



XAPP709 (v1.4) 2005 年 11 月 18 日

Virtex-4 FPGA デバイスを使用した DDR SDRAM コントローラ

著者 : Olivier Despaux

概要

このアプリケーション ノートでは、Virtex™-4 XC4VLX25 FF668 -10CES デバイスを使用してインプリメントした 200MHz DDR SDRAM (JEDEC DDR400、PC3200 規格) コントローラについて説明します。このインプリメンテーションでは、データの取り込みにダイレクトクロッキング、データライン遅延の調整に自動調整回路を使用します。

DDR SDRAM デバイスは、コストが低く、集積度が高いストレージリソースで、多数のメモリベンダーから入手可能です。このリファレンス デザインは、DDR400 SDRAM コンポーネントと DIMM を使用して開発されています。

DDR SDRAM について

DDR SDRAM の仕様は、米国電子工業会 (Electronic Industries Alliance : EIA) の一部である JEDEC のサイト (<http://www.jedec.org/>) から入手できます。DDR SDRAM の仕様は、JESD79C という名前で参照されています。

DDR SDRAM デバイスは、消費者向け製品、映像システムなどさまざまなアプリケーションで最も頻りに利用されているシリコンメモリリソースです。DDR SDRAM デバイスの周波数範囲は、200MHz または DDR400 までです。DRAM デバイスは、コンポーネントまたはモジュールコンフィギュレーションで使用できます。

DDR コントローラのコマンド

表 1 に、コントローラにより発行されるコマンドを示します。これらのコマンドは、次の制御信号を使用してメモリに転送されます。

- 行アドレスセレクト ($\overline{\text{RAS}}$)
- 列アドレスセレクト ($\overline{\text{CAS}}$)
- ライトイネーブル ($\overline{\text{WE}}$)
- クロックイネーブル (CKE) (デバイスのコンフィギュレーション後は High に保持)
- チップセレクト ($\overline{\text{CS}}$) (デバイスの動作中は Low に保持)

表 1 : DDR SDRAM のコマンド

信号番号	機能	$\overline{\text{RAS}}$	$\overline{\text{CAS}}$	$\overline{\text{WE}}$
1	モードレジスタの読み込み	L	L	L
2	オートリフレッシュ	L	L	H
3	プリチャージ ⁽¹⁾	L	H	L
4	バンクを選択し、行をアクティブにする	L	H	H
5	書き込みコマンド	H	L	L
6	読み出しコマンド	H	L	H

© 2004-2005 Xilinx, Inc. All Rights Reserved. XILINX、Xilinx ロゴ、およびその他本文に含まれる商標名は Xilinx の商標です。本文書に記載されている「Xilinx」、ザイリックスのロゴ、およびザイリックスが所有する製品名等は、米国 Xilinx Inc. の米国における登録商標です。その他に記載されている会社名および製品名等は、各社の商標または登録商標です。保証否認の通知 : Xilinx ではデザイン、コード、その他の情報を「現状有姿の状態」で提供しています。この特徴、アプリケーションまたは規格の一実施例としてデザイン、コード、その他の情報を提供しておりますが、Xilinx はこの実施例が権利侵害のクレームを全く受けないということを表明するものではありません。お客様がご自分で実装される場合には、必要な権利の許諾を受ける責任があります。Xilinx は、実装の妥当性に関するいかなる保証を行なうものではありません。この保証否認の対象となる保証には、権利侵害のクレームを受けないことの保証または表明、および市場性に対する適合性についての黙示的な保証も含まれます。

表 1 : DDR SDRAM のコマンド (続き)

信号番号	機能	RAS	CAS	WE
7	NOP (No Operation)	H	H	H

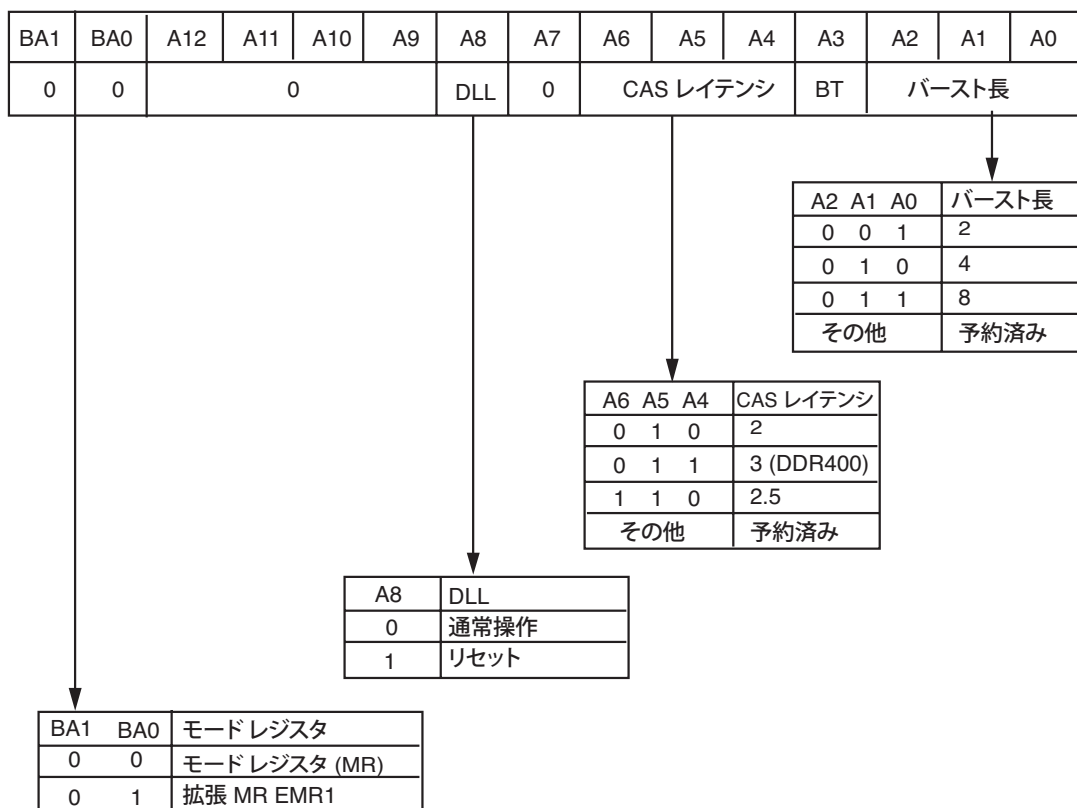
メモ :

