

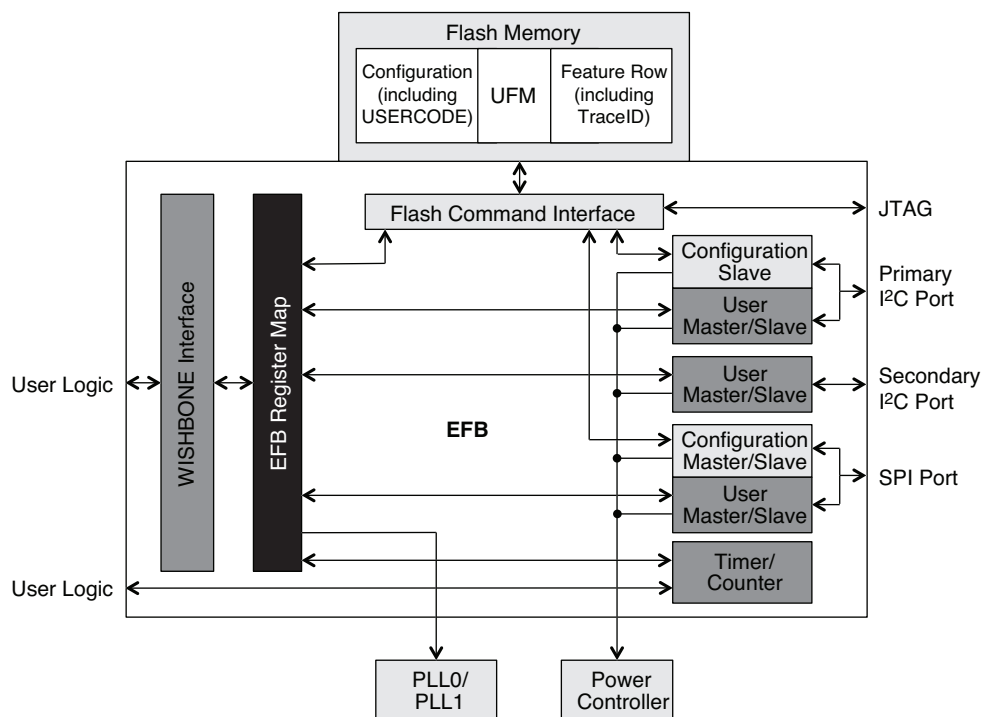
概要

MachXO2™ FPGA ファミリーは、高性能、低消費電力の FPGA ファブリックに、ハードマクロ化された制御機能とオンチップ UFM (User Flash Memory、ユーザフラッシュメモリ) が組み込まれています。ハードマクロ化された制御機能によって、デザインの実装が容易になり、LUT、レジスタ、クロック、配線などの汎用リソースを節約できます。このハードマクロ化された制御機能は、物理的には EFB (Embedded Function Block、組み込み機能ブロック) 内に配置されます。EFB モジュールは全ての MachXO2 デバイスに内蔵されていて、次の制御機能を持っています。

- ・ 2 系統の I²C コア
- ・ 1 系統の SPI コア
- ・ 1 個の 16 ビット・タイマ / カウンタ
- ・ 以下を含むフラッシュメモリ・インターフェイス
 - MachXO2-640 以上の規模ではユーザフラッシュメモリ (UFM)
 - コンフィグレーション・ロジック
- ・ PLL のダイナミック・コンフィグレーションへのインターフェイス
- ・ オンチップ・パワーコントローラへのインターフェイス (I²C 及び SPI 経由)

図 9-1 に EFB のアーキテクチャ、及び FPGA ユーザロジック用の WISHBONE インターフェイスを示します。

図 9-1. EFB (組み込み機能ブロック)



EFB に含まれるハード SPI と I²C、およびタイマ / カウンタ IP は、ラティスの参照デザインを使用して同じ機能を FPGA ロジックで実装した場合と比較して、500LUT 以上の節約を可能にします。

EFB レジスタマップは、スレーブ WISHBONE バスを通じた EFB ハードマクロ機能へのアクセスに使用されます。ハード IP はそれぞれ専用の 8 ビット・データ及び制御レジスタを持ちますが、例外としてフラッシュメモリ (UFM/ コンフィグレーション) は同じレジスタセットからアクセスされます。EFB レジスタマップにアクセス可能なポートは、全レジスタにアクセスできます。例えば、プライマリ I²C スレーブポートはタイマ / カウンタ・レジスタにアクセスできます。EFB レジスタマップを次に示します。

表 9-1. EFB メモリマップ

アドレス範囲 (16 進)	8 ビット・データ / 制御レジスタの機能
0x00-0x1F	PLL0 の動的アクセス ¹
0x20-0x3F	PLL1 の動的アクセス ¹
0x40-0x49	プライマリ I ² C
0x4A-0x53	セカンダリ I ² C
0x54-0x5D	SPI
0x5E-0x6F	タイマ / カウンタ
0x70-0x75	フラッシュメモリ (UFM/ コンフィグレーション)
0x76-0x77	EFB の割り込みソース

1. MachXO2 デバイスには最大 2 個の PLL があります。PLL0 のアドレス範囲は 0x00 ~ 0x1F。PLL1 (もしあれば) のアドレス範囲は 0x20 ~ 0x3F。PLL のコンフィグレーション・レジスタ及び推奨する使用法の詳細については、“TN1199, MachXO2 sysCLOCK PLL Design and Usage Guide (MachXO2 sysCLOCK PLL 設計と使用法ガイド)” を参照してください

設計ソフトウェアには EFB モジュールがプリミティブとして扱われ、本書で説明します。EFB の構成、及び Verilog または VHDL ソースコードの生成には IPexpress™を使用します。ソースコードはデザイン内にインスタンス化します。

WISHBONE バスインターフェイス

MachXO2 内の WISHBONE バスは、OpenCores の WISHBONE 標準に準拠し、FPGA ユーザロジックと EFB 機能ブロック間、及び各 EFB 機能ブロック間を接続します。ユーザロジックには、EFB の WISHBONE スレーブ・インターフェイスと通信するための、WISHBONE マスタ・インターフェイスを含める必要があります。WISHBONE マスタの例としては LatticeMico8™があります。

図 9-2 のブロック図は、FPGA コアと EFB との間の WISHBONE バス信号を示しています。表 9-2 に信号の詳細な定義を示します。

図 9-2. FPGA コアと EFB モジュール間の WISHBONE バスインターフェイス

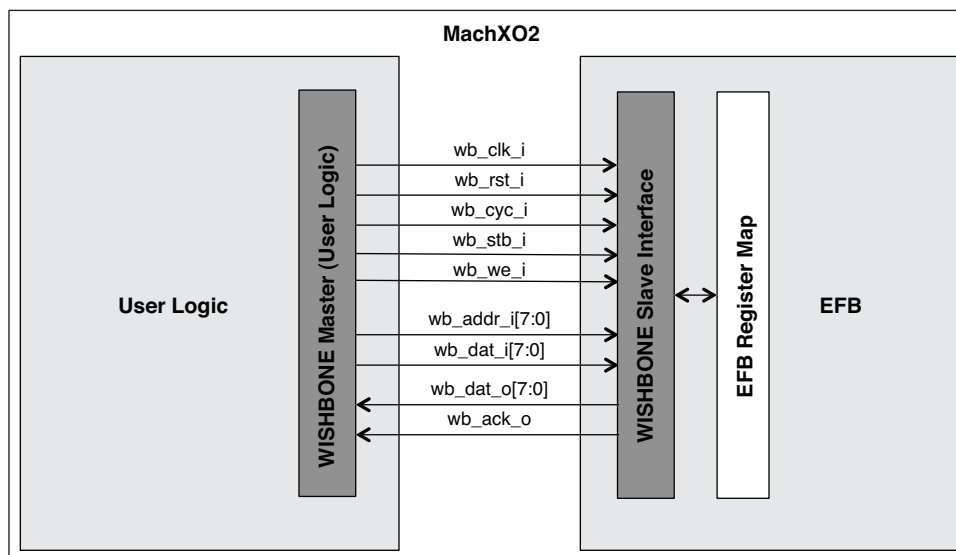


表 9-2. EFB モジュールの WISHBONE スレーブ・インターフェイス信号

信号名	I/O	幅	記 述
wb_clk_i	入力	1	EFB モジュール内の WISHBONE インターフェイスのレジスタ及びハードマクロ機能で使用されるクロック。立ち上がりエッジ有効で、クロック速度は最速 133MHz まで対応
wb_rst_i	入力	1	WISHBONE インターフェイス・ロジックをリセットする同期リセット信号で、レジスタ値には影響を与えない。アクティブなバスサイクルを終端する。ネゲート後に 1us 待ってから、その後の WISHBONE トランザクションを開始すること
wb_cyc_i	入力	1	WISHBONE マスタによってアサートされ、バスに有効なバスサイクルが存在することを示す
wb_stb_i	入力	1	バス上の現在のトランザクションのターゲットが、WISHBONE スレーブであることを示すストローブ信号。このストローブのアサートに対する応答として、EFB モジュールがアクノレッジをアサートする
wb_we_i	入力	1	レベルセンシティブなライト / リード制御信号。Low はリード動作、High はライト動作を示す
wb_adr_i	入力	8	EFB 固有のレジスタの選択に使用される 8 ビット幅アドレス
wb_dat_i	入力	8	リード・サイクル中に WISHBONE マスタが wb_dat_i バスサイクルを使用して、アドレス指定された EFB レジスタにデータを書き込む
wb_dat_o	出力	8	リード・メモリ・サイクル中に WISHBONE マスタが wb_dat_o を使用して、アドレス指定されたレジスタからデータを読み出す
wb_ack_o	出力	1	バスサイクルが完了し、EFB へのデータ書き込みが了承されたことを、WISHBONE マスタに知らせる。EFB からのデータリードは有効

EFB とインターフェイスするには、ユーザロジック内に WISHBONE マスタコントローラを作成する必要があります。複数マスタ構成では、WISHBONE 出力がユーザ定義のアービタで多重化されます。LatticeMico8 ソフトプロセッサは、マルチマスタ・バス構成を実装可能な Mico System Builder (MSB) プラットフォームで利用することもできます。同じサイクル中に 2 つのマスタがバスを要求した場合は、アービトレーションで選択された出力だけがスレーブ・インターフェイスに接続されます。

WISHBONE プロトコル

WISHBONE のプロトコルとコマンドシーケンスについては、“TN1246, Reference Guide for Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (MachXO2 デバイスのユーザフラッシュメモリとハードマクロ制御機能の使用法リファレンスガイド)” の WISHBONE 項を参照してください。

WISHBONE 設計のヒント

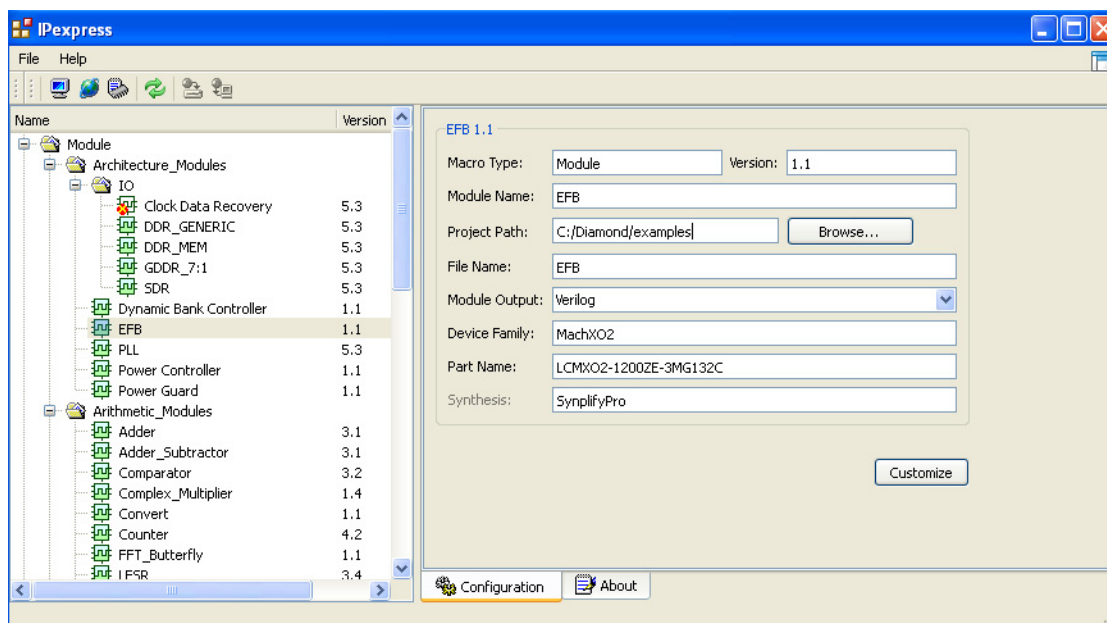
- 消費電力削減のためコンポーネントを動的にオフする場合に、WISHBONE クロックのソースになっているかどうかにかかわらず、EFB は MachXO2 の内部オシレータをイネーブルにする必要があります。内部オシレータがディセーブルされている場合は、以下の機能が使用できます。
 - SPI : ユーザスレーブ、またはユーザマスタ
 - I2C : I2C_1_CR[3:2]=11 と設定することにより SDA 遅延をオフにした後、内部オシレータをオフにできます。その後内部オシレータをディセーブルにしてユーザスレーブ、またはユーザマスタで動作できます
 - タイマカウンタ : ユーザクロックが選択されなければなりません
- 未使用の EFB WISHBONE の入力信号は '0' に接続する必要があります
- WISHBONE の仕様の詳細については、OpenCores のウェブサイトを参照してください
- ラティスの参照デザインの多くが WISHBONE バスを持っています

(www.latticesemi.com/products/intellectualproperty/aboutreferencedesigns.cfm)

IPexpress による EFB モジュールの生成

IPexpress は、EFB のハードマクロ化 IP 機能の構成（モード、パラメータ設定など）と、EFB モジュールの生成に使用されます。Lattice Diamond® のトップメニューで、**Tools → IPexpress** を選択して起動します。Diamond プロジェクトで MachXO2 デバイスをターゲットとすると、IPexpress ウィンドウの **Modules → Architecture Modules**（アーキテクチャ・モジュール）下に EFB モジュールが表示されます。

図 9-3. IPexpress の EFB モジュール

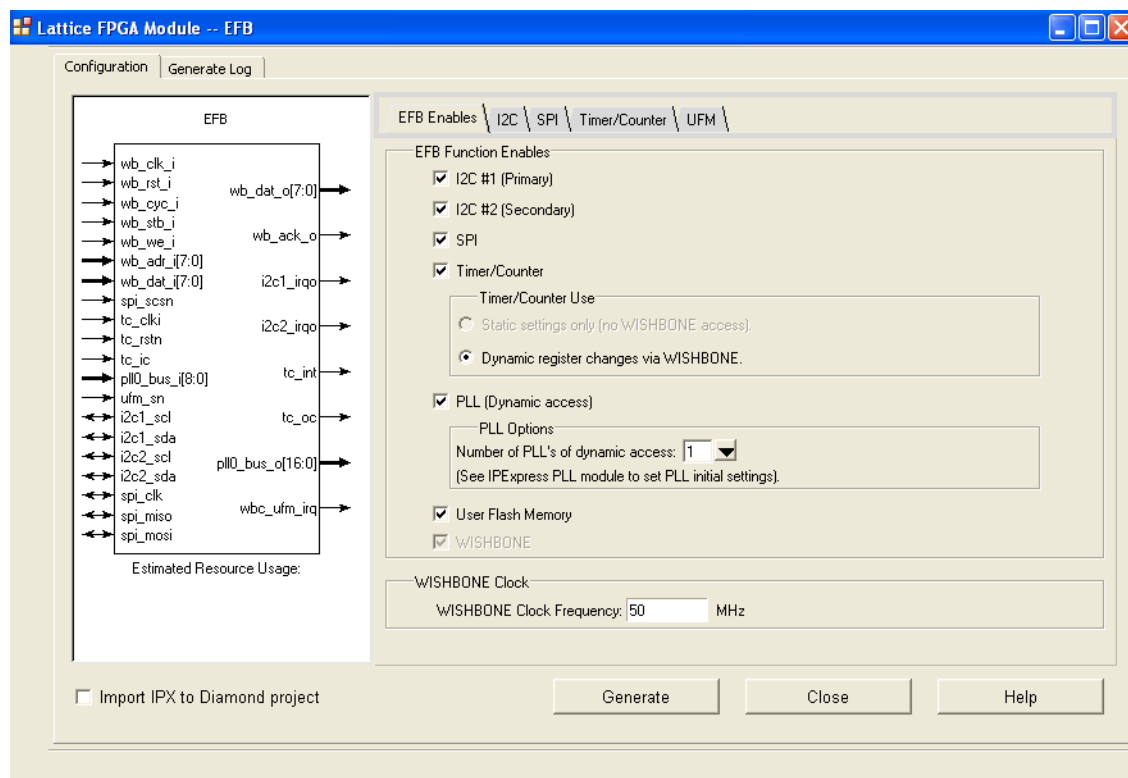


Project Path（プロジェクトパス）、File Name（ファイル名 = モジュール名）、Module Output（VHDL または Verilog）の各フィールドに入力・設定し、**Customize**（カスタム化）をクリックします。

