRESULT[4]** は CPU2 用に使用されます。
- **MBISTRESULT[3]** は CPU1 用に使用されます。
- **MBISTRESULT[2]** は CPU0 用に使用されます。

## 4.2 フィールドの説明

これらのフィールドは、パリティ付きの構成の MBIST データログレジスタに存在します。

**Datalog[87:77]** エラーの発生したアドレスを示す 11 ビットフィールド。

**Datalog[76:4]** エラーとなったデータと正しいデータとの XOR を含む、64 ビットフィールド。1'b1 のビットは、すべてエラーを示しています。

**Datalog[3:0]** 予期されるデータシードを含む 4 ビットフィールド。

これらのフィールドは、パリティなしの構成の MBIST データログレジスタに存在します。

**Datalog[78:68]** エラーの発生したアドレスを示す 11 ビットフィールド。

**Datalog[67:4]** エラーとなったデータと正しいデータとの XOR を含む、64 ビットフィールド。1'b1 のビットは、すべてエラーを示しています。

**Datalog[3:0]** 予期されるデータシードを含む 4 ビットフィールド。



# 付録 A

## 信号の説明

この付録では、MBIST コントローラの信号について説明します。本章は次のセクションから構成されています。

- 「MBIST コントローラのインタフェース信号」 (ページ A-2)
- 「その他の信号」 (ページ A-4)

## A.1 MBIST コントローラのインタフェース信号

パリティなしの構成での MBIST コントローラのインタフェース信号を、表 A-1 に示します。

**表 A-1 パリティなし構成での MBIST コントローラのインタフェース信号**

信号	タイプ	説明
MBISTOUTDATA[255:0]	入力	Cortex-A9 プロセッサからの MBIST データ出力
MBISTADDR[10:0]	出力	MBIST アドレス
MBISTARRAY[19:0]	出力	MBIST RAM のチップイネーブル。いずれか 1 つのビットのみがアクティブになります。表 A-3 (ページ A-3) を参照して下さい。
MBISTINDATA[63:0]	出力	Cortex-A9 プロセッサへの MBIST データ入力
MBISTBE[25:0]	出力	MBIST 書き込みイネーブル
MBISTWRITEEN	出力	グローバル書き込みイネーブル

パリティ付きの構成での MBIST コントローラのインタフェース信号を、表 A-2 に示します。

**表 A-2 パリティ付き構成での MBIST コントローラのインタフェース信号**

信号	タイプ	説明
MBISTOUTDATA[278:0]	入力	Cortex-A9 プロセッサからの MBIST データ出力
MBISTADDR[10:0]	出力	MBIST アドレス
MBISTARRAY[19:0]	出力	MBIST RAM のチップイネーブル。いずれか 1 つのビットのみがアクティブになります。表 A-3 (ページ A-3) を参照して下さい。
MBISTINDATA[71:0]	出力	Cortex-A9 プロセッサへの MBIST データ入力
MBISTBE[32:0]	出力	MBIST 書き込みイネーブル
MBISTWRITEEN	出力	グローバル書き込みイネーブル



表 A-3 MBISTARRAY のチップイネーブル (1 ビットのみアクティブ)

MBISTARRAY ビット	RAM 名
0	BTAC RAM 制御アレイ 0 およびターゲットアレイ 0
1	BTAC RAM 制御アレイ 1 およびターゲットアレイ 1
2	命令キャッシュタグ RAM アレイ 0 および 1
3	命令キャッシュタグ RAM アレイ 2 および 3
4	命令キャッシュデータ RAM ウェイ 0 (ブロック 0 および 1)
5	命令キャッシュデータ RAM ウェイ 1 (ブロック 2 および 3)
6	命令キャッシュデータ RAM ウェイ 2 (ブロック 4 および 5)
7	命令キャッシュデータ RAM ウェイ 3 (ブロック 6 および 7)
8	グローバル履歴バッファ
9	TLB RAM アレイ 0
10	TLB RAM アレイ 1
11	データキャッシュタグ RAM アレイ 0 および 1
12	データキャッシュタグ RAM アレイ 2 および 3
13	データキャッシュデータ RAM ウェイ 0 (ブロック 0 および 4)
14	データキャッシュデータ RAM ウェイ 1 (ブロック 1 および 5)
15	データキャッシュデータ RAM ウェイ 2 (ブロック 2 および 6)
16	データキャッシュデータ RAM ウェイ 3 (ブロック 3 および 7)
17	データ RAM
18	SCU タグ RAM アレイ 0 および 1
19	SCU タグ RAM アレイ 2 および 3