1. アドレス信号 A10 は、PRECHARGE ALL BANKS のときは High に保持し、信号バンクプリチャージのときは Low に保持します。

コマンドの機能

モードレジスタ

モードレジスタは、バースト長の選択、バーストのタイプ、CAS レイテンシ、操作モードなど、DDR SDRAM の操作モードを定義します。図 1 に、このコントローラで使用するモードレジスタの定義を示します。



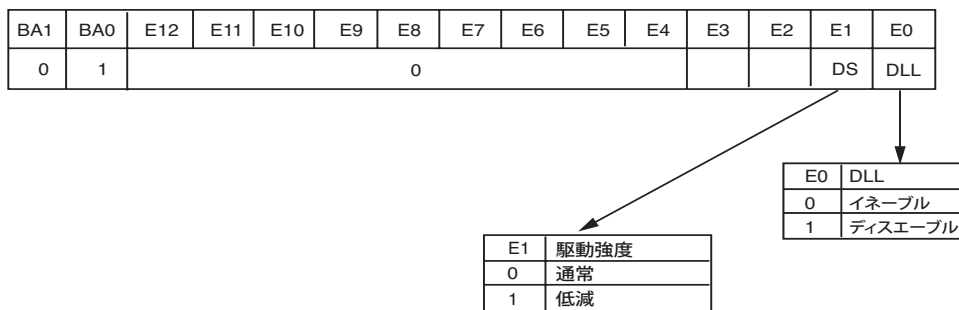
x709_01_090304

図 1 : DDR400 のモードレジスタの定義

バンクアドレス BA1 および BA0 は、モードレジスタを選択します。図 1 には、バンクアドレスビットのコンフィギュレーションも示されます。

拡張モードレジスタ

拡張モードレジスタは、DLL のイネーブル/ディスエーブル、DDR SDRAM インターフェイスの駆動強度など、モードレジスタでは制御できない機能を制御します。図 2 に、拡張モードレジスタの定義を示します。

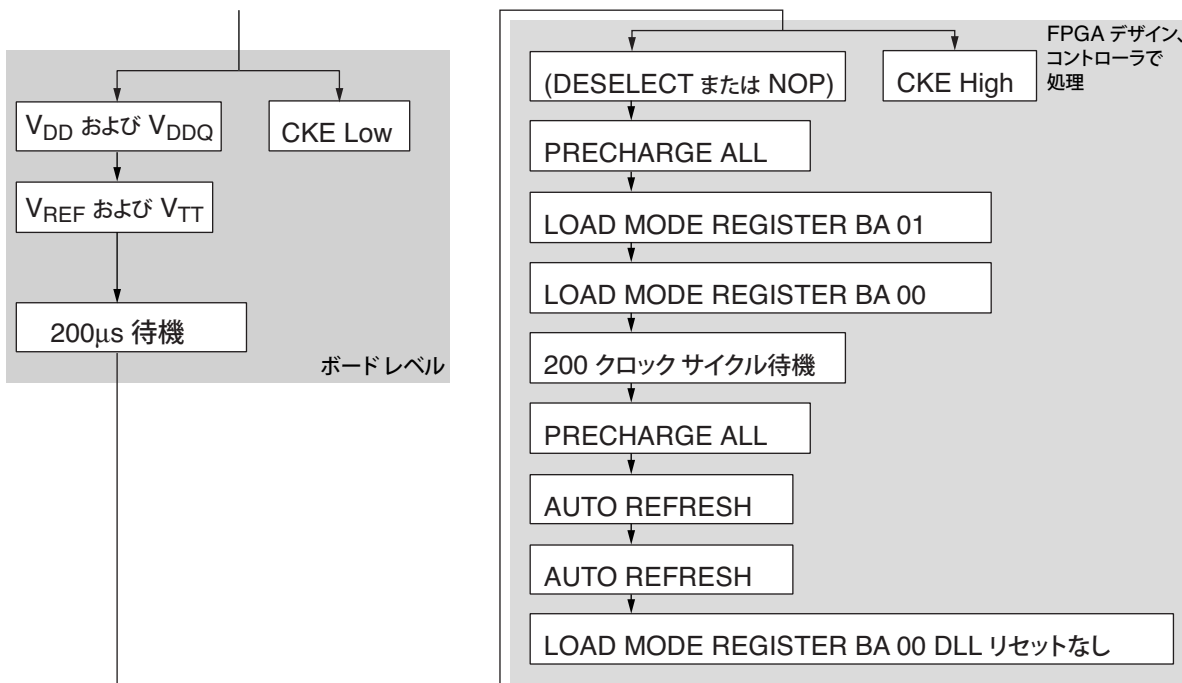


x709_02_032405

図 2 : DDR400 の拡張モード レジスタ

初期化シーケンス

コントローラ ステート マシンで使用される初期化シーケンスは、DDR SDRAM の仕様に従っています。コンフィギュレーションシーケンスには、2つの段階があります。1つ目の段階は電源投入時にハードウェアで処理され、2つ目の段階はFPGA メモリ コントローラ デザインで処理されます。図 3 に、初期化時に発行されるコマンドを示します。



x709_01_090204

図 3 : Virtex-4 を使用した DDR SDRAM の初期化シーケンス

初期化シーケンスが完了すると、コントローラにより DDR SDRAM メモリ デバイスに対してダミーの読み出しコマンドが発行されます。このプロセスにより、データパス モジュールで Virtex-4 入力遅延ブロックに対して適切なタップ数が選択されます。このダミー読み出しコマンド中にメモリにより発行されるデータ ストローブ (DQS) を使用して、入力 DQS と内部システム クロック (CLK0) 間のタイミング関係が決定されます。データパス モジュールに必要な遅延タップ数が決定されると、コントローラに Tap_select_done 信号が送信され、コントローラが IDLE 状態になります。