Customize ボタンをクリックすると、EFB の構成ダイアログが表示されます。EFB ウィンドウの左側には、各 IP 機能に関連する I/O がグラフィカルに表示されます。各 IP をイネーブルまたはディセーブルにすると、その I/O ピンが表示されたり、非表示されたりします。最初のタブ (EFB Enables) では、イネーブルするハードマクロ機能の選択、PLL コンフィグレーション設定への動的アクセス、UFM (ユーザフラッシュメモリ) のイネーブル、及び WISHBONE クロック周波数の入力に使用します。全ての機能をイネーブルにした EFB の例を図 9-4 に示します。

ハードマクロ IP 機能と UFM には、EFB ウィンドウにそれぞれ個別のコンフィグレーション設定タブがあります。これらのタブについては、機能ごとに後述します。全ての機能の構成後、**Generate** (生成) ボタンをクリックすると、EFB モジュールが生成されてデザイン内にインスタンス化できるようになります。

図 9-4. IPexpress による EFB モジュールの生成



利用可能な PLL モジュール数はデバイスの規模によって異なり、IPexpress の EFB の GUI はそれを反映したのものになります。MachXO2-256 と MachXO2-640 には PLL モジュールがないため、EFB ウィンドウの PLL チェックボックスが選択できないようになっています。MachXO2-640U、MachXO2-1200、MachXO2-1200U、及び MachXO2-2000 にはそれぞれ PLL が 1 個あり、MachXO2-2000U、MachXO2-4000、及び MachXO2-7000 にはそれぞれ PLL が 2 個で、EFB の WISHBONE スレーブ・インターフェイスから動的にアクセスできます。

WISHBONE クロック周波数はデフォルトで 50MHz に設定され、最高 133MHz までの値を入力できます。WISHBONE クロックは EFB の WISHBONE インターフェイス・レジスタで使用され、SPI 及び I²C ハードマクロ IP コアでも使用されます。EFB の多くの機能では、WISHBONE のクロックソースになっているかどうかにかかわらず、MachXO2 の内部オシレータをイネーブルする必要があります。内部オシレータがディセーブルされている場合は、以下の機能が使用できます。

- SPI : ユーザスレーブ、またはユーザマスタ
- I2C : I2C_1_CR[3:2]=11 と設定することにより SDA 遅延をオフにした後、内部オシレータをオフにできます。その後内部オシレータをディセーブルにしてユーザスレーブ、またはユーザマスタで動作できます

- タイマカウンタ：ユーザクロックが選択されなければなりません

他のモジュールと同様、EFB の設定は次に示すように Map Report（マップレポート）に表示されます。

Embedded Functional Block Connection Summary:

```
-----
Desired WISHBONE clock frequency: 2.0 MHz
Clock source:                      clk
Reset source:                      wb_rst
Functions mode:
  I2C #1 (Primary) Function:       ENABLED
  I2C #2 (Secondary) Function:     DISABLED
  SPI Function:                   ENABLED
  Timer/Counter Function:         DISABLED
  Timer/Counter Mode:             WB
  UFM Connection:                 DISABLED
  PLL0 Connection:                DISABLED
  PLL1 Connection:                DISABLED
```

I2C Function Summary:

```
-----
I2C Component:                    PRIMARY
I2C Addressing:                   7BIT
I2C Performance:                  100kHz
Slave Address:                    0b0001001

General Call:                     ENABLED
I2C Wake Up:                      DISABLED
I2C Component:                    UFM/Configuration
I2C Addressing:                   7BIT
I2C Performance:                  100kHz
Slave Address:                    0b0001000
```

SPI Function Summary:

```
-----
SPI Mode:                         BOTH
SPI Data Order:                   LSB to MSB
SPI Clock Inversion:              DISABLED
SPI Phase Adjust:                 DISABLED
SPI Wakeup:                       DISABLED
```

Timer/Counter Function Summary:

None

UFM Function Summary:

```
-----
UFM Utilization:                  EBR Initialization
Available General
Purpose Flash Memory:            511 Pages (511*128 Bits)
```

EBR Blocks with Unique
Initialization Data: 6

WID	EBR Instance
---	-----

```

0b0000000011 LCDCharMap_inst/LCDCharMap_0_0_0
0b00000000100 STRING_TABLE_INST/EXT_ROM_INST/pmi_romXhmenusdn8101024_0_0_0
0b00000000101
lm8_inst/u1_isp8/u1_isp8_prom/pmi_romXhprom_initadn18112048_1_1_0
0b00000000110
lm8_inst/u1_isp8/u1_isp8_prom/pmi_romXhprom_initadn18112048_0_0_3
0b00000000111
lm8_inst/u1_isp8/u1_isp8_prom/pmi_romXhprom_initadn18112048_0_1_2
0b00000001000
lm8_inst/u1_isp8/u1_isp8_prom/pmi_romXhprom_initadn18112048_1_0_1

```

ハードマクロ I²C IP コア

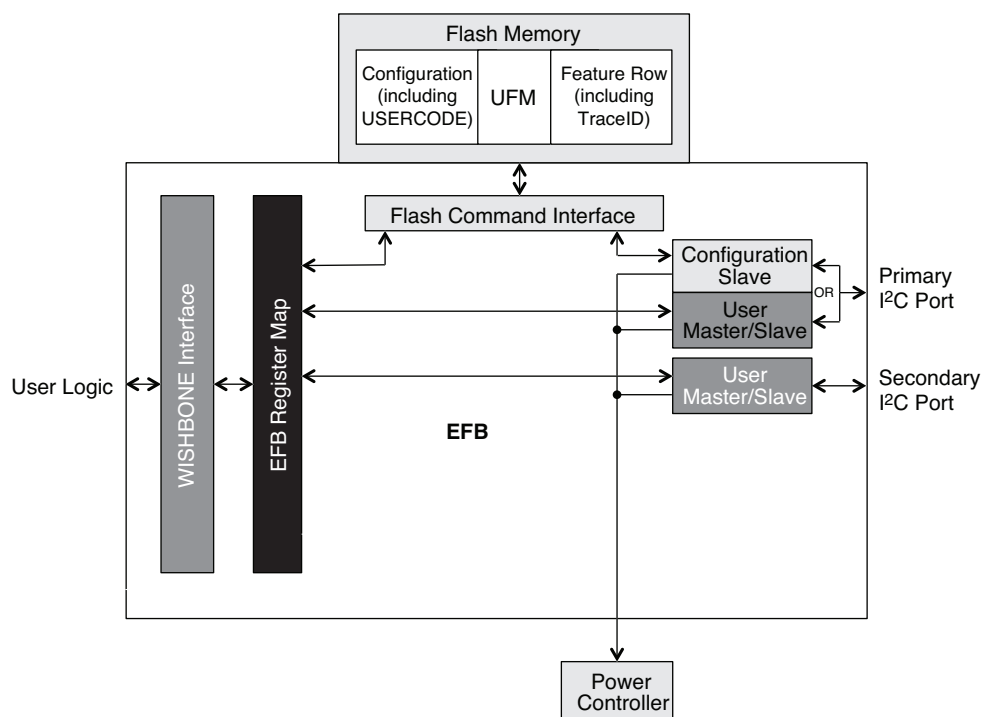
I²C は 2 線式シリアルバスで、同一基板上のデバイス間通信用として幅広く使用されています。各 MachXO2 デバイスには、“プライマリ”及び“セカンダリ” I²C として設計された、2 系統のハードマクロ化 I²C IP コアがあります。MachXO2 内の両コアは、I²C マスタまたは I²C スレーブとして動作できます。この 2 つのコアの違いは、プライマリコアには I/O ピンが事前割り当てされているのに対して、セカンダリコアのポートは任意の汎用 I/O に割り当て可能なことです。また、プライマリコアはフラッシュコマンド・インターフェイス経由で、フラッシュメモリ（UFM/ コンフィグレーション）にアクセスすることもできます。ハードマクロ I²C IP コアの機能とブロック図を、次に示します。

表 9-3. ハードマクロ化 I²C の機能

	プライマリ I ² C コンフィグレーション	プライマリ I ² C ユーザ	セカンダリ I ² C ユーザ
I ² C ポート、マスタ	No	Yes	Yes
I ² C ポート、スレーブ	Yes ¹	Yes ¹	Yes
フラッシュメモリ（UFM/ コンフィグレーション）のアクセス	Yes ¹	No	No
ユーザロジックのアクセス	No	Yes	Yes
専用 I/O の使用の必要性	Yes	Yes	No
パワーコントローラのスタンバイモードからの起動	Yes	Yes	Yes
パワーコントローラのスタンバイモードへの移行	Yes	No	No

1. プライマリポートは UFM/ コンフィグレーションのアクセスポートとして、或いはユーザ用途に用いることができますが、両方同時にはできません

図 9-5. I²C ブロック図



EFB の I²C コアがマスタの場合は、物理的なインターフェイスを通じて I²C バス上の他のデバイスを制御できます。EFB の I²C コアがスレーブの場合、そのデバイスは I²C マスタへの I/O を拡張できます。MachXO2 のプライマリ及びセカンダリコアはどちらも、次の I²C 機能に対応しています。

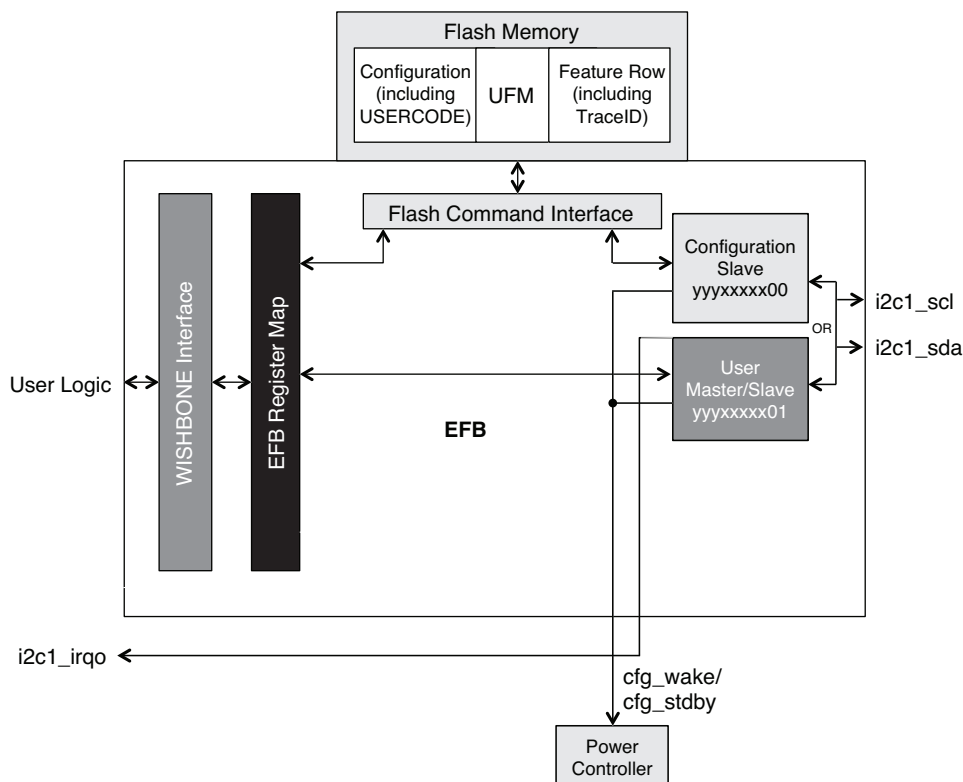
- ・ マスタ / スレーブモードに対応
- ・ 7ビット及び10ビット・アドレッシング
- ・ 50KHz、100KHz、及び400KHz のデータ転送速度に対応
- ・ 同報通知 (General Call) に対応 (I²C アドレス 0 を使用して、バス上の全てのデバイスをアドレス指定)
- ・ EFB WISHBONE スレーブ・インターフェイスからユーザロジックへのインターフェイス

プライマリ I²C

MachXO2 のプライマリ I²C コントローラを図 9-6 に示します。プライマリ・コントローラの主要機能には、次のものがあります。

- ・ 次の一方
 - I²C コンフィグレーション・スレーブとして、フラッシュメモリ (UFM/ コンフィグレーション) へのアクセス、または
 - I²C ユーザスレーブとしてユーザロジックへのアクセス
- ・ I²C コンフィグレーションまたはユーザスレーブとして、MachXO2 パワーコントローラへのアクセス
- ・ I²C ユーザマスタとして、MachXO2 に接続された周辺回路へのアクセス

図 9-6. プライマリ I²C のブロック図



プライマリ I²C コアは UFM へのアクセスと CFM のプログラミングに使用できます。しかしプライマリ I²C コアは、単一のデザインでは UFM/CFM かユーザ・ファンクションのどちらか一方にしか用いることはできません。図 9-6 のブロック図は、I²C ブロックとフラッシュメモリ間のインターフェイスを示しています。I²C ポートからの MachXO2 のプログラミングの詳細については、“TN1204, MachXO2 Programming and Configuration Usage Guide (MachXO2 プログラミングとコンフィグレーションのユーザガイド)” の I²C 項を参照してください。

バス上のスレーブ I²C 周辺回路は、そのスレーブ固有のアドレスを呼び出すことで、マスタ I²C からアクセスできます。プライマリ・コンフィグレーション・アドレスは “yyyxxxxx00”、プライマリ・ユーザアドレスは “yyyxxxxx01” で、“y” と “x” は IPexpress からユーザがプログラム可能です。

プライマリ・コンフィグレーション I²C は、パワーコントローラのスタンバイからのウェイクアップ、またはスタンバイへの移行に使用できます。プライマリ・ユーザは、パワーコントローラのスタンバイモードからのウェイクアップだけに使用できます。パワーコントローラの詳細については、“TN1198, Power Estimation and Management for MachXO2 Devices (MachXO2 デバイスの電源見積りと管理)” を参照してください。I²C パワーコントローラの機能は、後述するように IPexpress から設定できます。レジスタ設定については、“TN1246, Reference Guide for Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (MachXO2 デバイスのユーザフラッシュメモリ及びハードマクロ制御機能の使用法リファレンスガイド)” に説明があります。

表 9-4 にプライマリ I²C コアの IP 信号を示します。

表 9-4. プライマリ I²C ~ IP 信号

Signal Name	Pre-Assigned Pin Name	I/O	Width	Description
i2c1_scl	SCL	Bi-directional	1	I²C コアのオープンドレイン・クロックライン ~ I ² C コアがマスタ動作を実行中、信号は出力。スレーブ動作時、この信号は入力 この信号は、事前に割り当てられた対応するピンに直接配線する。各 MachXO2 デバイスの I ² C ポートのパッドとピンの位置については、“MachXO2 Family Data Sheet (MachXO2 ファミリー・データシート)” のピンテーブルを参照
i2c1_sda	SDA	Bi-directional	1	I²C コアのオープンドレイン・データライン ~ I ² C コアがデータを送信時、信号は出力。I ² C コアがデータを受信時、この信号は入力 この信号は、事前に割り当てられた対応するピンに直接配線する。各 MachXO2 デバイスの I ² C ポートのパッドとピンの位置については、“MachXO2 Family Data Sheet (MachXO2 ファミリー・データシート)” のピンテーブルを参照
i2c1_irqo	—	Output	1	I²C コアの割り込み要求出力信号 ~ この信号の本来の用途は、WISHBONE マスタコントローラ（マイクロコントローラまたはステートマシン）に接続し、特定の条件に適合した場合に割り込みを要求すること。その条件については、TN1246 の I ² C 項を参照
cfg_wake	—	Output	1	ウェイクアップ信号 ~ 機能シミュレーションのため、MachXO2 デバイスのパワーコントローラのみに接続するハードワイヤ信号。この信号がイネーブルとなるのは、EFB GUI の I ² C タブで Wakeup Enable（ウェイクアップ・イネーブル）機能がイネーブルとなっている場合のみ
cfg_stdby	—	Output	1	スタンバイ信号 ~ 機能シミュレーションのため、MachXO2 デバイスのパワーコントローラのみに接続するハードワイヤ信号。この信号がイネーブルとなるのは、EFB GUI の I ² C タブで Wakeup Enable（ウェイクアップ・イネーブル）機能がイネーブルとなっている場合のみ

セカンダリ I²C

MachXO2のセカンダリI²Cコントローラは、プライマリI²Cコントローラと同じ機能を提供しますが、MachXO2 コンフィグレーション・ロジックにアクセスできない点が異なります。i2c2_scl 及び i2c2_sda ポートは FPGA ファブリックの一般配線を用いて接続され、任意の GPIO（General Purpose I/O、汎用 I/O）に割り当てることができます。

セカンダリ I²C は、パワーコントローラのスタンバイモードからのウェイクアップに使用できます。パワーコントローラの詳細については、“TN1198, Power Estimation and Management for MachXO2 Devices (MachXO2 デバイスの電源見積りと管理)” を参照してください。I²C パワーコントローラの機能は、後述するように IPexpress から設定できます。レジスタ設定については、“TN1246, Reference Guide for Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (MachXO2 デバイスのユーザフラッシュメモリ及びハードマクロ制御機能の使用法リファレンスガイド)” に説明があります。

バス上のスレーブ I²C 周辺回路は、そのスレーブ固有のアドレスを呼び出すことで、ユーザマスタ I²C からアクセスされます。セカンダリ・ユーザアドレスは “yyyxxxxx10” で、“y” と “x” は IPexpress からユーザがプログラム可能です。図 9-7 にセカンダリ I²C コアのブロック図を示します。