## A.2 その他の信号

その他の信号を、表 A-4 に示します。

表 A-4 その他の信号

信号	タイプ	説明
nRESET	入力	グローバルのアクティブ LOW リセット信号
CLK	入力	クロック
MBISTDATAIN	入力	シリアルデータ入力
MBISTDSHIFT	入力	データログ シフト
MBISTRESETN	入力	MBIST リセット
MBISTRESULT[5:0]	出力	出力ステータスバス
MBISTRUN	入力	MBIST テストの実行
MBISTSHIFT	入力	命令シフト
MBISTENABLE	入力	MBIST モードイネーブル

# 付録 B

## リビジョン

この付録では、本書の各版の技術的な相違点について説明します。

**表 B-1 A 版と B 版の相違点**

変更内容	場所
技術的変更なし	-

**表 B-2 B 版と C 版の相違点**

変更内容	場所
MBIST コントローラについての記述の更新	「MBIST コントローラについて」 (ページ 1-2)
MBIST コントローラの信号の更新	図 1-2 (ページ 1-3)
MBIST コントローラのインタフェース信号の更新	図 1-4 (ページ 1-4)
信号名と設定の更新	表 2-2 (ページ 2-4)
MBIST コントローラインタフェースのビット情報の更新	表 2-3 (ページ 2-4)
命令キャッシュタグ RAM のデータ入力の明確化	図 2-3 (ページ 2-8)
命令キャッシュタグ RAM のデータ出力の明確化	図 2-4 (ページ 2-8)

表 B-2 B 版と C 版の相違点 (続き)

変更内容	場所
タグ RAM 制御の明確化	表 2-6 (ページ 2-10)
TLB RAM の説明の更新	「TLB RAM」 (ページ 2-11)
分岐先アドレスキャッシュ RAM の説明の更新	「分岐先アドレスキャッシュ RAM」 (ページ 2-10)

表 B-3 C 版と D 版の相違点

変更内容	場所
パリティ付きの MBIST コントローラ図の追加	図 1-3 (ページ 1-4)
パリティ付き構成の MBIST 信号の追加	表 1-2 (ページ 1-6)
パリティ付き構成での MBISTINDATA および MBISTOUTDATA のサイズについての情報の追加	「MBISTINDATA と MBISTOUTDATA のマッピング」 (ページ 2-6)
パターンの説明から矢印を削除	「パターン仕様」 (ページ 3-5)
データログのエラーデータ出力サイズを 72 ビットに変更	図 4-1 (ページ 4-2) 「フィールドの説明」 (ページ 4-3)
パリティ付きとパリティなしの両方の構成について、MBIST データログレジスタの形式を追加	「MBIST データログレジスタについて」 (ページ 4-2)
パリティ付きとパリティなしの両方の構成について、フィールドの説明を追加	「フィールドの説明」 (ページ 4-3)
パリティ付きの MBIST コントローラインタフェース信号を追加	表 A-2 (ページ A-2)