PRECHARGE コマンド

PRECHARGE コマンドは、特定のバンクの現在アクティブな行を非アクティブにします。PRECHARGE コマンドが発行された後、特定の時間 (t_{RP}) が経過すると、次の行をアクティブにできるようになります。入力 A10 は、1 つのバンクをプリチャージするか、すべてのバンクをプリチャージするかを指定します。

AUTO REFRESH コマンド

DDR デバイスは、7.8 μ s おきにリフレッシュする必要があります。コントローラには、カウンタのオートリフレッシュを要求する回路が組み込まれています。リフレッシュカウンタには、DCM の CLKDV 出力が使用されます。この出力は、オートリフレッシュカウンタに必要な低周波数のクロックを供給します。DCM の CLKDV で使用される BUFG を節約する場合は、DCM の高周波数の CLK0 出力または CLK/4 出力 (IDELAY 回路で使用) を使用することも可能です。オートリフレッシュ回路のクロックを変更する場合は、それに応じて mem_interface_top_parameters_0.v ファイルの max_ref_count を変更する必要があります。

auto_ref 信号は、AUTO REFRESH コマンドが必要であることを示し、AUTO REFRESH コマンドが発行されるまで High に保持されます。コントローラは、現在アクティブなバンクに対する処理を終了してから、AUTO REFRESH コマンドを発行します。

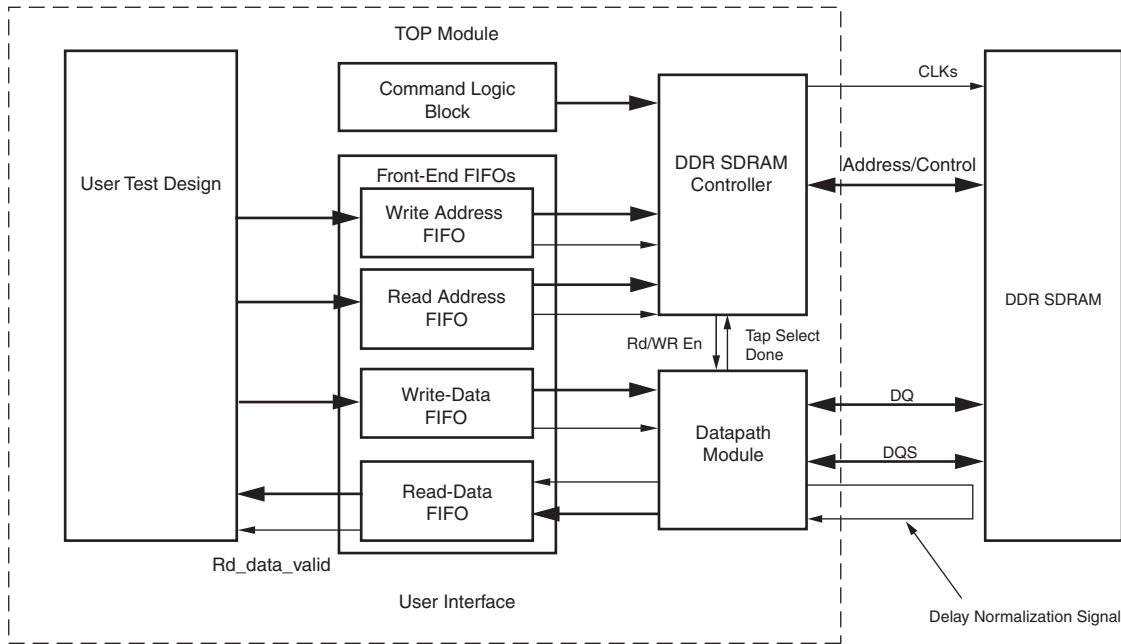
ACTIVE コマンド

ACTIVE コマンドは、メモリアレイ内のバンクに対して READ または WRITE コマンドを発行できるように、バンク内の行をアクティブにします。行がアクティブになると、 t_{RCD} 仕様に従って、その行に対して READ または WRITE コマンドを発行できるようになります。入力されたアドレスが現在アクティブな行と異なる行を指定している場合、コントローラはアドレス競合信号を送信し、PRECHARGE コマンドを発行して現在アクティブな行を非アクティブにし、ACTIVE コマンドを発行して新しい行をアクティブにします。

READ コマンド

READ コマンドは、アクティブな行に対してバースト読み出しアクセスを開始します。BA0 および BA1 はバンクアドレスを指定し、アドレス入力 A0 ~ Ai はアクティブ行の開始列の位置を指定します。読み出しバーストが終了した後、その行はプリチャージされるまでアクティブのままになります。