図 9-7. セカンダリ I²C のブロック図

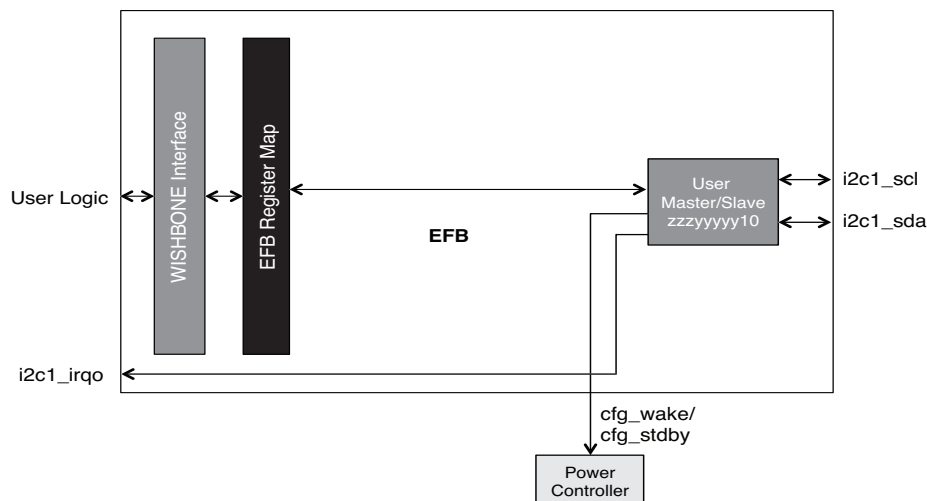


表 9-5 にセカンダリ I²C コアの IP 信号を示します。これらの信号は、MachXO2 デバイスの任意の GPIO で配線できます。

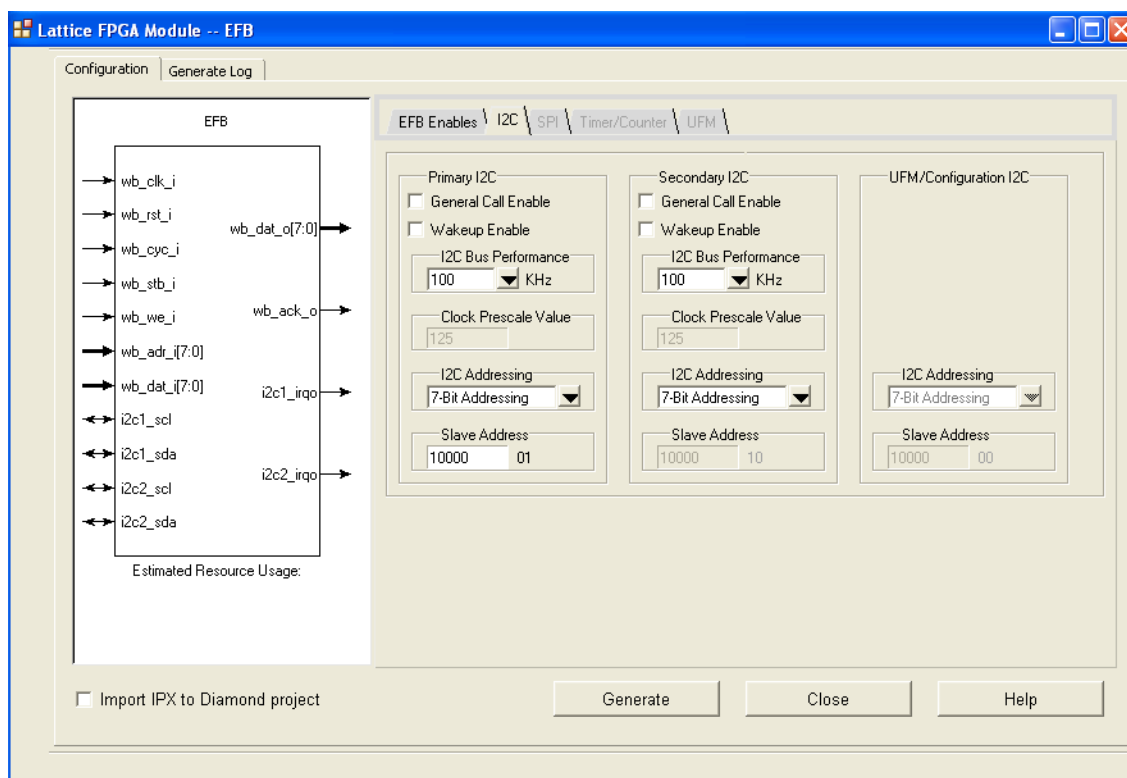
表 9-5. セカンダリ I²C ～ IP 信号

信号名	固有のピン名	I/O	幅	記 述
i2c2_scl	—	双方向	1	コアのオープンドレイン・クロックライン ~ I ² C コアがマスタ動作を実行中、信号は出力。スレーブ動作時、この信号は入力。MachXO2 の任意の GPIO で配線できる。
i2c2_sda	—	双方向	1	I ² C コアのオープンドレイン・データライン ~ I ² C コアがデータを送信時、信号は出力。I ² C コアがデータを受信時、この信号は入力。MachXO2 の任意の GPIO で配線できる。
i2c2_irqo	—	出力	1	I ² C コアの割り込み要求出力信号 ~ この信号の目的は、WISHBONE マスタコントローラ（マイクロコントローラまたはステートマシン）に接続し、特定の条件に適合した場合に割り込みを要求すること。その条件については、I ² C レジスタの定義を参照
cfg_wake	—	出力	1	ウェイクアップ信号 ~ 機能シミュレーションのため、MachXO2 デバイスのパワーコントローラのみ接続するハードワイヤ信号。この信号がイネーブルとなるのは、EFB GUI の I ² C タブで Wakeup Enable（ウェイクアップ・イネーブル）機能がイネーブルとなっている場合のみ
cfg_stdby	—	出力	1	スタンバイ信号 ~ 機能シミュレーションのため、MachXO2 デバイスのパワーコントローラのみ接続するハードワイヤ信号。この信号がイネーブルとなるのは、EFB GUI の I ² C タブで Wakeup Enable（ウェイクアップ・イネーブル）機能がイネーブルとなっている場合のみ

IPexpress による I²C コアの構成

IPexpress を使用して I²C コアを構成し、EFB モジュールを生成できます。EFB GUI で I²C タブを選択すると、I²C コアの構成可能な設定が表示されます。図 9-8 に I²C コアの構成例を示します。

図 9-8. IPexpress による EFB モジュールの I²C 機能の構成



同報通知イネーブル (General Call Enable)

この設定では、スレーブモードでの I²C 同報通知応答 (I²C アドレス 0 を使用してバス上の全デバイスに送信) がイネーブルになります。プライマリレジスタ I2C_1_CR またはセカンダリレジスタ I2C_2_CR の GCEN ビットをイネーブルにすると、この設定を動的に変更できます。

ウェイクアップ・イネーブル (Wake-up Enable)

パワーコントローラの詳細については、“TN1198, Power Estimation and Management for MachXO2 Devices (MachXO2 デバイスの電源見積りと管理)” を参照してください。

Wake-up Enable を選択すると、外部 I²C マスタが MachXO2 のスタンバイ電源状態を解除できるようになります。外部 I²C マスタが MachXO2 のウェイクに使用できる方法は、次の 2 つあります。

- ・ プライマリまたはセカンダリ・スレーブ I²C EFB アドレスの照合
- ・ 同報通知と、それに続く 0xF3 16 進コマンド Op コードの実行

I2C_1_CR または I2C_2_CR の WKUPEN ビットは動的に変更でき、ウェイクアップ機能をイネーブルまたはディセーブルにできます。

I²C バス動作速度 (I²C Bus Performance)

I²C 周波数を 50KHz、100KHz、400KHz から選択できます。これは、I²C バスの SCL クロック周波数になります。この GUI 値と、EFB Enables タブの WISHBONE Clock Frequency 属性との組み合わせから、ソフトウェアが (WB クロック) / (クロック・プリスケール値) という式を使用して、10 ビット・プリスケール・レジスタのクロック分周値を計算します。プライマリ I²C ボーレート・レジスタペア I2C_1_BR1、I2C_1_BR0、またはセカンダリ I²C ボーレート・レジスタペア I2C_2_BR1、I2C_2_BR0 にアクセスすることで、このプリスケール値を動的に変更できます。

I²C アドレッシング (I²C Addressing)

7 ビットまたは 10 ビットの I²C スレーブ・アドレッシング方式を選択できます。7 ビットアドレス及び 10 ビットアドレスの最後の 2 ビットはハードコードされ、I²C コンポーネントの 1 つを選択するようになっています。I²C アドレスのプログラマブル・ビットは I²C モジュール間で共有され、次のように定義されます。

yyyxxxxxww

ww ビットはハードコードされ、次のように定義されます

00 = プライマリ・コンフィグレーション・フラッシュメモリ (UFM/ コンフィグレーション) I²C

01 = プライマリ・ユーザ I²C

10 = セカンダリ・ユーザ I²C

11 = I²C コア選択

xxxxx ビットは IPexpress GUI を使用してプログラムでき、デフォルト値は 10000 です。

yyy ビットは、10 ビット・アドレッシングを選択した場合にプログラム可能となり、デフォルト値は 000 です。

プライマリ I²C アドレスは、フラッシュメモリ (UFM/ コンフィグレーション) I²C アドレスと長さが同じです。プライマリ及びセカンダリ I²C アドレス・サイズは、長さが異なってもかまいません。例えば、プライマリ I²C アドレスが 10 ビットで、セカンダリ I²C アドレスが 7 ビットということが可能です。

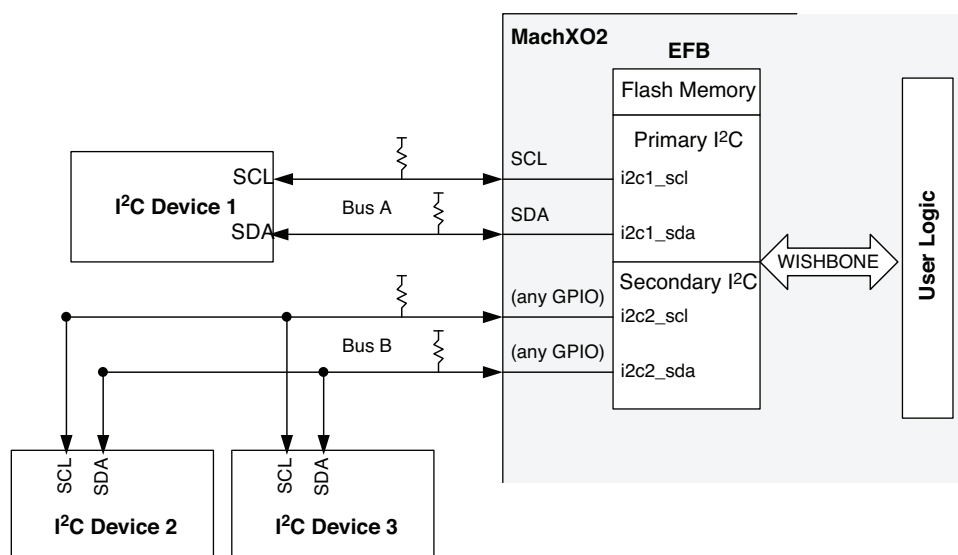
MachXO2 I²C のユースケース

以下で説明する I²C の使用例は図 9-9 を参照してください。

1. スレーブ外部 I²C デバイスにアクセスするマスタ MachXO2 I²C
 - a. MachXO2 ロジック内に WISHBONE バスマスタを実装
 - b. I²C デバイス 1、2、及び 3 は全てスレーブデバイス
 - c. WISHBONE バスマスタが EFB 内のプライマリ I²C コントローラへのトランザクションを実行し、バス A の外部スレーブ I²C デバイス 1 にアクセスする
 - d. WISHBONE バスマスタが EFB 内のセカンダリ I²C コントローラへのトランザクションを実行し、バス B の外部スレーブ I²C デバイス番号 2 または 3 にアクセスする
 - e. I²C レジスタの定義とコマンドシーケンスについては、“TN1246, Using User Flash Memory and Hardened Control Functions in MachXO2 Devices Reference Guide (MachXO2 デバイスのユーザフラッシュメモリとハードマクロ制御機能の使用法リファレンスガイド)” の I²C の項を参照
2. スレーブ MachXO2 I²C にアクセスする外部マスタ I²C デバイス
 - a. I²C デバイス 1、2、及び 3 は I²C マスタデバイス
 - b. バス A の外部マスタ I²C デバイスが I²C メモリサイクルを実行し、アドレス yyyxxxxx01 を使用して EFB プライマリ I²C コントローラにアクセスする
 - c. バス B の外部マスタ I²C デバイス 2 または 3 が I²C メモリサイクルを実行し、アドレス yyyxxxxx10 で EFB セカンダリ I²C ユーザにアクセスする
 - d. MachXO2 ファブリック内の WISHBONE バスマスタが、データの送受信を管理する必要がある。WISHBONE マスタは、割り込みまたはポーリング技術を使用してデータ転送を管理し、データのオーバーラン状態を防止できる
 - i. I²C レジスタの定義とコマンドシーケンスについては、“TN1246, Reference Guide for Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (MachXO2 デバイスのユーザフラッシュメモリとハードマクロ制御機能の使用法リファレンスガイド)” の I²C の項を参照
3. プライマリ I²C インターフェイスを使用して MachXO2 フラッシュメモリにアクセスする外部マスタ I²C デバイス

- a. バス A の外部マスタ I²C デバイス 1 が、アドレス yyyxxxxx00 を使用してバストランザクションを実行。外部マスタはこのアドレスを使用して、MachXO2 コンフィグレーション・ロジックと対話する。コンフィグレーション・ロジックは、フラッシュメモリ動作の実行に必要な制御を提供
 - b. I²C による MachXO2 デバイスのフラッシュメモリ (UFM/ コンフィグレーション) のアクセスの詳細については、本書で後述します。また、“TN1246, MachXO2 EFB Reference Guide (MachXO2 EFB リファレンスガイド)” を参照
 - c. I²C ポートからの MachXO2 のプログラミングの詳細については、“TN1204, MachXO2 Programming and Configuration Usage Guide (MachXO2 プログラミングとコンフィグレーションのユーザガイド)” の I²C 項を参照
4. 上記の使用例は相互に排他的ではありません。例えば：
- a. バス A の外部マスタデバイス 1 が、MachXO2 コンフィグレーション・ロジックにアクセスすると同時に、WISHBONE マスタがバス B の I²C スレーブデバイスにデータを転送できる
 - b. WISHBONE マスタがバス A のマイクロプロセッサ (I²C デバイス 1) にデータを転送し、将来のある時点でマイクロプロセッサが WISHBONE マスタにデータを返送できる。

図 9-9. I²C 回路



I²C 設計のヒント

1. I²C レジスタの定義とコマンドシーケンスについては、“TN1246, Using User Flash Memory and Hardened Control Functions in MachXO2 Devices Reference Guide (MachXO2 デバイスのユーザフラッシュメモリとハードマクロ制御機能の使用法リファレンスガイド)” の I²C 項を参照してください
2. EFB では、消費電力削減のためコンポーネントを動的にオフする場合に、WISHBONE クロックのソースになっているかどうかにかかわらず、MachXO2 の内部オシレータをイネーブルにする必要があります。唯一の例外として、I2C_1_CR[3:2]=11 と設定することにより SDA 遅延をオフにした後、内部オシレータをオフにできます。その後内部オシレータをディセーブルにしてユーザスレーブ、またはユーザマスタで動作できます
3. フラッシュメモリ (UFM/ コンフィグレーション) へのアクセスでは、I²C は JTAG ポート及びスレーブ SPI ポートよりも優先順位が低くなります。詳細については、29 ページの “フラッシュメモリ (UFM/ コンフィグレーション) のアクセス” を参照してください
4. プライマリ I²C ポートはフラッシュメモリへのアクセスとユーザ・ファンクションへのアクセスを、単一デザインで両方利用できるようにすることはできません。
5. Refresh コマンドの発行後、または PROGRAMN のトグル後にセカンダリセカンダリ I²C ポートをイネーブルにした場合は、I2C_STOP でステートマシンをリセットすることを推奨します。I2C STOP

- は、I2C_2_CMDR への 0x40 のレジスタの 1 回ライトで実行できます。その結果、SCK にブロック信号 STOP として短い Low パルスが発生します。遅延が発生することなく、通常の I²C 動作が開始されます
6. ラティスのウェブサイト (www.latticesemi.com/products/intellectualproperty/aboutreferencedesigns.cfm) には、次のような多数の I²C 参照デザインがあります。
 - a. RD1124 : I²C Slave Peripheral Using Embedded Function Block (組み込み機能ブロックを使用した I²C スレーブ周辺回路)
 - b. UG55 : MachXO2 Hardened I²C Master/Slave Demo (MachXO2 ハードマクロ I²C マスタ / スレーブ・デモ)
 7. I²C プロトコルの詳細については www.i2c-bus.org/ を参照してください
 8. UFM リード (0xCA) で正しいコマンド (0xCA 00 00 01) であることを確認
 9. コンフィグレーション・フラッシュ・リード (0x73) で正しいコマンド (0x73 00 00 01) であることを確認
 10. MachXO2 の入力バッファは汎用タイプのため 400MHz まで動作が可能な設計になっています。このため、低速の I2C クロックはノイズで誤動作しやすいので注意が必要です。入力波形の傾きが緩やかな場合、以下の方法で改善することができます：
 - ・ 2kΩ 程度の強い外部プルアップ抵抗を用いる
 - ・ ヒステリシス・オプションをオンにする
 - ・ グリッチフィルタを用いる