表 B-4 D 版と E 版の相違点

変更内容	場所
技術的変更なし	-

表 B-5 D 版と F 版の相違点

変更内容	場所
参考資料に Cortex-A9 マルチプロセッサ TRM を追加	「ARM の刊行物」 (ページ xv)
テクニカルリファレンス マニュアルのリセットシーケンスへの参照を追加	「MBIST コントローラについて」 (ページ 1-2)
LOW に固定するピンの一覧の追加	「MBIST コントローラインタフェース」 (ページ 2-2)
ビットフィールドの説明とタイトルを一致	「MBISTINDATA と MBISTOUTDATA のマッピング」 (ページ 2-6)
図とテキストの修正	「命令キャッシュデータ RAM およびデータキャッシュ データ RAM」 (ページ 2-6)
	「命令タグ、データタグ、SCU タグの RAM」 (ページ 2-7)
	「外部 RAM」 (ページ 2-10)
	「分岐先アドレスキャッシュ RAM」 (ページ 2-10)
	「TLB RAM」 (ページ 2-11)
「グローバル履歴バッファ RAM」 (ページ 2-12)	
プロセッサのタイミングについての相互参照を追加	「タイミング」 (ページ 2-16)
図の修正	図 2-20 (ページ 2-17)
	図 2-26 (ページ 2-20)
ビットフィールドとタイトルの一致	「CPU オンフィールド、MBIR[39:36]」 (ページ 3-8)
MBIR の値の訂正	表 3-6 (ページ 3-9)



# 用語集

この用語集では、ARM Limited の発行する技術文書で使用されている用語について説明します。

## TLB

変換ルックアサイドバッファ *参照*。

## キャッシュ

Cortex-A9 プロセッサとメインメモリとの間に配置され、使用頻度の高い命令やデータのコピーを格納および取得するために使用される、オンチップまたはオフチップの高速アクセスメモリ位置のブロック。これによって、メモリアクセスの平均速度が大幅に向上するため、Cortex-A9 プロセッサのパフォーマンスも向上します。