図 4 に、追加レイテンシが 0 の場合の READ コマンドを示します。この場合、読み出しレイテンシは CAS レイテンシ (DDR400 使用では 3) と同じになります。



x709_04_110805

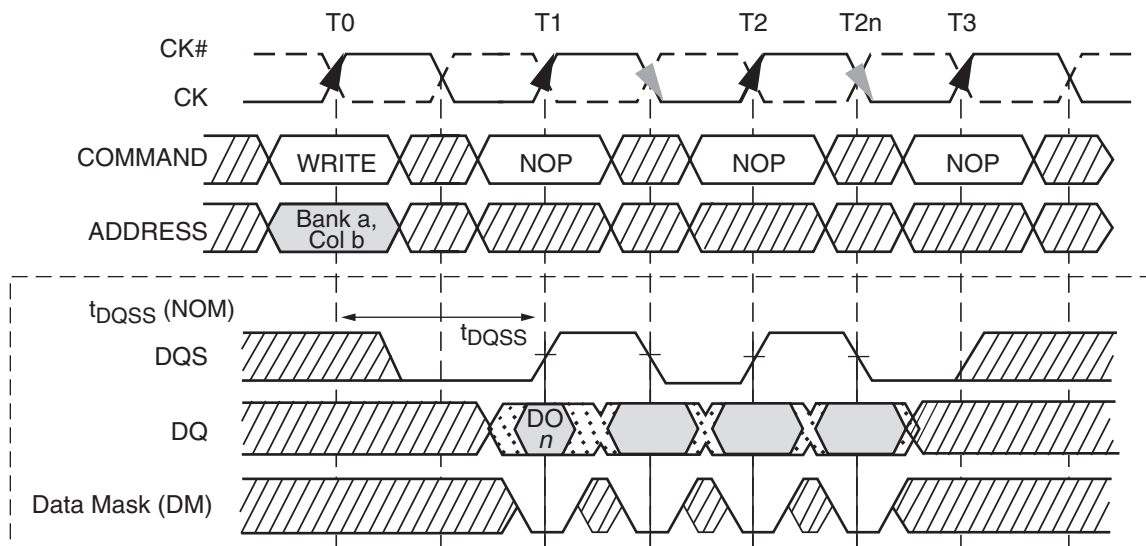
図 4： DDR SDRAM の読み出しアクセスの波形

WRITE コマンド

WRITE コマンドは、アクティブな行に対してバースト書き込みアクセスを開始します。BA0 および BA1 はバンクアドレスを指定し、アドレス入力 A0 ~ Ai はアクティブ行の開始列の位置を指定します。DDR SDRAM デバイスの書き込みレイテンシは、読み出しレイテンシから 1 クロック サイクル引いた値になります。

$$\text{書き込みレイテンシ} = \text{読み出しレイテンシ} - 1 = \text{CAS レイテンシ} - 1$$

図 5 に、書き込みレイテンシ 2 の場合の書き込みバーストの波形を示します。WRITE コマンドから DQS 信号の最初の立ち上がりエッジまでの時間は、書き込みレイテンシにより決まります。



X709_03_110805

図 5： DDR SDRAM の WRITE コマンドの波形

タイミング解析

Virtex-4 DDR400 リファレンス デザインでは、最高のパフォーマンスを達成し、タイミング マージンを最大にするため、デバイス特有の I/O およびクロック機能を使用しています。

リファレンス デザインのある物理レイヤでは、ダイレクト クロッキング手法を使用します。これについては、アプリケーション ノート XAPP701 『Memory Interfaces Data Capture Using Direct Clocking Technique』 (<http://www.xilinx.co.jp/bvdocs/appnotes/xapp701.pdf>) で説明しています。

このセクションでは、アドレス/制御パス、書き込み データパス、読み出し (キャプチャ) データパスのタイミング解析例を示します。

アドレス/制御パス

アドレス信号と制御信号は CLK180 に同期しており、CLK0 に対するメモリ デバイスのセットアップ タイムおよびホールド タイムが十分になるようにしています。

表 2 に、Virtex-4 にインプリメントされた、ダイレクト クロッキングを使用する DDR400 インターフェイスで、これらの信号をタイミング解析した例を示します。

表 2: アドレス信号と制御信号のタイミング解析

パラメータ	値 (ps)	立ち上がりエッジの不確定値	立ち下がりエッジの不確定値	説明
T _{CLOCK}	5000			クロック周期
T _{SETUP}	600	600	0	-
T _{HOLD}	600	-	600	-
T _{PACKAGE_SKEW}	±15	15	15	パッケージのスキュー
T _{JITTER}	±50	50	50	DCM の CLK0 および CLK180 出力を使用して DQ および DQS を生成
T _{CLOCK_TREE_SKEW}	±50	50	50	DQS と対応する DQ は近くに配置されているため、グローバルクロックラインのスキューは小さくなります。
T _{CLKOUT_PHASE}	±140	140	140	DCM 出力パラメータ値間の位相オフセットは、Virtex-4 データシート DS302 を参照
T _{PCB_LAYOUT_SKEW}	±400	400	400	ボード上のデータラインと対応するストロープ間のスキュー
不確定値の合計	2510	1255	1255	-
有効ウィンドウ	2490	1255	3745	-

図 6 に、アドレス信号と制御信号のタイミング マージンを示します。これらの信号は CLK180 を基準としているので、QDR_K クロックのエッジに対しては、立ち上がりエッジ マージンよりも立ち下がりエッジ マージンの方が長くなります。これにより、使用するグローバル クロック バッファの数が少なくなり、立ち上がりエッジのマージンも適切なものになります。

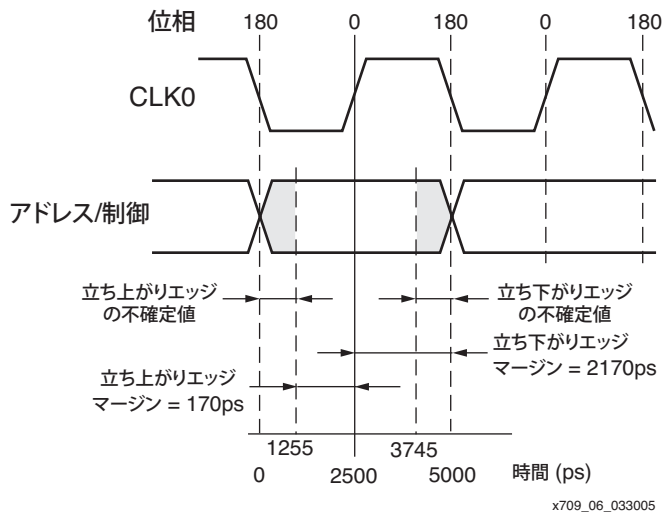


図 6： アドレス信号と制御信号のタイミング マージン

書き込みデータパス

書き込みデータパスは、CLK270 に同期していますが、書き込みデータは DDR 値として転送されるので、CLK0 の立ち上がりエッジおよび立ち下がりエッジの両方に対してセットアップ タイムおよびホールド タイムを適切なものにする必要があります。