ハードマクロ化 SPI IP コア

SPI は幅広く使用されている 4 線式シリアルバスで、全二重動作でデバイス間の通信が行われます。MachXO2 EFB に内蔵される SPI コントローラは、SPI マスタ / スレーブまたは SPI スレーブとして構成できます。IP コアがマスタ / スレーブとして構成された場合は、スレーブ SPI インターフェイスを持つ最大 8 個の他のデバイスを制御できます。コアがスレーブとして構成された場合は、外部 SPI マスタデバイスとインターフェイスできます。SPI コアは、MachXO2 のコンフィグレーション・ロジックまたは他のユーザロジックとインターフェイスします。ハードマクロ化された SPI IP コアの機能とブロック図を、次に示します。

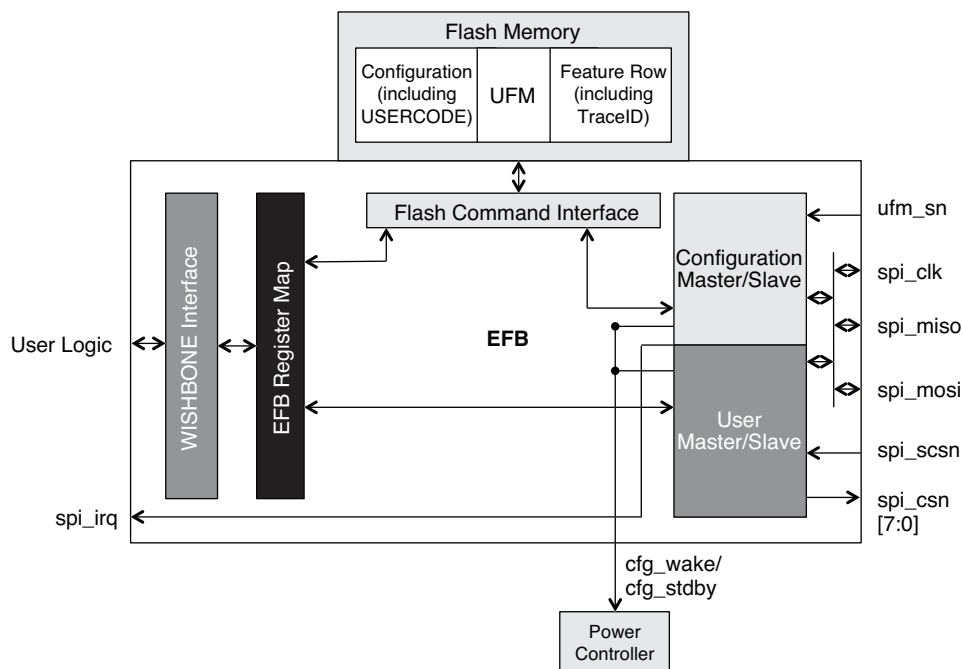
表 9-6. ハードマクロ化 SPI の機能

	コンフィグレーション SPI	ユーザ SPI
スレーブ SPI ポート	Yes	Yes
マスタ SPI ポート	No ¹	Yes
フラッシュメモリ (UFM/ コンフィグレーション) のアクセス	Yes	No
専用 I/O の使用の必要性	Yes	Yes ²
パワーコントローラのスタンバイモードからのウェイクアップ	Yes	Yes
パワーコントローラのスタンバイモードへの移行	No	Yes

1. SRAM のコンフィグレーション専用

2. 任意の GPIO を spi_csn[7:1] 及び spi_scsn に使用できる

図 9-10. SPI ブロック図



MachXO2 デバイスの SPI IP コアは次の機能をサポートしています。

- ・ 構成可能なマスタ及びスレーブモード
- ・ CPU 割り込み機能のあるモード故障エラーフラグ
- ・ ダブルバッファ・データレジスタによるスループットの向上
- ・ 極性と位相がプログラマブルなシリアルクロック
- ・ LSB ファーストまたは MSB ファーストのデータ転送
- ・ EFB WISHBONE スレーブ・インターフェイスによるカスタムロジックへのインターフェイス

マスタ / スレーブ SPI モードの場合

- ・ ユーザ SPI コントローラでは、8 個のマスタ・チップセレクト (spi_csn[7:0]) ポートを使用できます。そのため、スレーブ SPI インターフェイスを持つ最大 8 個の外部デバイスを制御できます。
- ・ 電源投入時のコンフィグレーション SPI では、外部 SPI フラッシュメモリから MachXO2 デバイスが起動するよう SPI ポートがイネーブルになっている場合、その SPI ポートはマスタ SPI コントローラとして機能し、特定の SPI フラッシュメモリを選択するためのマスタ・チップセレクトとして spi_csn[0] が使用されます。SPI ポートからの MachXO2 のプログラミングの詳細については、“TN1204, MachXO2 Programming and Configuration Usage Guide (MachXO2 プログラミングとコンフィグレーションのユーザガイド)” の SPI の項を参照してください。

スレーブ SPI モードの場合

- ・ ユーザ SPI コアには、スレーブ・チップセレクト (spi_scsn) ピンが 1 本あります。そのため、マスタ SPI インターフェイスを持つ外部デバイスからユーザ SPI を選択できます。ユーザロジックは、FPGA ロジック内の WISHBONE マスタから、EFB WISHBONE インターフェイスを通じてアクセスされます。
- ・ コンフィグレーション SPI には、スレーブ・チップセレクト (ufm_sn) ピンが 1 本あります。外部 SPI マスタは、チップセレクト入力のアサートすることで、MachXO2 のコンフィグレーション・ロジックにアクセスできます。外部 SPI マスタは、SN がアサートされた状態のプログラミングバス転送を使用して、

MachXO2 のコンフィグレーション・フラッシュ及びユーザフラッシュメモリをプログラムできます。

この使用法ガイドでは、ユーザ SPI のアクセスを中心に説明します。SPI ポートからの MachXO2 のプログラミングの詳細については、“TN1204, MachXO2 Programming and Configuration Usage Guide (MachXO2 プログラミングとコンフィグレーションのユーザガイド)” の SPI の項を参照してください。

スレーブ・コンフィグレーション SPI ポートは、パワーコントローラのスタンバイからのウェイクアップ、またはスタンバイへの移行に使用できます。スレーブ・ユーザ SPI ポートは、パワーコントローラのスタンバイモードからのウェイクアップだけに使用できます。パワーコントローラの詳細については、“TN1198, Power Estimation and Management for MachXO2 Devices (MachXO2 デバイスの電源見積りと管理)” を参照してください。SPI パワーコントローラの機能は、後述するように IPexpress から設定できます。レジスタ設定については、“TN1246, Reference Guide for Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (MachXO2 デバイスのユーザフラッシュメモリ及びハードマクロ制御機能の使用法リファレンスガイド)” に説明があります。

SPI インターフェイス信号

SPI インターフェイスではシリアル送信プロトコルが使用されます。データはシリアルに送信され（送信デバイスからシフトアウトされ）、受信デバイスにシフトインされてシリアルに受信されます。マスタデバイスは、チップセレクトをアサートして特定のスレーブデバイスを選択することで、スレーブデバイスがコマンド / データをシフトインし、データのシフトアウトで応答できるようにします。

表 9-7 に IP コアで生成される信号を示します。各信号には、使用方法と、デザインプロジェクトの中での接続方法に関する説明が記述されています。

表 9-7. SPI ～ IP 信号

信号名	固有のピン名	I/O	幅	記 述
spi_clk	MCLK/CCLK	双方向	1	SPI コアがマスタモード時、信号は出力（MCLK）で、SPI コアがスレーブモード時、信号は入力（CCLK） この信号は、事前に割り当てられた対応するピンに直接配線されること。各 MachXO2 デバイスの SPI 信号のパッドとピンの位置については、“MachXO2 Family Data Sheet (MachXO2 ファミリー・データシート)” のピンテーブルを参照
spi_miso	SPISO/SO	双方向	1	SPI コアがマスタモード時は入力（SPISO）。SPI コアがスレーブモード時は出力（SO） この信号は、事前に割り当てられた対応するピンに直接配線する。各 MachXO2 デバイスの SPI 信号のパッドとピンの位置については、“MachXO2 Family Data Sheet (MachXO2 ファミリー・データシート)” のピンテーブルを参照
spi_mosi	SISPI/SI	双方向	1	SPI コアがマスタモード時は出力（SISPI）で、SPI コアがスレーブモード時は入力（SI） この信号は、事前に割り当てられた対応するピンに直接配線する。各 MachXO2 デバイスの SPI 信号のパッドとピンの位置については、“MachXO2 Family Data Sheet (MachXO2 ファミリー・データシート)” のピンテーブルを参照
spi_csn[7:0]	CSSPIN	出力	8	マスタ・チップセレクト（アクティブ Low）。マスタ SPI モードの MachXO2 SPI コントローラを使用すると、最大 8 個の個別スレーブ SPI デバイスにアクセス可能 信号 spi_csn[0] は、事前に割り当てられた対応するピンに直接配線する。各 MachXO2 デバイスの SPI 信号のパッドとピンの位置については、“MachXO2 Family Data Sheet (MachXO2 ファミリー・データシート)” のテーブルを参照
spi_scsn	—	入力	1	ユーザスレーブ・チップセレクト（アクティブ Low）。外部 SPI マスタコントローラがこの信号をアサートし、SPI コントローラの送信データ / 受信データ・レジスタとの間でデータを転送。MachXO2 デバイスの任意の GPIO で配線できる

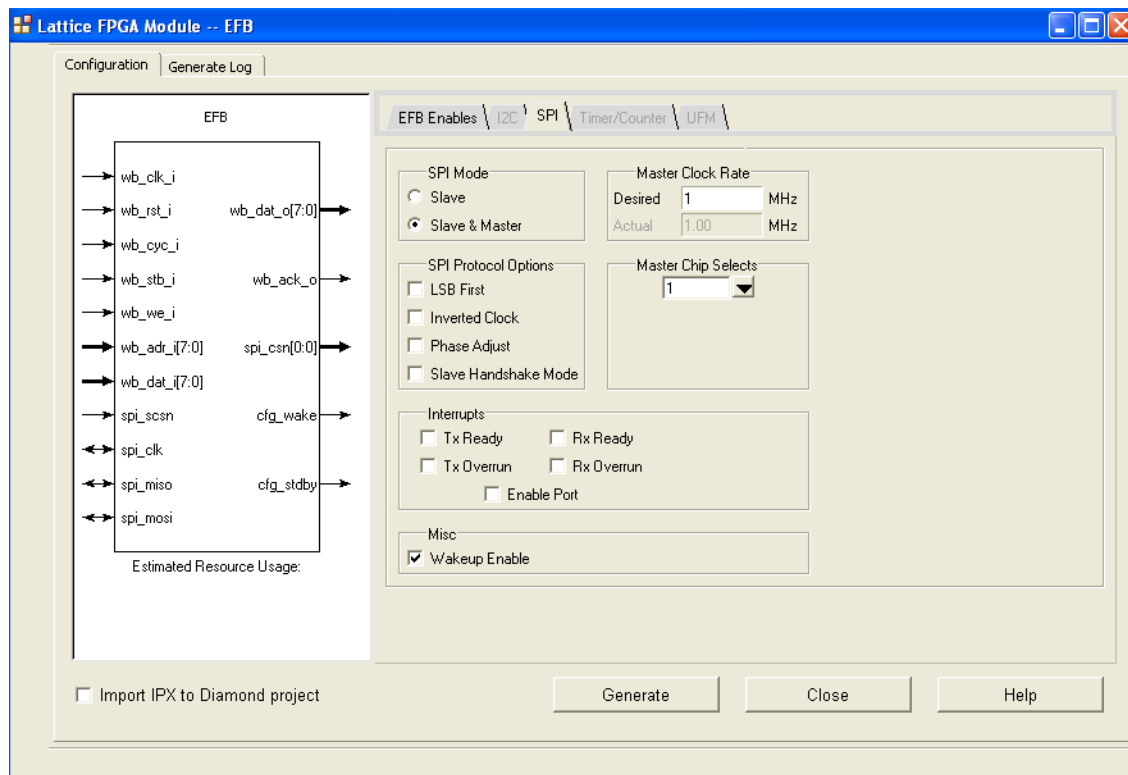
表 9-7. SPI ~ IP 信号 (Continued)

信号名	固有のピン名	I/O	幅	記 述
ufm_sn	SN	入力	1	コンフィグレーション・ロジックのチップセレクト（アクティブ Low）は、フラッシュメモリの UFM 及びコンフィグレーション・セクタの選択専用 この信号は事前に割り当てられた対応するピンに直接配線されること。各 MachXO2 デバイスの SPI 信号のパッドとピンの位置については、“MachXO2 Family Data Sheet（MachXO2 ファミリー・データシート）” のピンテーブルを参照 SN は、EFB プリミティブに 'ufm_sn' が存在するかどうかにかかわらず、SPI コアがインスタンス化されると必ずアクティブピンとなる。そのため、SN はユーザ I/O として回復できない。外部マスタ SPI バスに接続しない場合は、SN を外部的に High に接続し、内部の弱いプルアップを強化できる SN は、ブランクまたは消去済みデバイスの内部でもアクティブとなる
spi_irq	—	出力	1	SPI コアの割り込み要求出力信号。この信号の目的は、WISHBONE マスタコントローラ（マイクロコントローラまたはステートマシン）に接続すること。特定の条件に適合するとアサートされる。このような条件は、SPI レジスタ設定を使用して制御される
cfg_wake	—	出力	1	ウェイクアップ信号 ~ MachXO2 デバイスのパワーコントローラのみ に接続この信号がイネーブルとなるのは、EFB GUI の SPI タブで Wakeup Enable（ウェイクアップ・イネーブル）機能がイネーブルとなっている場合のみ
cfg_stdby	—	出力	1	スタンバイ信号 ~ MachXO2 デバイスのパワーコントローラのみ に接続この信号がイネーブルとなるのは、EFB GUI の SPI タブで Wakeup Enable（ウェイクアップ・イネーブル）機能がイネーブルとなっている場合のみ

IPexpress による SPI コアの構成

IPexpress は、SPI コントローラの構成と、デザインに組み込むための Verilog または VHDL ソースコードの生成に使用されます。EFB GUI で SPI タブを選択すると、SPI コアの構成可能な設定が表示されます。図 9-11 に SPI コントローラの構成例を示します。

図 9-11. IPexpress による EFB モジュールの SPI 機能の構成



SPI モード (SPI Mode)

本オプションでは、ユーザが SPI コアの内部モードを "Slave" または "Slave & Master" モードに設定できます。"Slave & Master" を選択すると、マスタ・クロックレートやマスタ・チップセレクトなどの SPI マスタ設定がイネーブルになります。レジスタ SPICR2 の MSTR ビットを変更すると、このオプションを動的に更新できます。

マスタ・クロックレート (SPI Master Clock Rate)

Desired Frequency (希望する周波数)：SPI マスタとして構成された EFB SPI コントローラは、バス上の SPI スレーブデバイスへの出力クロックを出力します。出力クロックは、WISHBONE クロック周波数を 2 のべき乗値で分周されます。SPI マスタはマスタ・クロックレートを使用して、全ての SPI バストランザクション及び内部動作のタイミングを合わせます。MachXO2 の SPI マスタ・インターフェイスは、最高 45 MHz まで動作できます。入力、ダイアログの EFB Enables タブにある WISHBONE Clock Frequency で行います。

除数は、FPGA がユーザモードのときに変更できます。SPIBR レジスタの分周値を更新すると、SPI コントローラがリセットされ、新しい出力クロック周波数を使用するようになります。

Actual Frequency (実際の周波数)：入力 WISHBONE クロックを、ユーザが要求した周波数どおりに分周できない場合があります。その場合に実際の周波数値は、この読み取り専用フィールドに返されます。希望する SPI クロックと WISHBONE クロックの両方のフィールドに有効な値が指定され、どちらかが更新されると、このフィールドには小数点以下 2 位までに丸められた値が返されます。

SPI プロトコル・オプション (SPI Protocol Options)

LSB First (LSB 先出し)：この設定では、データバイトのシリアルシフトの順序が指定されます。データの順序 (MSB と LSB のどちらが先か) は SPI コア内部でプログラムできます。レジスタ SPICR2 の LSBF ビットを変更すると、このオプションを動的に更新できます。

Inverted Clock (反転クロック): サンプル及び出力データに使用されるクロック極性の反転は、SPI コアに対してプログラムできます。選択すると、有効クロックエッジが立ち上がりから立ち下がりになります。レジスタ SPICR2 の CPOL ビットにアクセスすることで、このオプションを動的に更新できます。

Phase Adjust (位相調整): 特定の要件を持つ SPI デバイスに対しては、別のクロックとデータの設定をします。このオプションでは、ユーザがアプリケーションに合った位相を指定できます。レジスタ SPICR2 の CPHA ビットにアクセスすることで、このオプションを動的に更新できます。