この用語集の最後のページにあるキャッシュ用語の図も *参照*。

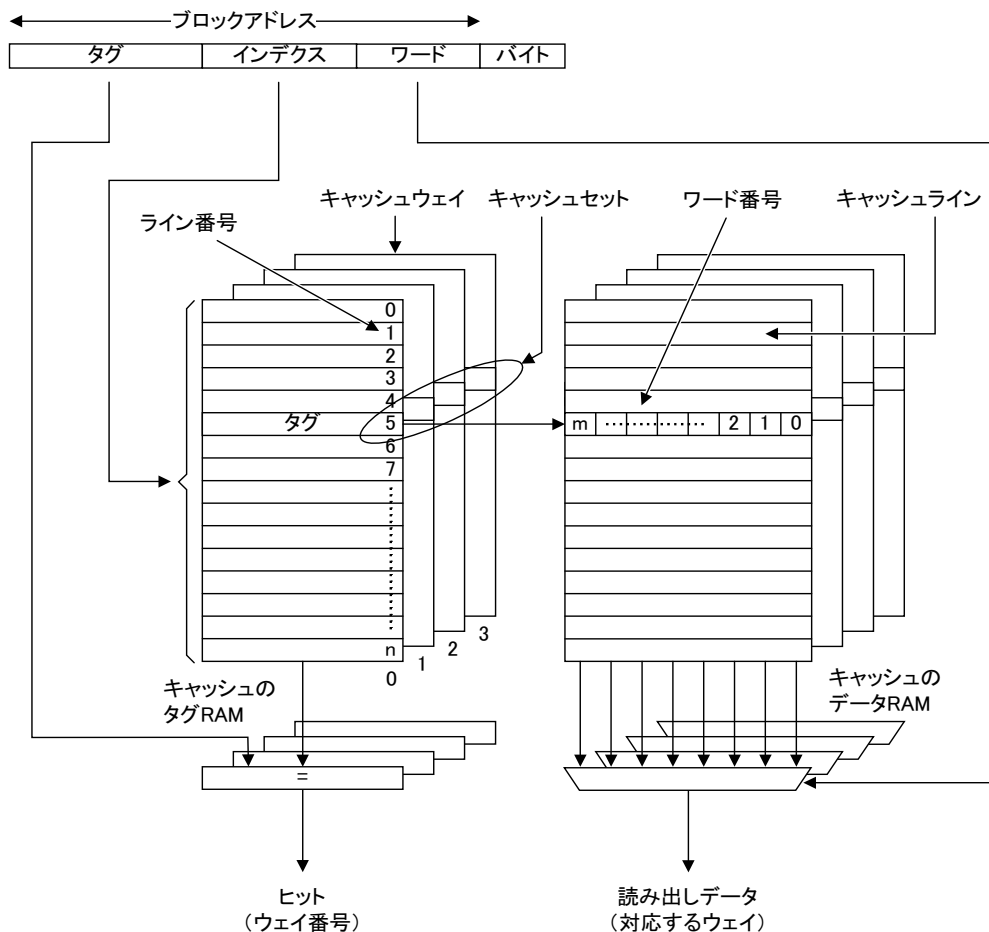
## キャッシュミス

命令または命令がアクセスするデータがキャッシュに存在しないために、メインメモリへのアクセスが必要になり、高速で処理できないメモリアクセス。

## キャッシュ用語の図

この図は、次のキャッシュ用語を示したものです。

- ブロックアドレス
- キャッシュライン
- キャッシュセット
- キャッシュウェイ
- インデクス
- タグ



### キャッシュライン

キャッシュ内の記憶域の基本単位。キャッシュラインは、サイズが常に2のべき乗（通常は4または8ワード）で、適切なメモリ境界にアラインしている必要があります。

この用語集の最後のページにあるキャッシュ用語の図も参照。

### コア

Cortex-A9 プロセッサの一部で、ALU、データパス、汎用レジスタ、プログラムカウンタ、命令デコードおよび制御回路が含まれます。

### タグ

キャッシュ内のキャッシュラインの識別に使用されるブロックアドレスの上位部分。CPUからのブロックアドレスは、セット内の各タグと並列に比較され、対応するラインがキャッシュに存在するかどうか判断されます。存在する場合はキャッシュヒットとなり、そのラインをキャッシュからフェッチ



できます。ブロックアドレスがどのタグにも対応しない場合はキャッシュミスとなり、そのラインは次のレベルのメモリからフェッチされる必要があります。

この用語集の最後のページにあるキャッシュ用語の図も参照。

**データキャッシュ**

Cortex-A9 プロセッサとメインメモリとの間に配置され、使用頻度の高いデータのコピーを格納および取得するために使用される、オンチップの高速アクセスメモリ位置のブロック。これによって、メモリアクセスの平均速度が大幅に向上するため、Cortex-A9 プロセッサのパフォーマンスも向上します。

**ハーフワード**

16 ビットのデータ項目。

**バイト**

8 ビットのデータ項目。

**ブロックアドレス**

タグ、インデクス、ワードフィールドで構成されるアドレス。タグビットによって、キャッシュヒット時に照合するキャッシュエントリを保持するウェイが識別されます。インデクスビットによって、アドレス指定されるセットが識別されます。ワードフィールドには、キャッシュエントリ内の特定のワード、ハーフワード、バイトの識別に使用可能なワードアドレスが格納されます。

この用語集の最後のページにあるキャッシュ用語の図も参照。

**変換ルックアサイドバッファ (TLB)**

最近使用されたページテーブルエントリのキャッシュで、メモリアクセスのたびにページテーブルウォークを行うオーバーヘッドを回避するために使用されます。メモリ管理ユニットの一部です。

**命令キャッシュ**

Cortex-A9 プロセッサとメインメモリとの間に配置され、使用頻度の高い命令のコピーを格納および取得するために使用される、オンチップの高速アクセスメモリ位置のブロック。これによって、メモリアクセスの平均速度が大幅に向上するため、Cortex-A9 プロセッサのパフォーマンスも向上します。

**ワード**

32 ビットのデータ項目。