表 3 に示す書き込みデータパスのタイミング解析には、メモリ クロックのデューティ サイクルの最大歪みが含まれています。この解析は、Virtex-4 デバイス (-10 スピード グレード) で行われたものです。

表 3： 書き込みデータパスのタイミング解析

パラメータ	値 (ps)	立ち上がりエッジの不確定値	立ち下がりエッジの不確定値	説明
T _{CLOCK}	5000	-	-	クロック周期
T _{DCD}	±250	250	250	メモリ クロックのデューティ サイクルの歪み (クロック周期の 5%)
T _{CLOCK_PHASE}	2250	-	-	データ周期は、クロック周期の半分から 10% のデューティ サイクルの歪みを減算した値
T _{SETUP}	400	400	0	メモリ データシートからのセットアップ タイム
T _{HOLD}	400	0	400	メモリ データシートからのホールド タイム
T _{PACKAGE_SKEW}	±15	15	15	パッケージのスキュー
T _{JITTER}	±50	50	50	DCM の CLK0 および CLK180 出力を使用して DQ および DQS を生成

表 3: 書き込みデータパスのタイミング解析 (続き)

パラメータ	値 (ps)	立ち上がりエッジの不確定値	立ち下がりエッジの不確定値	説明
T _{CLOCK_SKEW_FPGA}	±50	50	50	DQS と対応する DQ は近くに配置されているため、グローバルクロックラインのスキューは小さくなります。
T _{CLKOUT_PHASE}	±140	140	140	DCM 出力パラメータ値間の位相オフセットは、Virtex-4 データシートを参照
T _{PCB_LAYOUT_SKEW}	±20	20	20	ボード上のデータラインと対応するストロープ間のスキュー
不確定値の合計	1850	925	925	-
有効ウィンドウ	650	925	1575	データ有効ウィンドウ

図 7 に、書き込みデータパスのタイミング マージンを示します。CLK0 の立ち上がりエッジに対する解析のみを示します。CLK0 の立ち下がりエッジに対する解析はこれと同じです。

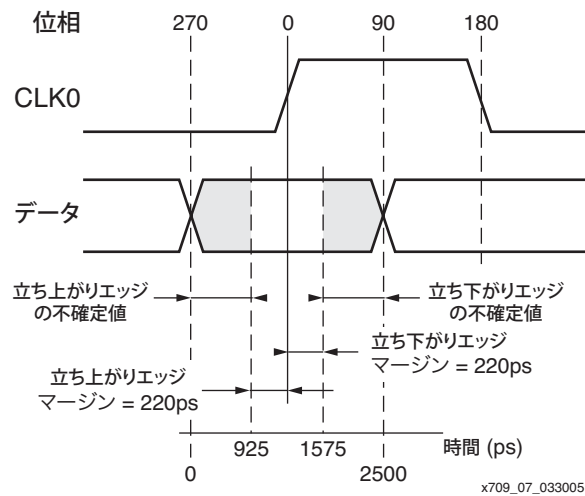


図 7: 書き込みデータパスのタイミング マージン

読み出しデータパス

読み出しデータパスの値は、前述のダイレクトクロッキング手法を使用して、FPGA クロックドメインに直接取り込まれます。そのため、データ取り込みのタイミング解析は CLK0 を基準にし、IDELAY タップ遅延の精度を考慮する必要があります。また、DQS ストロープとバス上のデータビットの間のスキューも考慮する必要があります。表 4 に、Virtex-4 デバイス (-10 スピードグレード) を使用した読み出しデータ取り込みのタイミング解析を示します。

表 4: 読み出しデータパスのタイミング解析

パラメータ	値 (ps)	立ち上がりエッジの不確定値	立ち下がりエッジの不確定値	説明
T _{CLOCK}	5000	-	-	クロック周期
T _{D_{CD}_MEMORY_DLL}	±150	150	150	Virtex-4 データシートに記述された、メモリに送信される信号上の DCM からのデューティサイクルの歪み
T _{DATA_PERIOD}	2350	-	-	データ周期は、クロック周期の半分から DCM のデューティサイクルの歪みを減算した値
T _{D_{QS}_TO_D_Q_SKEW}	400	400	0	メモリ デバイスのデータシートからの DQS から DQ へのスキューパラメータ
T _{HOLD_FACTOR}	500	0	500	メモリ ベンダーのデータシートからのホールド係数パラメータ
T _{PACKAGE_SKEW}	±15	15	15	パッケージのスキュー
T _{IDOCK}	340	340	0	Virtex-4 デバイス上のフリップフロップのセットアップタイム
T _{I_OCKD}	-150	0	-150	Virtex-4 デバイス上のフリップフロップのホールドタイム
T _{IDELAY_MEASURES}	±1 tap	75	75	評価エラー、DQS エッジごとに 1 タップの不確定さ (200MHz リファレンスクロックで 75ps)
T _{JITTER}	±100	100	100	DCM の CLK0 出力を使用する際に、間接的にストロープジッタの原因となるクロックジッタ
T _{CLOCK_SKEW_FPGA}	±50	50	50	DQS と対応する DQ は近くに配置されているため、グローバルクロックラインのスキューは小さい値
T _{PCB_LAYOUT_SKEW}	±50	50	50	ボード上のデータラインと対応するストロープ間のスキュー
不確定値の合計	-	1180	790	不確定値の合計
データ有効ウィンドウ	530	1180	1710	