Slave Handshake Mode (スレーブ・ハンドシェイクモード): SPI プロトコルに対するラティス固有の拡張機能をイネーブルします。内部サポート回路 (WISHBONE ホストなど) が最初のデータが必要な時間内に応答できない場合、及びスレーブ・リードアウト・データを定期的に速い SPI クロックレートで 사용할 ことができるようにする場合に使用します。レジスタ SPICR2 の SDBRE ビットにアクセスすることで、このオプションを動的に更新できます。

マスタ・チップセレクト (Master Chip Selects)

SPI コントローラはマスタ動作として、最大 8 本の独立したチップセレクトを生成できます。マスタからアクセスされる SPI デバイスごとに、専用のチップセレクトが必要です。レジスタ SPICSR を変更することで、このオプションを動的に更新できます。

SPI コントローラの割り込み (SPI Controller Interrupts)

TX Ready: SPI 送信データレジスタ (SPITXDR) が空であることを示す割り込みで、割り込みビットはレジスタ SPIIRQ の IRQTRDY です。イネーブルすると、TRDY がアサートされたことを示します。本ビットに '1' をライトすると、割り込みがクリアされます。レジスタ SPICSR のビット IRQTRDYEN を変更することで、このオプションを動的に変更できます。

RX Ready: 受信データレジスタ (SPIRXDR) に有効な受信データが格納されていることを示す割り込みで、割り込みはレジスタ SPIIRQ のビット IRQRRDY です。イネーブルすると、RRDY がアサートされたことを示します。本ビットに '1' をライトすると、割り込みがクリアされます。レジスタ SPICSR のビット IRQRRDYEN を変更することで、このオプションを動的に変更できます。

TX Overrun: SPI マスタのときに、スレーブ SPI チップセレクト (SPI_SCSN) が Low にドライブされたことを示す割り込みで、割り込みはレジスタ SPIIRQ のビット IRQMDF です。イネーブルすると、MDF (Mode Fault) がアサートされたことを示します。本ビットに '1' をライトすると、割り込みがクリアされます。レジスタ SPICSR のビット IRQMDFEN を変更することで、このオプションを動的に変更できます。

RX Overrun: SPIRXDR が、新しいデータをその前のデータよりも先に受信したことを示す割り込みで、割り込みはレジスタ SPIIRQ のビット IRQROE です。イネーブルすると、ROE がアサートされたことを示します。本ビットに '1' をライトすると、割り込みがクリアされます。レジスタ SPICSR のビット IRQROEEN を変更することで、このオプションを動的に変更できます。

Enable Port (割り込み): SPI コア信号の割り込み要求出力信号 (spi_irq) をイネーブルします。この信号の目的は、WISHBONE マスタコントローラ (マイクロコントローラまたはステートマシン) に接続し、特定の条件に適合した場合に割り込みを要求することです。

ウェイクアップ・イネーブル (Wake-up Enable)

ユーザスレーブ SPI チップセレクト (spi_csn[0]) が Low にドライブされたときに、SPI コアがパワーコントローラにウェイクアップ信号を送出し、デバイスがスタンバイモードから起動することをイネーブルします。レジスタ SPICR1 のビット WKUPEN_USER を変更することで、このオプションを動的に更新できます。

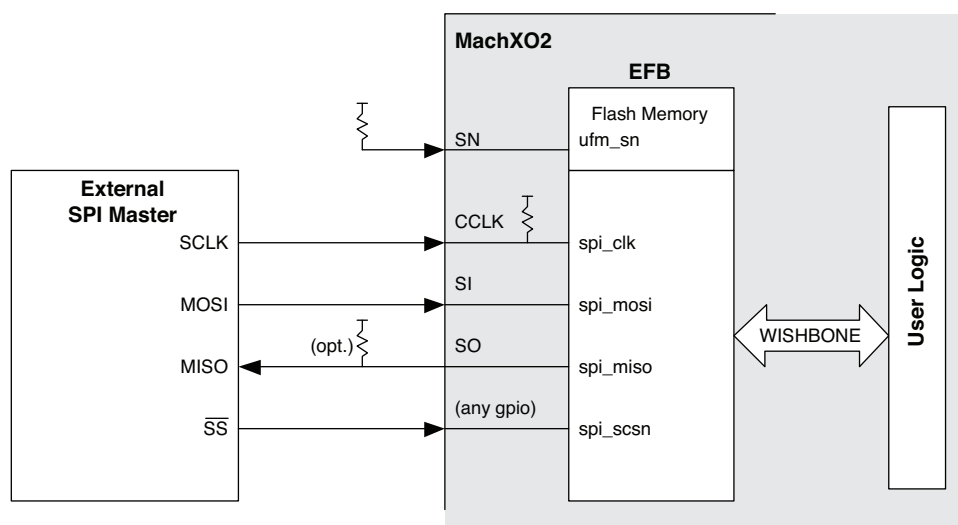
MachXO2 SPI のユースケース

以下で説明する SPI の使用例は、以下の図を参照してください。

1. スレーブ MachXO2 のユーザ SPI にアクセスする外部マスタ SPI デバイス

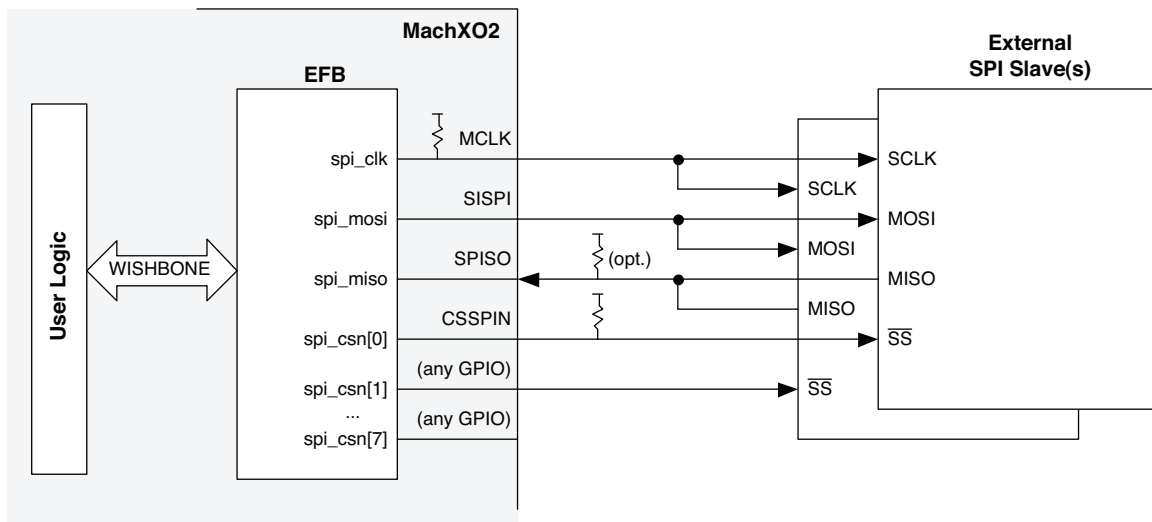
- a. 外部マスタ SPI は、専用の SI、SO、CCLK ピンを使用して MachXO2 に接続され、spi_scsn には任意の汎用 I/O が使用される。EFB の SPI モードはスレーブオンリーに設定される
- b. WISHBONE マスタコントローラが、MachXO2 の汎用ロジックアレイに実装される。マスタコントローラは、SPI ステータスレジスタをポーリングするか、SPI コントローラで生成された割り込みに応答することで、送信または受信データがあるかどうかをモニタする
 - i. SPI レジスタの定義とコマンドシーケンスについては、“TN1246, Reference Guide for Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (MachXO2 デバイスのユーザフラッシュメモリとハードマクロ制御機能の使用法リファレンスガイド)” の SPI 項を参照
- c. コンフィグレーション・ロジックを選択する SN が High にプルアップされているため、外部 SPI マスタは MachXO2 のコンフィグレーション・ロジックにアクセスできない

図 9-12. スレーブ MachXO2 のユーザ SPI にアクセスする外部マスタ SPI デバイス



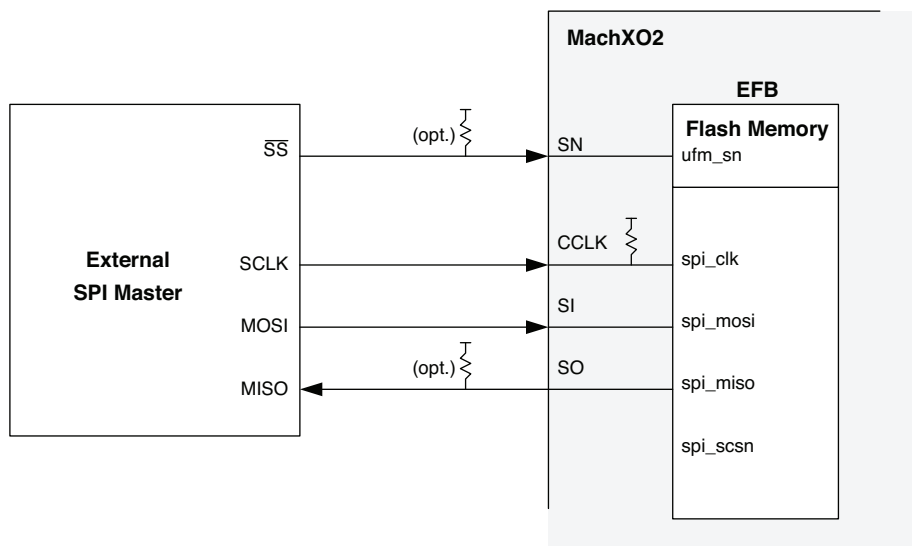
2. 1 個または複数のスレーブ SPI デバイスにアクセスする MachXO2 のユーザ SPI マスタ
 - a. MachXO2 の SPI マスタは、専用 SPI ポートピンを使用して外部 SPI スレーブデバイスに接続され、チップセレクトは次のように構成される
 - i. MachXO2 の SPI マスタ・チップセレクト spi_scsn[0] には専用の CSSPIN が使用され、外部スレーブ・チップセレクトに接続される
 - ii. MachXO2 の SPI マスタ・チップセレクト spi_scsn[1] には任意の I/O が使用され、別の外部スレーブ・チップセレクトに接続される
 - iii. spi_scsn[7:0] を使用して、最大 8 個の外部スレーブ SPI を接続できる
 - b. WISHBONE マスタコントローラは、MachXO2 の一般ロジックに実装され、スレーブ SPI デバイスへの転送を制御する。ポーリング方式または SPI コントローラ割り込みを使用して、データの送受信を管理する
 - i. SPI レジスタの定義とコマンドシーケンスについては、“TN1246, Reference Guide for Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (MachXO2 デバイスのユーザフラッシュメモリとハードマクロ制御機能の使用法リファレンスガイド)” の SPI 項を参照

図 9-13. 1 個または複数のスレーブ SPI デバイスにアクセスする MachXO2 のユーザ SPI マスタ



3. MachXO2 のコンフィグレーション・ロジックにアクセスする外部マスタ SPI デバイス
 - a. 外部 SPI マスタは、MachXO2 の専用スレーブ・コンフィグレーション SPI ポートピンに接続され、外部 SPI マスタのチップセレクトは、MachXO2 のコンフィグレーション論理ブロックをイネーブルする SN 入力を制御する。外部マスタは、コンフィグレーション論理ブロックにコマンドを送信し、コンフィグレーション・フラッシュ及び UFM とのインターフェイスを可能にする
 - i. SPI による MachXO2 デバイスのフラッシュメモリ (UFM/ コンフィグレーション) のアクセスの詳細については、本書で後述する
 - ii. SPI ポートからの MachXO2 のプログラミングの詳細については、“TN1204, MachXO2 Programming and Configuration Usage Guide (MachXO2 プログラミングとコンフィグレーションのユーザガイド)” の SPI の項を参照

図 9-14. MachXO2 のコンフィグレーション・ロジックにアクセスする外部マスタ SPI デバイス



SPI 設計のヒント

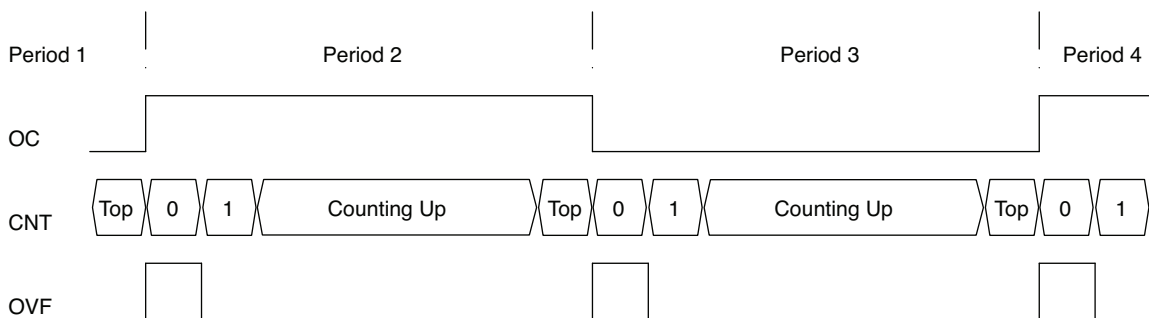
1. SPI レジスタの定義とコマンドシーケンスについては、“TN1246, Reference Guide for Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (MachXO2 デバイスのユーザフラッシュメモリとハードマクロ制御機能の使用法リファレンスガイド)” の SPI の項を参照してください
2. EFB では、消費電力削減のためコンポーネントを動的にオフする場合、WISHBONE クロックのソースになっているかどうかにかかわらず、MachXO2 の内部オシレータをイネーブルにする必要があります。例外はユーザスレーブ、またはユーザマスタ動作です
3. SPI バスは双方向性です。送信した全てのバイトに対するバイトが受信されます。ROE (Receiver Overrun Error) 防止のため、RX データは全て破棄してください
4. SN は、EFB プリミティブに 'ufm_sn' が存在するかどうかにかかわらず、SPI コアがインスタンス化されると必ずアクティブピンになります。そのため、SN はユーザ I/O として回復できません。外部マスタ SPI バスに接続しない場合は、SN を外部的に High に接続し、内部の弱いプルアップを強化する必要があります
5. ラティスのウェブサイト (www.latticesemi.com/products/intellectualproperty/aboutreferencedesigns.cfm) には、次のような多数の SPI 参照デザインがあります
 - a. RD1125: SPI Slave Peripheral Using Embedded Function Block (組み込み機能ブロックを使用した SPI スレーブ周辺回路)
 - b. UG56: MachXO2 Hardened SPI Master/Slave Demo (MachXO2 ハードマクロ SPI マスタ / スレーブ・デモ)

タイマ / カウンタ

MachXO2 EFB にはタイマ / カウンタ機能が内蔵されています。このタイマ / カウンタは汎用 16 ビットタイマ / アップダウン・カウンタモジュールで、独立した出力比較ユニットがあり、PWM (Pulse Width Modulation) に対応しています。このタイマ / カウンタは次の機能に対応しています。

- ・ 独自の 4 つの動作モード
 - ウォッチドッグタイマ
 - コンペアマッチ (比較一致時) タイマクリア
 - 高速 PWM
 - 位相及び周波数補正 PWM
- ・ プログラマブルなクロック入力ソース
- ・ プログラマブルな入力クロックのプリスケール
- ・ FPGA ファブリックへの割り込み出力
- ・ オーバフロー、出力比較一致、及び入力キャプチャという 3 つの独立した割り込みソース
- ・ 自動リロード
- ・ 入力キャプチャユニットでタイムスタンプ設定に対応
- ・ 出力で波形生成
- ・ PWM 期間が可変でグリッチのない PWM 波形生成
- ・ 制御及びステータスレジスタへの内部 WISHBONE バスアクセス
- ・ プリロード制御レジスタと直接リセット入力付きのスタンドアロンモード

図 9-16. タイマ / カウンタの出力波形



ウォッチドッグタイマ (Watchdog Timer) モード

ウォッチドッグタイマは、システム動作の監視用に使用され、システムのマイクロコントローラまたは組み込みステートマシンが動作しなくなった場合に、リセットまたは割り込みを発生します。シナリオの 1 つとして、マイクロコントローラが処理を開始する前に、ウォッチドッグタイマを 0x0000 にリセットすることがあります。マイクロコントローラは、タイマが最終カウントに達する前に、処理を完了してウォッチドッグタイマをリセットする必要があります。マイクロプロセッサが余裕をもってタイマをクリアしなかった場合、ウォッチドッグタイマはストローブ信号時間切れをアサートします。システムは "時間超過" ストローブを使用して、システムを正常にリカバリします。

ウォッチドッグタイマのもう 1 つの使用法としては、システムモジュールを定期的に電源オフし、消費電力を節約するという方法があります。MachXO2 のオンチップ・パワーコントローラとの連動に使用することもできます。

ウォッチドッグタイマ・モードのタイマ / カウンタで最もよく使用されるポートは、クロック、リセット、及び割り込みです。オプションとして、WISHBONE インターフェイスを使用して TCICR レジスタからタイムスタンプを読み取り、カウンタのトップ値を更新することもできます。

高速 PWM (Fast PWM) モード

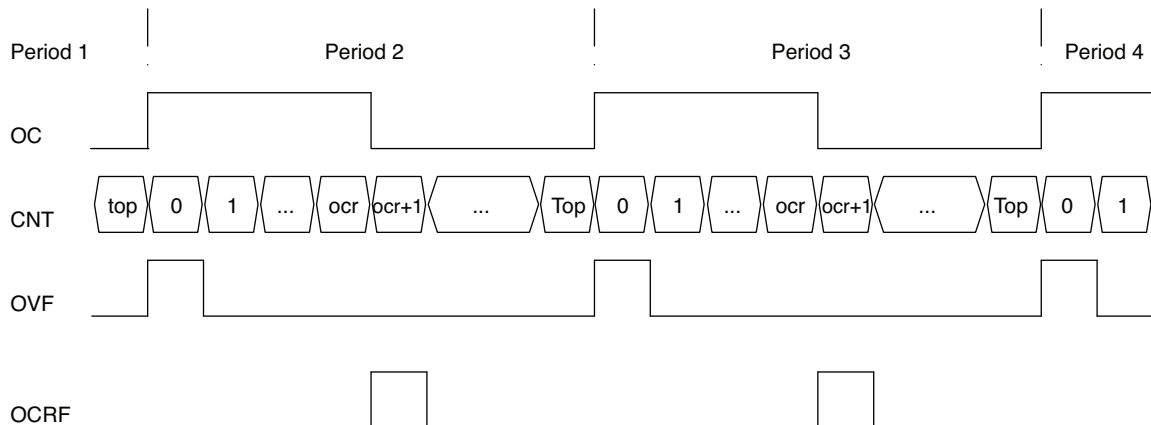
PWM (Pulse-Width Modulation) は、アナログ回路をデジタル制御するためによく使用される技術です。PWM では、パルス幅が変調された方形波パルスを使用して、波形の平均値を変化させます。設計者は、TCTOP レジスタに 16 ビット値をロードして波形の周期とデューティサイクルを変化させることで、カウンタのトップ値を定義し、カウンタ出力との比較値を TCOCR レジスタから得ることができます。

タイマ / カウンタの出力は、カウンタ値が TCTOP レジスタにロードされたトップ値と一致するとクリアされます。TCOCR レジスタにロードされた比較値とカウンタの値が一致すると、出力がセットされます。クリア / セット機能は反転可能です。その場合は、カウンタ値がトップ値と一致するとタイマ / カウンタの出力がセットされ、カウンタ値が比較値と一致するとクリアされます。

割り込みラインは、OVF (Overflow Flag、オーバーフローフラグ) 及び OCRF (Output Compare Flag、出力比較フラグ) として使用できます。

図 9-8 に PWM 波形の生成を示します。タイマ / カウンタの出力は、カウンタがトップ値と一致するとセットされ、カウンタが比較値と一致するとクリアされるよう構成されます。

表 9-8. PWM 波形の生成



位相及び周波数補正 PWM (Phase and Frequency Correct PWM) モード

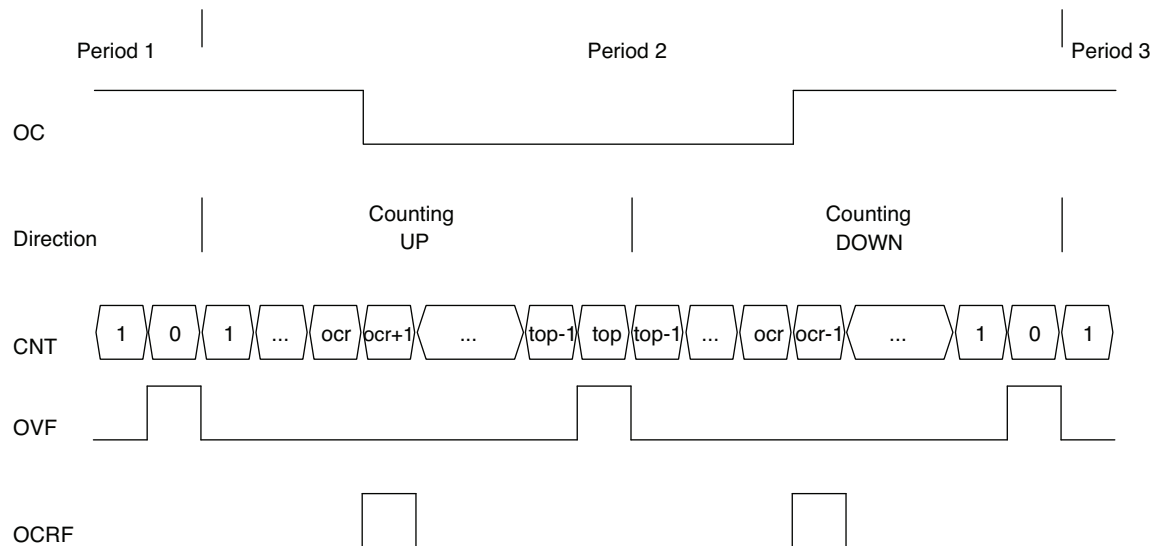
位相及び周波数補正 PWM モードでは、カウンタがトップ値（トップ値 -1）まで増加すると、カウント方向がアップからダウンに変化します。カウンタが 0x0001 から 0x0000 に減少すると、次の動作が行われます。

1. TCTOP が、TCTOPSET レジスタにロードされた値で更新される
2. TCOCR が、TCOCRSET レジスタにロードされた値で更新される
3. 1 クロックサイクルの間、オーバフロー TCSR[OVF] がアサートされる

タイマ / カウンタの出力が更新されるのは、カウンタ値が TCOCR レジスタの比較値と一致した場合だけです。これは、1 周期に 2 回発生します。最初の一致が発生するのはカウンタがカウントアップ時で、2 回目の一致が発生するのはカウンタがカウントダウン時です。両方の一致が発生すると、出力比較フラグ TCSR[OCRF] がアサートされます。タイマ / カウンタの出力は最初の比較一致でセットされ、2 回目の比較一致でクリアされます。セットとクリアの順序は逆にできます。

このモードでは、生成される波形の周波数（トップ値がベース）と位相（比較値がベース）をユーザが調整できます。

図 9-17. 位相及び周波数補正 PWM 波形の生成



タイマ / カウンタの IP 信号

表 9-9 に IP で生成される信号を示します。各信号には、使用方法とデザインプロジェクト中での接続方法に関する説明が記述されています。

表 9-9. タイマ / カウンタ ~ IP 信号

信号名	I/O	幅	記 述
tc_clk	入力	1	タイマ / カウンタ入力信号。オンチップ・オシレータに接続可能。クロック信号の上限は 133 MHz
tc_rstn	入力	1	アクティブ Low のリセット信号。16 ビットカウンタをリセットする
tc_ic	入力	1	アクティブ High の入力キャプチャ・トリガイベント。PWM 以外のモードの WISHBONE インターフェイスで使用可能。イネーブルすると、この信号の立ち上がりエッジが検出され、カウンタ値（TCCNT レジスタ）のキャプチャが同期されて、その値を TCICR レジスタにロードすることで、WISHBONE インターフェイスから値にアクセスできるようになる。一般的な用途は、カウンタによるタイムスタンプ動作の実行
tc_int	出力	1	オーバフロー、出力比較一致、入力キャプチャなど、特定のイベントの発生を示す割り込み信号
tc_oc	出力	1	タイマ / カウンタの出力信号

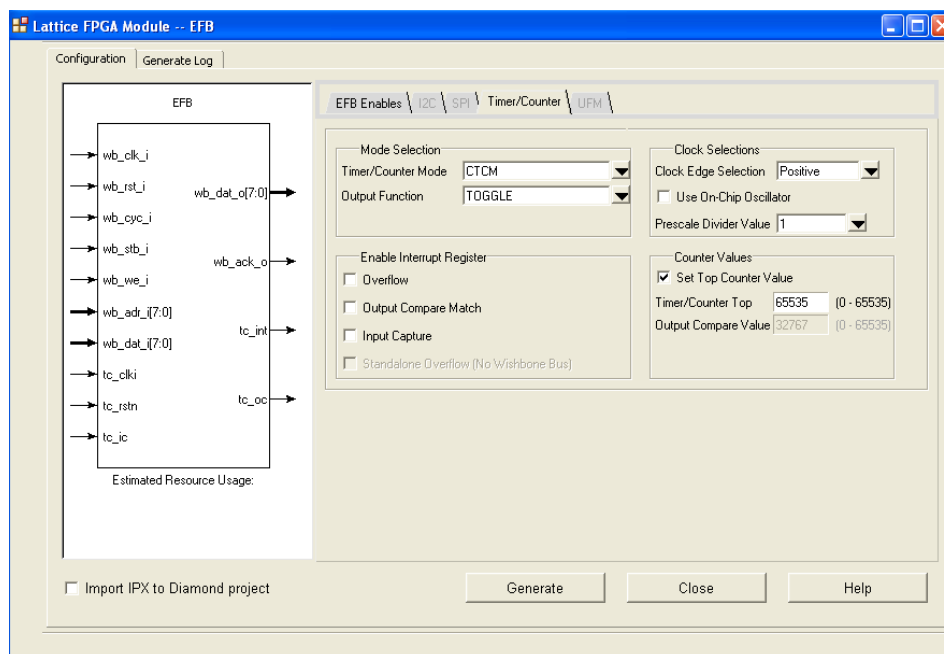
タイマ / カウンタの構成

タイマ / カウンタの構成には IPexpress を使用します。EFB GUI で Timer/Counter（タイマ / カウンタ）タブを選択すると、タイマ / カウンタ・コアの構成可能な設定が表示されます。

タイマ / カウンタは、WISHBONE インターフェイスの有無にかかわらず使用できます。WISHBONE インターフェイスがない場合の用途については、設計者が IPexpress の EFB GUI で値を選択 / 入力します。値は、プログラマブルなビットストリームを使用してタイマ / カウンタでプログラムします。タイマ / カウンタと WISHBONE インターフェイスを組み合わせると、ユーザが WISHBONE インターフェイスからレジスタの内容を動的に更新できます。メイン EFB GUI である EFB Enables タブでは、タイマ / カウンタを WISHBONE と組み合わせるかどうかを、スクリーンショットで選択できます。

図 9-18 に特定のデザイン用に構成されたタイマ / カウンタの例を示します。

図 9-18. タイマ/カウンタの構成



タイマ/カウンタ・モード (Timer/Counter Mode)

このオプションでは、ユーザが次の 4 つの動作モードの 1 つを選択できます。

- ・ CTCM ~ コンペアマッチ (比較一致時) タイマクリア
- ・ WATCHDOG ~ ウォッチドッグタイマ
- ・ FASTPWM ~ 高速 PWM
- ・ PFCPWM ~ 位相及び周波数補正 PWM

レジスタ TCCR1 のビット TCM[1:0] を変更することで、このオプションを動的に更新できます。

出力機能 (Output Function)

タイマ/カウンタ IP の出力信号 (tc_oc) の機能を以下から選択できます。

- ・ STATIC ~ タイマ/カウンタの出力が Low に固定されます
- ・ TOGGLE ~ タイマ/カウンタの出力が、Timer/Counter Mode で定義された条件に基づいてトグルします
- ・ WAV_GENERATE ~ Timer/Counter Mode で定義された条件に基づいたセット / クリアによって、波形が生成されます
- ・ INV_WAV_GENERATE ~ 波形が反転されます

レジスタ TCCR1 のビット OCM[1:0] を変更することで、このオプションを動的に更新できます。

クロックエッジ選択 (Clock Edge Selection)

入力クロックソースのエッジ (正または負) を選択し、オンチップ・オシレータの使用をイネーブルします。次の選択肢があります。

- ・ PCLOCK ~ ユーザロジックからのクロックの立ち上がりエッジ
- ・ POSC ~ 内部オシレータからのクロックの立ち上がりエッジ
- ・ NCLOCK ~ ユーザロジックからのクロックの立ち下がりエッジ

- ・ NOSC ~ 内部オシレータからのクロックの立ち下がりエッジ

レジスタ TCCR0 のビット CLKSEL 及び CLKEDGE を変更することで、このオプションを動的に更新できます。

プリスケール分周値 (Pre-scale Divider Value)

16 ビットカウンタに到達する前に入力クロックを分周できるよう、プリスケール分周値 (0、1、8、64、256、1024) が用意されています。レジスタ TCCR1 のビット PRESCALE[2:0] を変更することで、このオプションを動的に更新できます。

タイマ / カウンタのトップ値 (Timer/Counter Top)

カウンタのトップ値を選択できます。レジスタ TCTOPSET1 及び TCTOPSET0 のビット TCTOPSET を変更することで、このオプションを動的に更新できます。

出力比較値 (Output Compare Value)

カウンタの出力比較値を選択できます。レジスタ TCOCRSET1 及び TCOCRSET0 のビット TCOCRSET を変更することで、このオプションを動的に更新できます。

割り込みレジスタのイネーブル (Enable Interrupt Registers)

- ・ Overflow ~ カウンタが TCTOP0/1 レジスタの値と一致したことを示す割り込みで、割り込みはレジスタ TCIRQ のビット IRQOVF です。イネーブルすると、OVF がアサートされたことを示します。本ビットに '1' をライトすると、割り込みがクリアされます。レジスタ TCIRQEN のビット IRQOVFEN を変更することで、このオプションを動的に更新できます。
- ・ Output Compare Match ~ カウンタが TCOCR0/1 レジスタの値と一致したことを示す割り込みで、割り込みはレジスタ TCIRQ のビット IRQOCRF です。イネーブルすると、OCRF がアサートされたことを示します。本ビットに '1' をライトすると、割り込みがクリアされます。レジスタ TCIRQEN の IRQOCRFEN ビットを変更することで、このオプションを動的に更新できます。
- ・ Input Capture ~ ユーザが TC_IC 入力信号をアサートしたことを示す割り込みで、割り込みはレジスタ TCIRQ のビット IRQICRF です。イネーブルすると、ICRF がアサートされたことを示します。本ビットに '1' をライトすると、割り込みがクリアされます。レジスタ TCIRQEN の IRQICRFEN ビットを変更することで、このオプションを動的に更新できます。
- ・ Standalone Overflow ~ WISHBONE インターフェイスがない場合に使用され、使用可能な唯一の割り込み要求となります。

MachXO2 Timer/Counter タイマ / カウンタのユースケース

1. 割り込みのある基本的カウンタ
 - a. 静的または動的動作にタイマ / カウンタを構成する。動的動作では WISHBONE バスインターフェイスがイネーブルとなり、WISBONE マスタがタイマ / カウンタを制御できる。
 - b. Timer/Counter タブで
 - i. モード選択制御を使用して、Timer/Counter Mode を構成
 1. CTCM (コンペアマッチ・タイマクリア) モードを選択
 2. 出力機能 (Output Function) を選択
 - a. 0 を保持 (STATIC)
 - b. トグル (TOGGLE)
 - ii. クロック選択 (Clock Selections) を更新
 1. タイマ / カウンタが反応するクロックエッジを選択
 - a. 正 (PCLOCK) または負 (NCLOCK) のエッジ
 2. クロックのプリスケール分周を選択
 - iii. カウンタ値を構成
 1. トップカウンタ値をイネーブル

2. タイマ / カウンタのトップ値を選択
 - iv. オプションの割り込みをイネーブル
2. ウォッチドッグタイマ
 - a. 静的または動的動作にタイマ / カウンタを構成する。動的動作では WISHBONE バスインターフェイスがイネーブルとなり、WISBONE マスタがタイマ / カウンタを制御できる。
 - b. Timer/Counter タブで
 - i. モード選択制御を使用して Timer/Counter Mode を構成
 1. WATCHDOG モードを選択
 2. 出力機能 (Output Function) を選択
 - a. Hold static 0 を保持 (STATIC)
 - ii. クロック選択 (Clock Selections) を更新
 1. タイマ / カウンタが反応するクロックエッジを選択
 - a. 正 (PCLOCK) または負 (NCLOCK) のエッジ
 2. クロックのプリスケール分周を選択
 - iii. カウンタ値を構成
 1. トップカウンタ値をイネーブル
 2. タイマ / カウンタのトップ値を選択
 3. 出力比較値 (Output Compare Value) を選択
 - iv. 割り込み (TC_INT) をイネーブル
 1. 出力比較一致 (Output Compare Match) を選択
 2. 入力キャプチャ (Input Capture) を選択
3. デューティサイクルと周期が可変の PWM 出力
 - a. 静的または動的動作にタイマ / カウンタを構成する。動的動作では WISHBONE バスインターフェイスがイネーブルとなり、WISBONE マスタがタイマ / カウンタを制御できる
 - b. Timer Counter タブで
 - i. モード選択制御を使用して Timer/Counter Mode を構成
 1. FASTPWM モードを選択
 2. Output Function を選択
 - a. 0 を保持 (STATIC)
 - b. PWM (WAVE_GENERATOR)
 - c. 補完的 (Complement) PWM (INV_WAVE_GENERATOR)
 - ii. Clock Selections を更新
 1. タイマ / カウンタが反応するクロックエッジを選択
 - a. 正 (PCLOCK) または負 (NCLOCK) のエッジ
 2. クロックのプリスケール分周を選択
 - iii. PWM 値を構成
 1. PWM の周期を選択 (Timer Counter Top)
 2. PWM のデューティサイクルを選択 (Output Compare Value)
 - a. デューティサイクルの計算式 $(\text{Output Compare Value}) / (\text{Timer Counter Top})$:
 $[(\text{Timer Counter Top}) - (\text{Output Compare Value})] / (\text{Timer Counter Top})$
4. デューティ 50:50 で位相と周期が可変の PWM
 - a. 静的または動的動作にタイマ / カウンタを構成します。動的動作では WISHBONE バスインターフェイスがイネーブルとなり、WISBONE マスタがタイマ / カウンタを制御できます。
 - b. Timer Counter タブで