図 8 に、読み出しデータパスのタイミング マージンを示します。

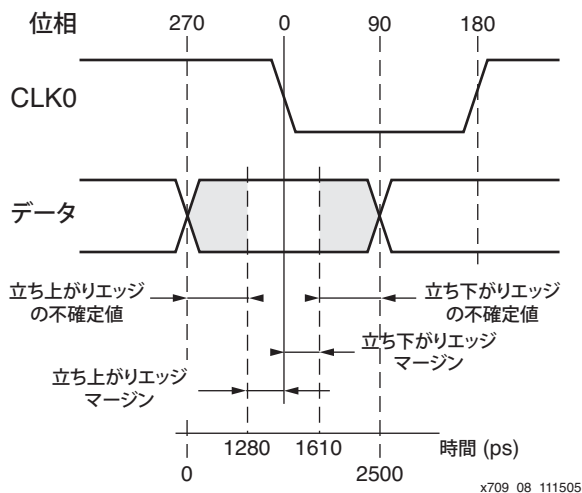


図 8：読み込みデータパスのタイミング マージン

DDR SDRAM インターフェイス のインプリメン テーション

このセクションでは、DDR SDRAM コントローラおよびインターフェイスの特性、インターフェイスのブロック図 (図 9)、コントローラ ステート マシン (図 10) を示します。

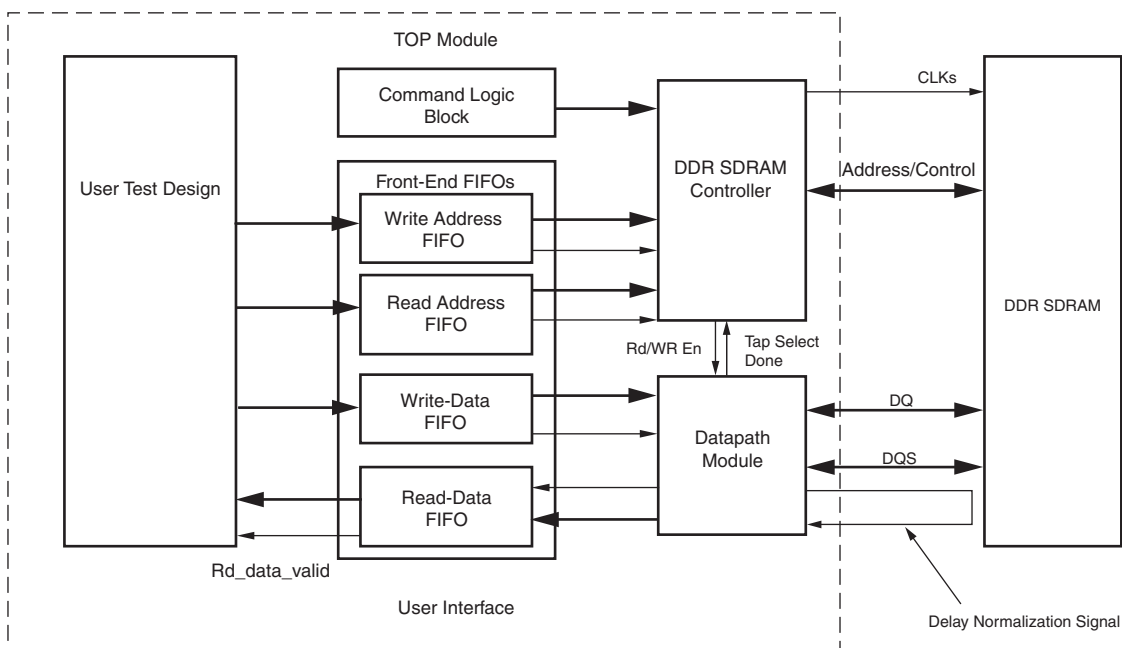


図 9：コントローラ デザインのブロック図

ハードウェア テスト ベンチ

ハードウェア テストベンチは、DDR SDRAM コントローラのデザインをすべての面からテストするためのアドレスとデータ パターンを提供します。ユーザー バックエンドには、バックエンド ステート マシン、読み出しデータ コンパレータ、およびデータ生成モジュールが含まれます。データ生成モジュールでは、メモリ デバイスに書き込まれるさまざまなアドレスおよびデータ パターンが生成されます。アドレス位置はブロック RAM にあらかじめ保存されており、ここでは ROM として使用されます。保存されているアドレス値は、DDR SDRAM デバイスの異なる行およびバンクへのアクセスをテストするために選択されています。データ パターン ジェネレータには、データ パターンを生成するステート マシンが含まれています。バックエンド ステート マシンはユーザー デザインとして機能し、書き込みイネーブル信号または読み込みイネーブル信号を送信して、データ ジェネレータ モジュールでアクセスする必要のある FIFO を指定します。

ユーザー インターフェイス

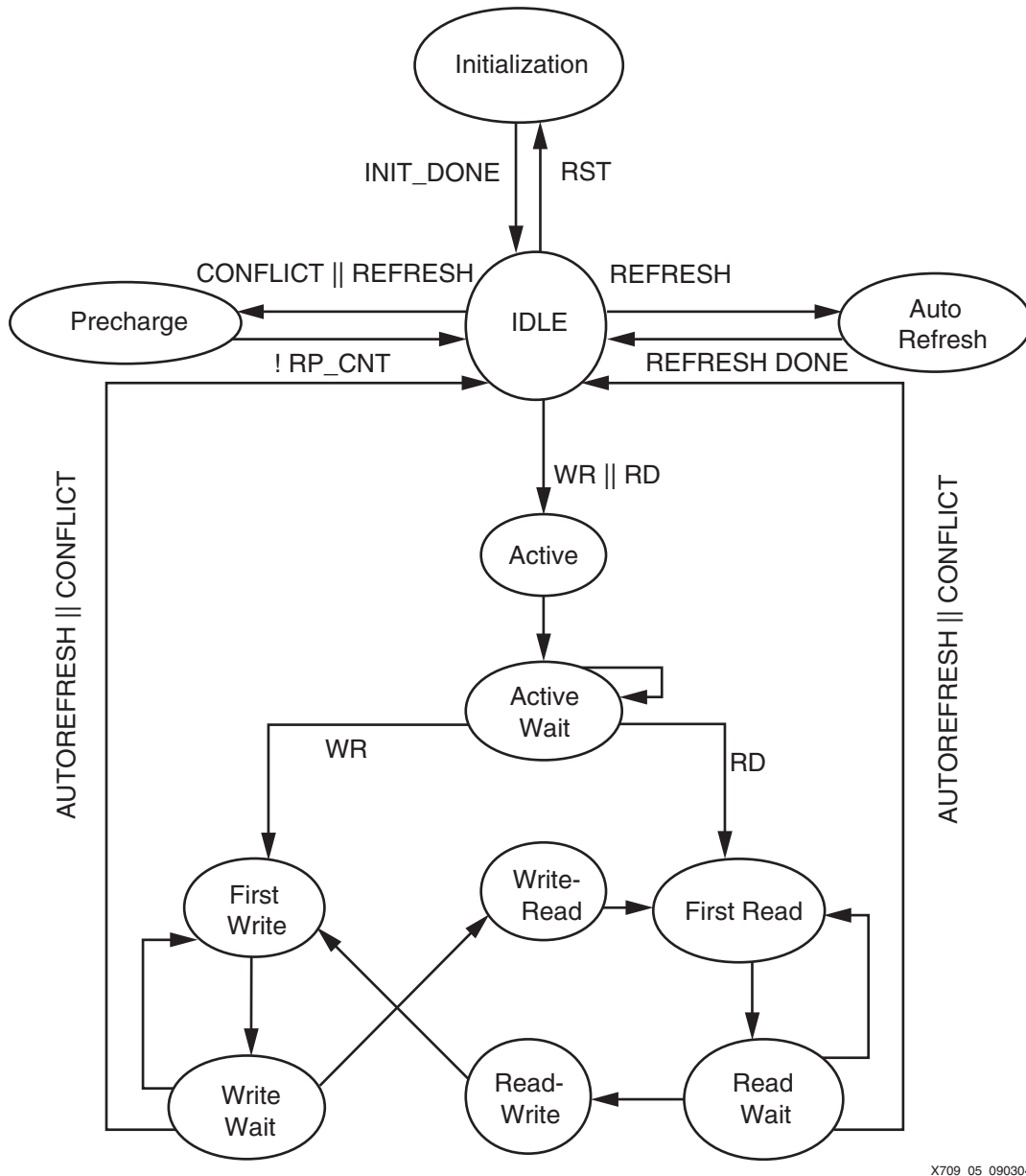
バックエンド ユーザー インターフェイスは、書き込みアドレス FIFO、書き込みデータ FIFO、読み出しアドレス FIFO、および読み出しデータ FIFO の 4 つの FIFO で構成されています。最初の 3 つの FIFO はユーザー バックエンド モジュールによりアクセスされ、読み出しデータ FIFO は取り込まれた読み出しデータを保存するためにデータパス モジュールによりアクセスされます。

コマンド論理ブロック

コマンド論理ブロックは、コントローラにコマンドを送信します。このブロックは、ユーザーのコマンド生成モジュールで変更したり、コマンド FIFO と置き換えることができます。このデザインで使用されるコマンド生成モジュールは、READ コマンドと WRITE コマンドを交互に生成します。コマンド論理ブロックは、コントローラにコマンドを送信する単純なステート マシンです。

DDR SDRAM コントローラ インターフェイス

図 10 に、DDR SDRAM コマンド生成ステート マシンを示します。



X709_05_090304

図 10 : DDR SDRAM コントローラ ステート マシン

コントローラからメモリにコマンドが送信される前に、次の操作が実行されます。

1. コマンド論理ブロックにより **WRITE** または **READ** コマンドが生成されます。
2. コントローラにより書き込みまたは読み出しアドレス **FIFO** にイネーブル信号が送信されます。
3. すべてのバンクがプリチャージされている場合は対応するバンクの行がアクティブになり、既にアクティブな行がある場合は、その行とバンクのアドレスが新しい行とバンクのアドレスと比較されます。異なる場合は、現在アクティブな行がプリチャージされ、**READ/WRITE** ステートになるまえに **ACTIVE** コマンドが発行されます。
4. **WRITE** ステートのときに **READ** コマンドが検出されると、**write_to_read** タイム後に **READ** コマンドが発行されます。同様に、**READ** ステートのときにコマンド論理ブロックから **WRITE** コマンドが検出されると、**read_to_write** タイム後に **WRITE** コマンドが発行されます。
5. コマンドは、**DDR** メモリに対して発行される前に、パイプライン化されてアドレス信号に同期化されます。

リファレンス デザインの仕様

ダイレクト クロッキング データ キャプチャ手法を使用した **DDR SDRAM** メモリ コントローラのリファレンス デザインは、**Memory Interface Generator (MIG)** ツールから使用できます。このツールは、ザイリンクスの **CORE Generator™** に含まれています。最新のリファレンス デザインを入手するには、次のザイリンクスの **Web** サイトの **IP** アップデートをダウンロードしてください。

http://www.xilinx.co.jp/xlnx/xil_sw_updates_home.jsp

このデザインは、ハードウェア上で **200MHz** 以上の周波数で特性評価されています。表 5 に、リファレンス デザインの仕様を示します。

表 5: リファレンス デザインの仕様

パラメータ	仕様/詳細
動作周波数	200 MHz (DDR400 - PC3200)
Virtex-4 デバイスのスピード グレード	10
テストベンチおよび ChipScope™ を使用した場合のコンポーネント デザインのデバイス使用率	1285 スライス
DIMM インターフェイスのみのスライス数	1900 スライス
コンポーネント インターフェイスのみのスライス数	525 スライス
サポートされるバースト モード	4 (2 および 8 に手動変更可能)
サポートされる CAS レイテンシ	3 (2 および 2.5 に手動変更可能)
HDL 言語	Verilog、VHDL
バス幅	コンポーネント 16 ビット、DIMM 144 ビット
コンポーネントの検証に使用されたデバイス	Micron MT46V32M16
DIMM の検証に使用されたデバイス	Micron MT18VDDF6472AG-40BG4