- i. モード選択制御を使用して、Timer/Counter Mode を構成
 - 1. FASTPWM を選択
 - 2. Output Function を選択
 - a. 0 を保持 (STATIC)
 - b. PWM (WAVE_GENERATOR)
 - c. 補完的 (Complement) PWM (INV_WAVE_GENERATOR)
- ii. Clock Selections を更新
 - 1. タイマ / カウンタが反応するクロックエッジを選択
 - a. 正 (PCLOCK) または負 (NCLOCK) のエッジ
 - 2. クロックのプリスケール分周を選択
- iii. PWM 値を構成
 - 1. PWM 周期を選択 (Timer Counter Top)
 - 2. 位相調整 (Phase Adjustment) を選択 (Output Compare Value)
 - a. 位相調整の計算式 : $(360 \text{ degrees}) * (\text{Output Compare Value}) / (\text{Timer Counter Top})$

タイマ / カウンタ設計のヒント

- 1. タイマ / カウンタ・レジスタの定義とコマンドシーケンスについては、“TN1246, Reference Guide for Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (MachXO2 デバイスのユーザフラッシュメモリとハードマクロ制御機能の使用法リファレンスガイド)” のタイマ / カウンタの項を参照してください。
- 2. EFB では、消費電力削減のためコンポーネントを動的にオフする場合に、WISHBONE クロックのソースになっているかどうかにかかわらず、MachXO2 の内部オシレータをイネーブルにする必要があります。例外はユーザクロックが選択されている場合です。

フラッシュメモリ (UFM/ コンフィグレーション) のアクセス

JTAG、SPI、I²C、または WISHBONE インターフェイスを使用して、フラッシュメモリのコンフィグレーション・ロジック・インターフェイスにアクセスできます。MachXO2 のフラッシュメモリは、次の 3 つのセクタで構成されます。

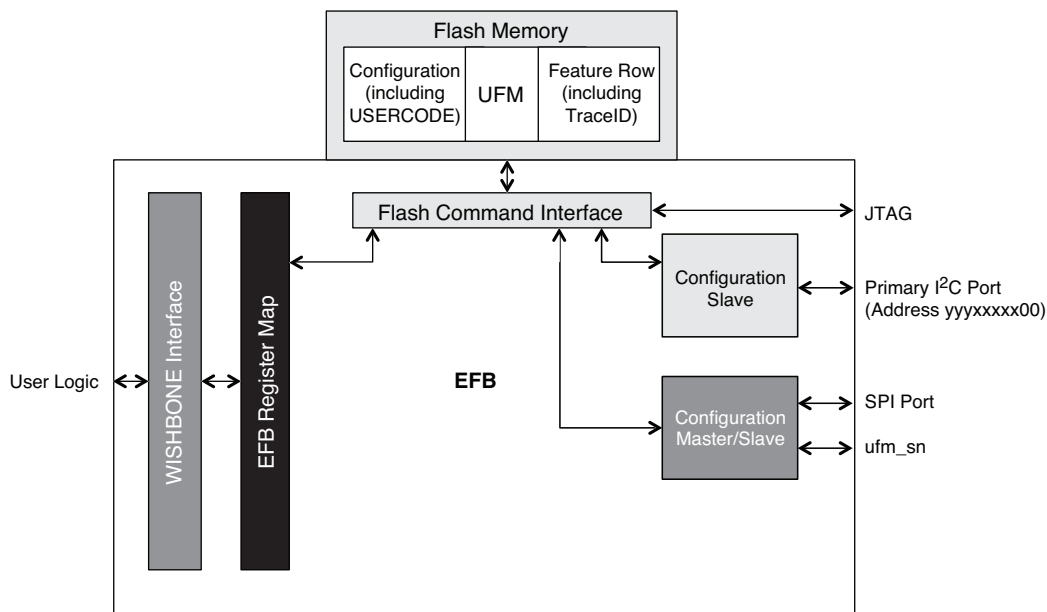
- ・ ユーザフラッシュメモリ (UFM)
- ・ コンフィグレーション
- ・ フィーチャ行

フラッシュメモリへのアクセス用ポート、及びブロック図を表 9-10 に示します。。

表 9-10. フラッシュメモリ (UFM/ コンフィグレーション) のアクセス

	UFM	コンフィグレーション・フラッシュメモリ
JTAG	Yes	Yes
スレーブ SPI ポート、チップセレクト (ufm_sn) あり	Yes	Yes
スレーブ SPI ポート、チップセレクト (spi_scsn) あり	No	No
プライマリ・スレーブ I ² C ポート・アドレス (yyyyxxxx00)	Yes	Yes
プライマリ・スレーブ I ² C ポート・アドレス (yyyyxxxx01)	No	No
セカンダリ・スレーブ I ² C ポート・アドレス (yyyyxxxx10)	No	No
WISHBONE	Yes	Yes

図 9-19. フラッシュメモリ (UFM/ コンフィグレーション) のブロック図



フラッシュメモリ (UFM/ コンフィグレーション) のアクセスポート

コンフィグレーション・ロジックは、インターフェイスからフラッシュメモリへのアクセスを、次の優先順位で調停します。優先順位の高いポートのフラッシュメモリへのアクセスは、優先順位の低いポートをブロックすることで可能になります。

1. JTAG ポート ~ 全ての MachXO2 デバイスには JTAG ポートがあり、それを使用してフラッシュメモリ (UFM/ コンフィグレーション) へのリード / ライト動作が実行されます。JTAG ポートは、

IEEE 1149.1 及び IEEE 1532 仕様に準拠しています。JTAG は、フラッシュメモリ (UFM/ コンフィグレーション) への最高の優先順位を持っています。

2. スレーブ SPI ポート ~ 全ての MachXO2 デバイスにはスレーブ SPI ポートがあり、それを使用してフラッシュメモリ (UFM/ コンフィグレーション) へのリード / ライト動作が実行されます。フラッシュメモリ (UFM/ コンフィグレーション) のスレーブ・チップセレクト (ufm_sn) をアサートすると、アクセスが可能になります。
3. プライマリ I²C ポート ~ 全ての MachXO2 デバイスには I²C ポートがあり、それを使用してフラッシュメモリ (UFM/ コンフィグレーション) へのリード / ライト動作が実行されます。プライマリ I²C ポートのアドレスは yyyyxxxx00 ('y' 及び 'x' はユーザがプログラム可能) です。
4. WISHBONE スレーブ・インターフェイス ~ EFB モジュールの WISHBONE スレーブ・インターフェイスでは、WISHBONE マスタを作成することで、FPGA のユーザロジックからフラッシュメモリ (UFM/ コンフィグレーション) にアクセスできます。WISHBONE バス信号だけでなく、割り込み要求出力信号が FPGA ファブリックに渡されます。WB マスタによる UFM/CF のプログラミング動作に役立つよう、IRQ (wbc.ufm_irq) が用意されています。IRQ は構成可能で、R/W FIFO 及び調停エラーに関する情報を提供します。

注: R1 デバイスの WISHBONE 経由でフラッシュメモリ (UFM/ コンフィグレーション) にアクセスするには、ハード SPI ポートまたはプライマリ PC ポートをイネーブルする必要があります。詳細については、"AN8086, Designing for Migration from MachXO2-1200-R1 to Standard (Non-R1) Devices (マigrations の設計)" を参照してください

注: イネーブル・コンフィグレーション・インターフェイス・コマンド 0x74 透過モードを使用して、フラッシュメモリ (UFM/ コンフィグレーション) インターフェイスをイネーブルすると、次のようなデバイスの一部機能が一時的にディセーブルになります

- ・パワーコントローラ
- ・GSR
- ・ハードマクロ・ユーザ SPI ポート
- ・ハードマクロ・プライマリユーザ PC ポート

ディセーブル・コンフィグレーション・インターフェイス・コマンド 0x26 と、それに続いてバイパス・コマンド 0xFF を使用すると、フラッシュメモリ (UFM/ コンフィグレーション) インターフェイスのディセーブル後に、機能が回復されます。

UFM へのインターフェイス

MachXO2-640 以上の規模のデバイスには、1 セクタのユーザフラッシュメモリ (UFM) が用意されています。JTAG、コンフィグレーション SPI、プライマリ・コンフィグレーション PC、または WISHBONE インターフェイスを使用して、コンフィグレーション・ロジック・インターフェイスから UFM セクタにアクセスできます。UFM はオンチップのフラッシュメモリから分離されたセクタとなっていて、ページで構成されます。各ページは 128 ビット (16 バイト) あります。表 9-11 に、各デバイス内の UFM リソースをビット数、バイト数、及びページ数で示します。)

表 9-11. MachXO2 デバイスの UFM リソース

	MachXO2-256	MachXO2-640	MachXO2-640U	MachXO2-1200	MachXO2-1200U	MachXO2-2000	MachXO2-2000U	MachXO2-4000	MachXO2-7000
UFM ビット数	0	24,576	65,536	65,536	81,920	81,920	98,304	98,304	262,144
UFM バイト数	0	3,072	8,192	8,192	10,240	10,240	12,288	12,288	32,768
UFM ページ数	0	192	512	512	640	640	768	768	2048

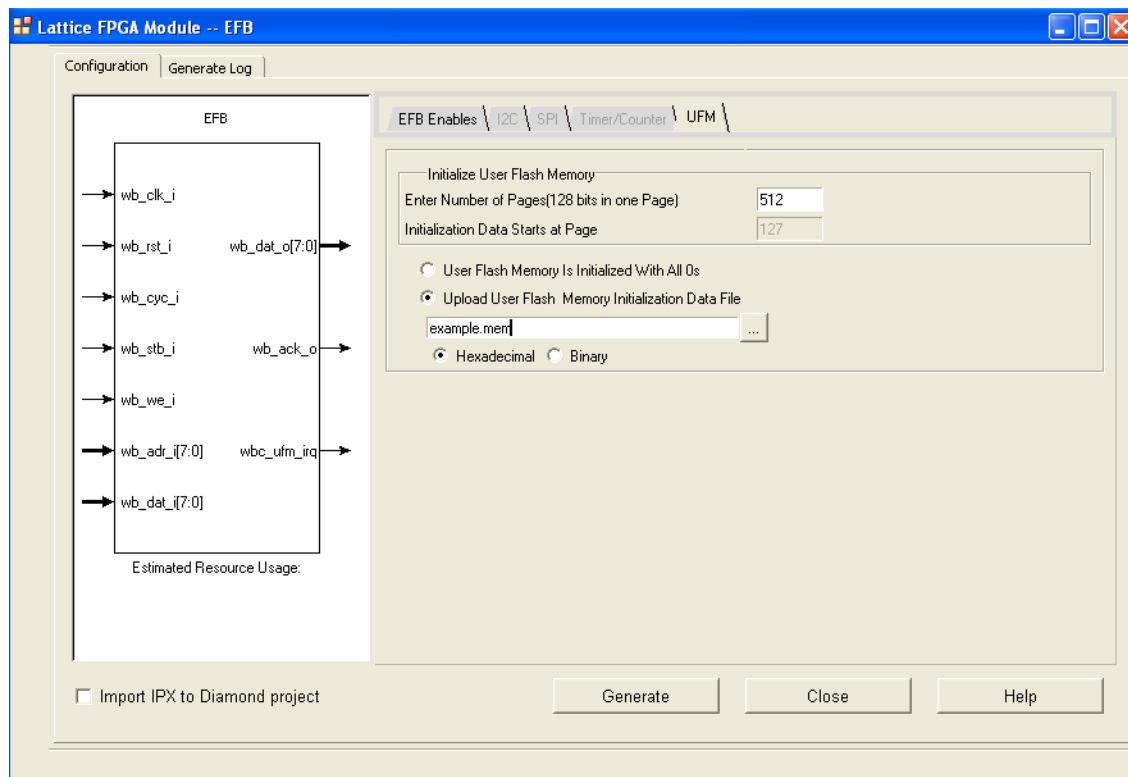
UFM は汎用フラッシュメモリです。プログラミングや消去の様々な仕様については、MachXO2 ファミリのデータシートを参照してください。一般的に、UFM はシステムレベルのデータ、組み込みブロック RAM の初期化データ、またはマイクロプロセッサまたはステートマシン用の実行可能コードの格納に使用されます。UFM リソースの分割には、次のツールを使用します。

- ・ IPexpress: UFM をイネーブルし、メモリを初期化するには IPexpress を使用します
- ・ Spreadsheet View (スプレッドシート・ビュー) (Diamond) : グローバルな sysConfig コンフィグレーション・オプションによって、UFM の使用が制御されます。利用可能なオプションの詳細については、“TN1204, MachXO2 Programming and Configuration Usage Guide (MachXO2 プログラミングとコンフィグレーションのユーザガイド)” をご覧ください

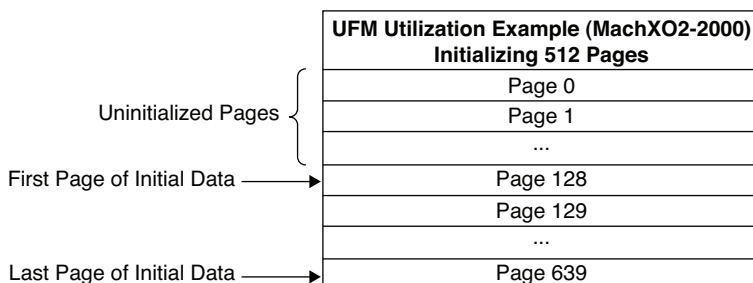
IPexpress による UFM の初期化

IPexpress を使用することで、UFM セクタをシステムレベルの不揮発性データで初期化し、EFB モジュールを生成できます。EFB GUI で UFM タブを選択すると、UFM の構成可能な設定が表示されます。

図 9-20. IPexpress による UFM セクタの初期化



初期化データをプリロードするページ数を入力します。初期化データは "アドレス下位揃え (bottom justified)" されるため、初期化される最初のページのアドレスを、読み出しのみのフィールド "Initialization Data Starts at Page (初期化データの開始ページ)" で確認できます。本書で使用する例では、MachXO2-2000 の UFM セクタの 512 ページが、メモリファイル "example.mem" の内容で初期化されます。



MachXO2-2000 には UFM セクタが 640 ページあります。初期化データは、UFM のボトム（上位）のアドレス指定可能ページに格納され、トップ（下位）のアドレス指定可能ページは未初期化のまま（全てゼロ）となります。メモリファイルの形式はページベースとなり、2 進と 16 進のどちらの形式でも表現できます。

UFM 初期化用メモリファイル

初期化用メモリファイルのプロパティと形式は、次のとおりです。

拡張子: .mem
 フォーマット: 2 進、16 進
 データ幅: 1 ページ（ファイルの 1 行が 128 ビット）
 行数: 使用可能なページ数以下

例 1. 2 進数

1010...1010 (UFM 初期化データの開始ページに配置、アドレス = N)
 1010...1010 (ページアドレス = N + 1)
 1010...1010 (ページアドレス = N + 2)
 ...
 1010...1010 (最上位 UFM ページアドレスに配置)

例 2. 16 進数

A...B (UFM 初期化データの開始ページに配置、アドレス = N)
 C...D (ページアドレス = N + 1)
 E...F (ページアドレス = N + 2)
 ...
 A...F (最上位 UFM ページアドレスに配置)

ページの最上位バイト（バイト 15）は、行の左端にあります。ページの最下位バイト（バイト 0）は行の右端にあって、行の最上位ビットは左端（ビット 127）、最下位ビットは右端（ビット 0）にあります。

EBR の初期化

UFM セクタを EBR 初期化データと共有することにした場合は、デザインの計画時に次に示すセクタマップを例として参照する必要があります。ただし、現時点で Diamond は EBR マッピングに対応していません。