デザイン ファイル

表 6 に、リファレンス デザイン ファイルを示します。

表 6 : デザイン ファイルのリスト (Verilog バージョンの説明を含む)

モジュール名	各モジュールの説明
top.v	DDR コントローラと物理レイヤの最上位モジュール
clk_module.v	メモリ インターフェイスの DCM_BASE プリミティブおよび IDELAYCTRL モジュール用に 200MHz クロックをインスタンス化します。リファレンス デザインでは、次の 2 つのグローバル クロック リソースを使用しており、これらはファイルを手動で編集して保存できます。 <ul style="list-style-type: none"> • DCM の CLKDV 出力から生成されたカウンタ クロック。デザインのタイミングが満たされている場合は、大型のカウンタを使用するとクロック リソースを節約できます。 • IDELAY_CTRL モジュール用のクロック。エンジニアリング サンプル デバイス用に低周波数のクロックを供給します。製品デバイスを使用する場合は、このモジュールはメインのインターフェイス クロックを使用します。
data_path.v	物理レイヤの最上位。サブ モジュール tap_ctrl、data_tap_inc、idelay_ctrl、idelay_rd_en、v4_dqs_iob、v4_dq_iob、および rd_data_fifo をインスタンス化します。
data_tap_inc.v	ストロープに対応するデータ ビットのタップ選択コントローラをインプリメントします。
idelay_ctrl.v	デザインに IDELAY プリミティブが使用されている場合に必要な IDELAYCTRL プリミティブをインスタンス化します。
idelay_rd_en.v	正規化された読み出しイネーブル信号用に IDELAY プリミティブおよび IOB フリップフロップをインスタンス化します。
tap_ctrl.v	DQS 信号の 2 つの遷移を検出し、対応するデータ ビットを FPGA の内部クロック (CLK) の中心に揃えるために必要なタップ遅延を判断するモジュール。
v4_dm_iob.v	双方向データ用に IDELAY プリミティブおよび IOB フリップフロップをインスタンス化します。
v4_dq_iob.v	双方向データ用に IDELAY プリミティブおよび IOB フリップフロップをインスタンス化します。
v4_dqs_iob.v	双方向ストロープ用に IDELAY プリミティブおよび IOB フリップフロップをインスタンス化します。
controller.v	<ul style="list-style-type: none"> • 書き込みアドレス、書き込みデータ、読み出しアドレス FIFO に読み出しイネーブル信号を送信します。 • コントローラ ステート マシンを含みます。DDR デバイスに対して適切なコマンド 信号を送信します。AUTO REFRESH コマンドは、コントローラによりコマンドの間隔を考慮して生成されます。 • DDR デバイスに対してアドレス信号を送信します。
test_bench.v	メモリ インターフェイス用の合成可能なテストベンチ。
backend_rom.v	メモリ アレイに書き込み、アクセスするデータおよびアドレスを保存します。

表 6 : デザイン ファイルのリスト (Verilog バージョンの説明を含む) (続き)

モジュール名	各モジュールの説明
cmp_rd_data.v	ビット エラーが発生した場合に、エラー信号を生成します。読み込みデータを予測されるデータ値と比較します。
user_interface.v	読み込みデータ用の FIFO16 プリミティブをインスタンス化します。FIFO は、立ち上がりエッジのデータ用および立ち下がりエッジのデータ用にそれぞれ 1 つずつあります。
backend_fifo.v	DDR SDRAM インターフェイス用のバックエンド FIFO をインスタンス化します。書き込みアドレス、書き込みデータ、読み出しアドレス、読み出しデータ FIFO が含まれます。FIFO は、Virtex-4 FIFO16 プリミティブまたは LUT RAM インスタンスを使用して作成された FIFO を使用してインプリメントされます。
Command_logic.v	インターフェイスのアービタが含まれます。このモジュールを編集して、読み出しおよび書き込みアクセスの条件などを変更できます。
Rd_data_fifo.v	読み込みデータ用の FIFO16 プリミティブをインスタンス化します。FIFO は、立ち上がりエッジのデータ用および立ち下がりエッジのデータ用にそれぞれ 1 つずつあります。
RAM_8D.v	非同期 FIFO を構築するために LUT RAM をインスタンス化します。クロックの立ち上がりエッジに同期してデータを送信します。
parameter.v	DDR SDRAM リファレンス デザインを 200MHz で動作させる場合に使用される値が含まれます。

デザイン階層

図 11 に、リファレンス デザインの階層を示します。



図 11： デザイン階層

改訂履歴

次の表に、この文書の改訂履歴を示します。

日付	バージョン	改訂内容
2004年9月10日	1.0	初期リリース
2004年11月16日	1.1	「ACTIVE コマンド」および「WRITE コマンド」セクションを変更。リファレンス デザインへのリンクを追加。表 6 を完了。
2005年4月1日	1.2	図 2、図 5、「デザイン階層」、表 5、表 6 を更新。ML461 プラットフォームでのハードウェアテストの結果でリファレンス デザインをアップデート。「タイミング解析」を追加。
2005年8月27日	1.3	表 2 および表 3 をわかりやすく変更。ChipScope ファイルを使用してリファレンス デザインをアップデート。
2005年11月18日	1.4	リファレンス デザインのリンクをアップデート。(「リファレンス デザインの仕様」を参照)。読み出しデータのタイミング解析 (表 4 を参照) および 図 11 を更新。