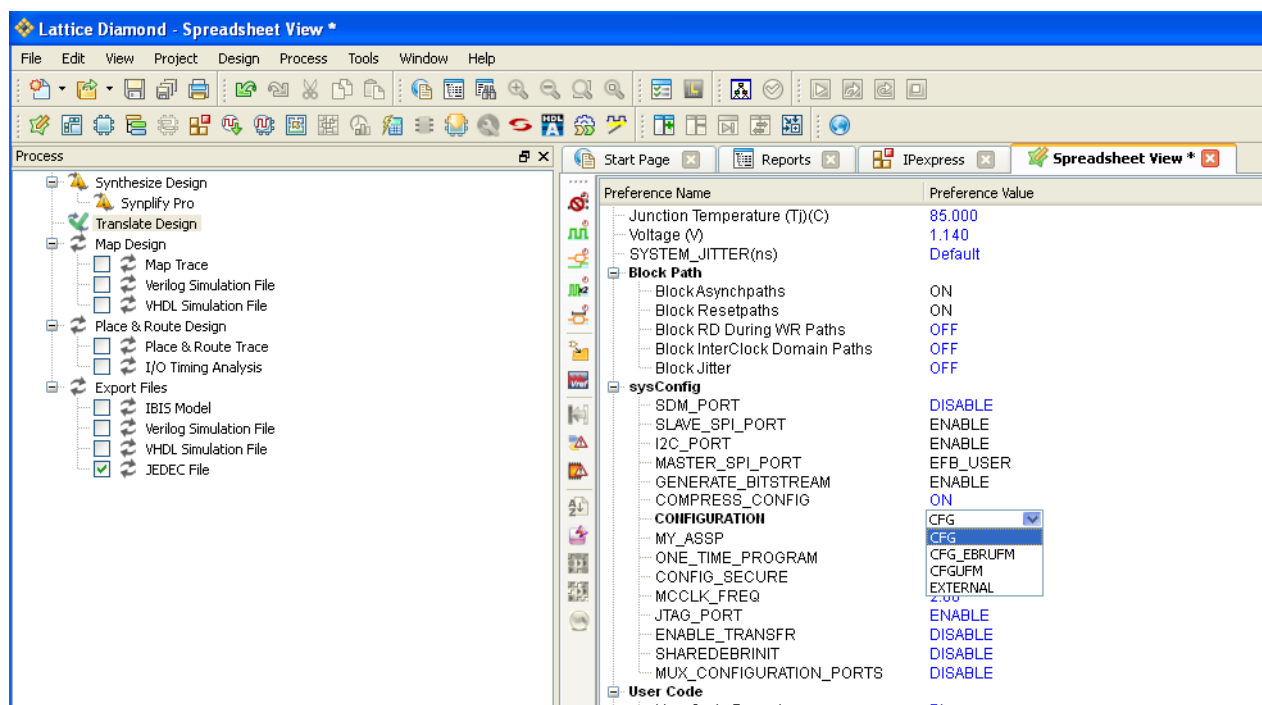
UFM Sector (MachXO2-2000)	
Block N Location and Initiliazation Data	Page 0 (EBR Block N Location)
	Page 1 (EBR Block N Initialization Data)
	...
	Page 72 (EBR Block N Initialization Data)
Block M Location and Initialization Data	Page 73 (EBR Block M Location)
	Page 74 (EBR Block M Initialization Data)
	...
	Page 145 (EBR Block M Initialization Data)
Pages Available for General Purpose Non-volatile Data	Page 146
	...
	...
	Page 639

一括消去動作を実行するときは、EBR 初期化データが失われないよう注意する必要があります。UFM セクタを消去する前に、EBR ブロック位置と EBR 初期化データが格納されたページを、2 次メモリリソースにコピーしておく必要があります。これにより UFM セクタの消去後、データを UFM に書き戻すことができます。電源投入時、または再構成コマンドが実行されると、EBR 初期化データが自動的に対応する EBR ブロックにロードされます。EBR ブロック位置データは、コンフィグレーション・ロジックに EBR ブロックの位置を知らせます。

Lattice Diamond ソフトウェアでの UFM

UFM セクタは、MachXO2 のフラッシュメモリ・リソースの 1 つです。Diamond の Spreadsheet View (スプレッドシート・ビュー) にある CONFIGURATION オプションでは、UFM の動作が制御されます。各オプションは、UFM に次のような影響を与えます。

図 9-21. Diamond Spreadsheet View での UFM



CONFIGURATION パラメータのオプション

- CFG_EBRUFM - The SRAM configuration (not including EBR initialization data) is stored in the Configuration Flash. EBR ~ SRAM コンフィグレーション (EBR 初期化データを含まない) がコンフィグレーション・フラッシュに格納されます。EBR 初期化データがある場合は、それが UFM セクタの最下位ページアドレス (ページ 0 から開始) に格納されます。EBR 初期化データのマッピング後に未占有の UFM ページは、汎用メモリとして使用可能です。
- CFG ~ SRAM コンフィグレーション (EBR 初期化データがあればそれを含む) はコンフィグレーション・フラッシュアレイに格納され、UFM にオーバーフローしません。UFM セクタ全体を汎用フラッシュメモリとして使用可能です。
- CFGUFM ~ SRAM コンフィグレーション (EBR 初期化データがあればそれを含む) はコンフィグレーション・フラッシュに格納され、UFM へのオーバーフローを許可されます。UFM を汎用メモリとして使用できません。
- EXTERNAL ~ SRAM コンフィグレーション (EBR 初期化データがあればそれを含む) は外付けメモリの SPI メモリに格納されます。UFM セクタ全体を汎用フラッシュメモリとして使用可能です。

コンフィグレーション・フラッシュメモリ

EFB モジュールの WISHBONE インターフェイスでは、WISHBONE マスタが MachXO2 デバイスのコンフィグレーション・リソースにアクセスできます。このことは、USERCODE や TraceID などのコンフィグレーション・リソースからのデータリードで特に便利です。WISHBONE マスタは、コンフィグレーション・ロジックの透過モードを使用して、コンフィグレーション・フラッシュメモリを更新できます。新しいデザインは、電源投入後、またはコンフィグレーションのリフレッシュ動作後にアクティブになります。

MachXO2 のプログラミングの詳細については、“TN1204, MachXO2 Programming and Configuration Usage Guide (MachXO2 プログラミングとコンフィグレーションのユーザガイド)” 及び “TN1246 Using User Flash Memory and Hardened Control Functions in MachXO2 Devices Reference Guide (MachXO2 デバイスのユーザフラッシュメモリとハードマクロ制御機能の使用法リファレンスガイド)” を参照してください。

TraceID の詳細については、“TN1207, Using TraceID in MachXO2 Devices (MachXO2 デバイスでの TraceID の使用法)” を参照してください。

表 9-12. MachXO2 デバイスのコンフィグレーション・フラッシュリソース

	MachXO2-256	MachXO2-640	MachXO2-640U	MachXO2-1200	MachXO2-1200U	MachXO2-2000	MachXO2-2000U	MachXO2-4000	MachXO2-7000
CFM ビット数	73,600	147,328	278,400	278,400	409,344	409,344	737,024	737,024	1,179,136
CFM バイト数	9,200	18,416	34,800	34,800	51,168	51,168	92,128	92,128	147,392
CFM ページ数	575	1,151	2,175	2,175	3,198	3,198	5,758	5,758	9,212

フラッシュメモリ (UFM/ コンフィグレーション) 設計のヒント

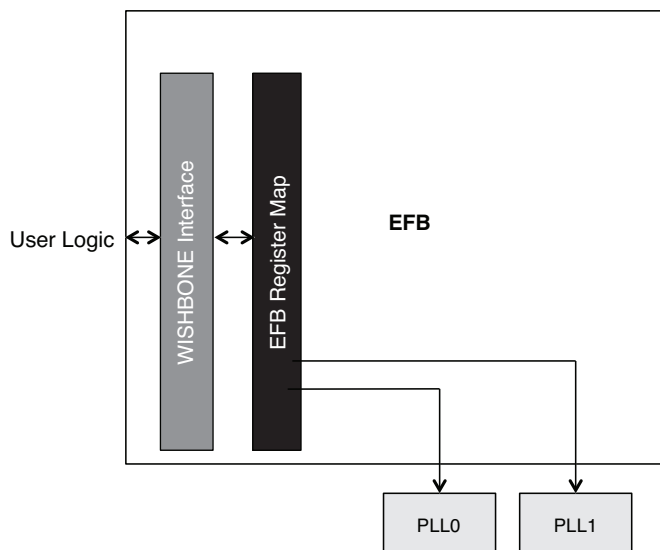
1. フラッシュメモリ (UFM/ コンフィグレーション) レジスタの定義と、コマンドシーケンスの詳細については、“TN1246, Reference Guide for Using User Flash Memory and Hardened Control Functions in MachXO2 Devices (MachXO2 デバイスのユーザフラッシュメモリとハードマクロ制御機能の使用法リファレンスガイド)” のフラッシュメモリ (UFM/ コンフィグレーション) の項を参照してください。
2. MachXO2 のプログラミングの詳細については、“TN1204, MachXO2 Programming and Configuration Usage Guide (MachXO2 プログラミングとコンフィグレーションのユーザガイド)” を参照してください。
3. TraceID の詳細については、“TN1207, Using TraceID in MachXO2 Devices (MachXO2 デバイスでの TraceID の使用法)” を参照してください。
4. EFB では、消費電力削減のためコンポーネントを動的にオフする場合に、WISHBONE クロックのソースになっているかどうかにかかわらず、MachXO2 の内部オシレータをイネーブルにする必要があります。
5. フィーチャ行にはコンフィグレーション・ポートの恒久的設定が格納されているため、コンフィグレーション・フラッシュにアクセスするときは、初期プログラミング後にフィーチャ行をリードオンリーにすることを推奨します。
6. UFM 及び SPI 機能をイネーブルすると、コンフィグレーション・フラッシュのチップセレクト (ufm_sn) がイネーブルされます。コンフィグレーション・フラッシュ /UFM のアクセスにスレーブ SPI インターフェイスを使用しない場合は、ufm_sn が非アクティブ (High) になります。
7. フラッシュメモリ及び UFM のアクセスに使用するバスには優先順位があります。バスの優先順位は JTAG ポートが最も高く、スレーブ SPI ポート、プライマリ I²C ポート、WISHBONE スレーブ・インターフェイスの順に低くなります。優先順位の低いポートのフラッシュメモリへのアクセスは、優先順位の低いポートをブロックすることで可能になります。各コンフィグレーション・ポートのマスタによる、コンフィグレーション・フラッシュ /UFM への同時アクセスを防止するプロセスを定義する必要があります。
8. プライマリ I²C ポートはフラッシュメモリへのアクセスとユーザ・ファンクションへのアクセスを、単一デザインで両方利用できるようにすることはできません。
9. イネーブル・コンフィグレーション・インターフェイス・コマンド 0x74 透過モードを使用して、フラッシュメモリ (UFM/ コンフィグレーション) インターフェイスをイネーブルすると、パワーコントローラ、GSR、ハードマクロ・ユーザ SPI ポートが一時的にディセーブルになります。ディセーブル・コンフィグレーション・インターフェイス・コマンド 0x26 と、それに続いてバイパス・コマンド 0xFF を使用すると、フラッシュメモリ (UFM/ コンフィグレーション) インターフェイスのディセーブル後に、機能が回復されます。
10. フラッシュメモリは、0 は消去済み、1 は書き込み済みを定義します。

11. ライト動作（ビット ⇒ 1）の最小単位は 1 ページ（16 バイト）です。
12. 消去動作（ビット ⇒ 0）の最小単位は 1 セクタです（デバイスには二つ、UFM セクタとコンフィグレーション・フラッシュ・セクタのみです）。
13. UFM の消去 / プログラミングのサイクル数には制限があります。このサイクル数については DS1035 に説明があります。UFM には静的データ、またはあまり変化しないデータを格納することを推奨します。
14. ラティスのウェブサイト（www.latticesemi.com/products/intellectualproperty/aboutreferencedesigns.cfm）には、次のような多数のフラッシュメモリ（UFM/ コンフィグレーション）参照デザインがあります。
 - a.RD1126: RAM-Type Interface for Embedded User Flash Memory（組み込みユーザフラッシュメモリ用 RAM 型インターフェイス）
 - b.UG57: MachXO2 Programming via WISHBONE Demo（WISHBONE からの MachXO2 プログラミング・デモ）

動的 PLL 構成設定用インターフェイス

EFB モジュールの WISHBONE インターフェイスを使用すると、MachXO2 デバイス内の PLL（Phase Locked Loop）の構成可能な設定を動的に更新できます。

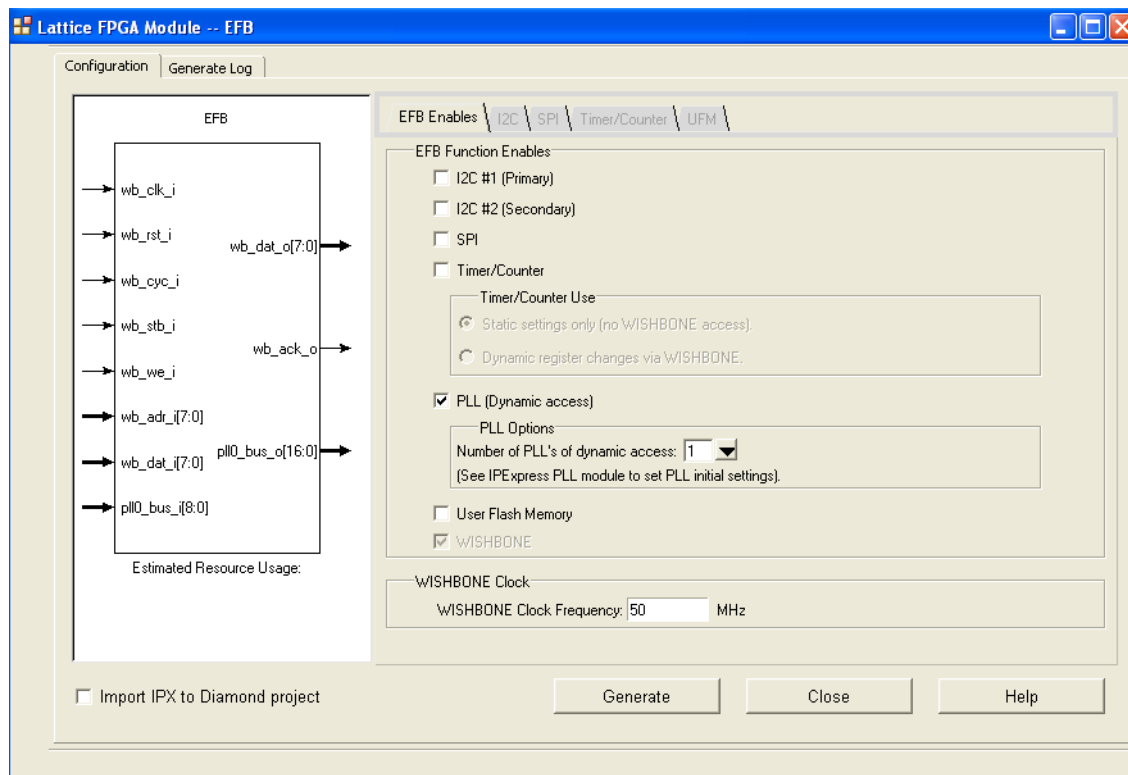
図 9-22. 動的 PLL 設定の EFB インターフェイス



MachXO2 デバイスには最大 2 個の PLL があります。PLL0 のアドレス範囲は、EFB レジスタマップ内で 0x00 ～ 0x1F です。PLL1（もしあれば）のアドレス範囲は、EFB レジスタマップ内で 0x20 ～ 0x3F です。PLL の構成ビットと推奨する使用法については、“TN1199, MachXO2 sysCLOCK PLL Design and Usage Guide (MachXO2 sysCLOCK PLL 設計と使用法ガイド)” を参照してください。

図 9-23 に示すように、PLL コンポーネントへの WISHBONE インターフェイスを、EFB GUI の IPexpress でイネーブルできます。

図 9-23. 動的 PLL 構成設定用インターフェイス



WISHBONE インターフェイスから PLL 設定を動的に制御するインターフェイスをイネーブルにすると、PLL への専用接続に使用される次のポートを持つ IP が生成されます。

表 9-13 に、IP で生成される信号を示します。各信号には、使用方法と、デザインプロジェクトの中での接続方法に関する説明が記述されています。

表 9-13. PLL インターフェイス ～ IP 信号

信号名	I/O	幅	記 述
pll0_bus_i	入力	9	入力データ及び制御バス。このバスは、デザイン内にインスタンス化される PLL コンポーネントだけに接続する
pll0_bus_i	入力	9	入力データ及び制御バス。このバスは、デザイン内にインスタンス化される PLL コンポーネントだけに接続する
pll0_bus_o	出力	17	出力データ及び制御バス。このバスは、デザイン内にインスタンス化される PLL コンポーネントだけに接続する
pll0_bus_1	出力	17	出力データ及び制御バス。このバスは、デザイン内にインスタンス化される PLL コンポーネントだけに接続する

テクニカルサポート支援

直通電話 :1-800-LATTICE (北米)

+1-503-268-8001 (北米以外)

e-mail: techsupport@latticesemi.com

インターネット :www.latticesemi.com

日本語版更新履歴

日付	バージョン	新ページ	更新内容
2012 年 6 月	04.0	-	旧 TN1205 を 1205 と TN1246 に分割した初版
2012 年 10 月	04.1	7	表 9-3、注記追加
		7	図 9-5、差し替え
		8	プライマリ I2C 項、(図の上) 記述更新、(図の下) 文章追加
		8	図 9-6、差し替え
		14	I2C 設計のヒント項、#4 追加
		29	タイマ / カウンタユースケース、3-IV-2、比較 --> キャプチャ
		32	ページ中央、注記の項目追加
		37	フラッシュメモリ・・・項、#8 追加、#12 補足修正
2013 年 8 月	04.2	8	ページ上部機能一覧中、I2C クロックストレッチを削除
		10	表 9-5、_scl / _sda に " 外部ノイズフィルタ " 使用推奨を追記
		13-14	I2C 設計のヒント節、第 7 ~ 10 項目を追加
2014 年 1 月	04.3	4	WISNBONE 設計のヒント、1 項に記述追加
		5	EFB 生成について、内部オシレータに関する記述追加
		11	表 9-5、i2c2_sci / _sda 記述の最後の推奨文をそれぞれ削除
		14	I2C 使用例の 4b、最後の文章を削除
		14	I2C 設計のヒント、第二項に例外記述追記
		23	SPI 設計のヒント、第二項に例外記述追記
		31	タイマカウンタ設計のヒント、第二項に例外記述追記
2014 年 9 月	4.4	15	『I2C 設計のヒント』項 10 の記述を更新