AN81623

PSoC® 3、PSoC 4、PSoC 5LP デジタル設計のベストプラクティスについて

著者: Todd DustMark Ainsworth

関連プロジェクト: なし

関連部品ファミリ: すべての PSoC 3 と PSoC 5LP 部品、および PSoC 4 4200 ファミリ

ソフトウェア バージョン: PSoC Creator™ 2.2 SP1 以降

すべてのアプリケーション ノートの一覧を表示するには、ここをクリックしてください。

本アプリケーション ノートの最新版または関連プロジェクト ファイルについては、
http://www.cypress.com/go/AN81623 をご覧ください。

本アプリケーション ノート (AN81623) は、デジタル ハードウェアの設計理論を簡単に紹介し、PSoC 3、PSoC 4 (4200 ファミリ) および PSoC 5LP 内の強力かつ柔軟性の高いサブシステムについて説明します。また本書は、PSoC Creator を用いてデジタル設計する際のベストプラクティスを紹介し、静的タイミング解析 (STA) レポート ファイルの使用方法を示します。

目次

はじめに 2
デジタル設計とは? 3
- デジタル設計の基本概念 3
- デジタル設計の方法 3
- デジタル デザインにおけるタイミング問題 5

PSoC におけるデジタル設計 7
- PSoC デジタル サブシステム 7
- PSoC 3 および PSoC 5LP のクロック 8
- PSoC における同期化 11
- PSoC Creator 静的タイミング解析 (STA) 12

PSoC デジタル設計上の注意事項 13
- トピック 1: コンポーネントのデータシート仕様 13
- トピック 2: クロックを使用 13
- トピック 3: 制御レジスタ 17
- トピック 4: ラッチを作らない 17
- トピック 5: デジタル機能コンポーネントを使用 18
- トピック 6: ピンとのインターフェース 19
- トピック 7: 固定機能ブロックとのインターフェース 20

PSoC Creator STA レポートの使用 21
- STA の温度条件を設定 21
- STA レポートを探す 21
- STA レポートの要素 21
- STA レポートを使用して警告を解除 28
- クロックの公称周波数および必要な周波数 33
- ホールド時間、回復時間および除去時間の違反 35

まとめ 35
関連アプリケーション ノート 36
ワールドワイド販売と設計サポート 38

はじめに

PSoC 3、PSoC 4 および PSoC 5LP のペリフェラル システムは、高性能かつ高い柔軟性をもつプログラミング可能なシステムです。固定機能ブロックの一式 (4 個のタイマー、I2C、USB、CAN) に加え、最大 24 個のプログラム可能なユニバーサル デジタル ブロック (UDB) およびデジタル システム インターコネクト (DSI) という広範な信号配線システムを備えています。図 1 は、このシステムがいくつかの PSoC デバイスにおいてどのように配置および使用されるかを示します。

図 1. PSoC 3 および PSoC 5LP の
プログラム可能なデジタル アーキテクチャ

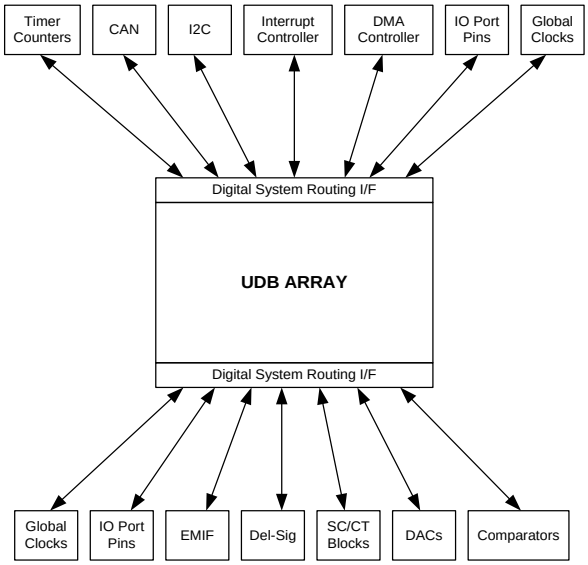
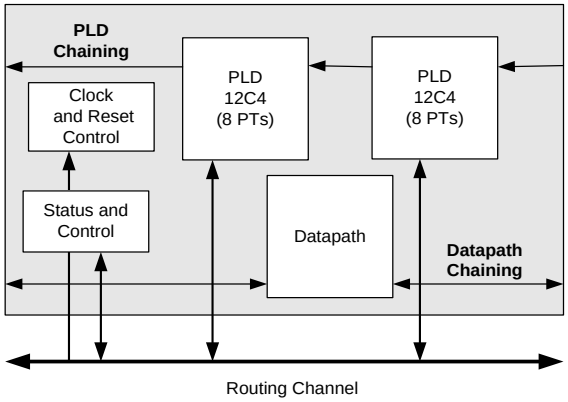


図 2 に示すように、各 UDB は 2 個の小さいプログラマブル論理デバイス (PLD)、プログラム可能な 8 ビット ALU を内蔵した、データパス モジュールおよびその他のレジスタと機能を含んでいます。

図 2. PSoC 3 および PSoC 5LP UDB のブロック図



UDB の PLD とデータパスをプログラムし、UDB 内部および UDB 同士間で信号を配線することで、CPU の負担を大幅に軽減する複雑なカスタム ペリフェラルを作成できます。図 1 に示すように、DSI による追加の配線により、これらのペリフェラルは I/O ピンや DMA、アナログ システムなどその他の PSoC 要素とインターフェースできます。

PSoC Creator IDE はプログラム済みのペリフェラルの大きなライブラリを提供しており、ユーザー独自のカスタム デザインを設計することが可能です。

PSoC のコンフィギュレーション可能性により、デザインをシステム レベルで最適化する、かつてない機会をもたらします。ただし、従来の MCU ユーザーが気付いていない全く新しい一連の設計上の注意事項があります。本アプリケーション ノートは PSoC 3 および PSoC 5LP の様々なデジタル設計トピックに関する情報を提供しています。

本アプリケーション ノートは、ユーザーが PSoC Creator を使った PSoC 3、PSoC 4 または PSoC 5LP のアプリケーションの開発に精通していることを前提としています。これらの製品に慣れていない場合は、「AN54181, Getting Started with PSoC 3」、「AN79953, Getting Started with PSoC 4」および「AN77759, Getting Started with PSoC 5LP」を参照してください。PSoC Creator の初心者である場合、PSoC Creator ホームページを参照してください。

一般的なデジタル設計原理の基本概念は、「デジタル設計とは ?」で説明します。

デジタル設計原理に精通していて、PSoC Creator を使用してデジタル設計を PSoC に適用する方法を理解したい場合は、「PSoC におけるデジタル設計」および「PSoC Digital の設計上の注意事項」を参照してください。

PSoC Creator STA レポート ファイルの読み方の具体的情報については、「PSoC Creator STA レポートの使用法」を参照してください。

より高度な PSoC デジタル設計リソースは、「関連アプリケーション ノート」に記載します。

注 : 本アプリケーション ノートに示す例は、ほとんどの条件下で指定された静的タイミング解析 (STA) 警告を生成するために設計されています。ただし、配線や配置、クロック周波数によっては、警告を生成できない場合もあります。これは特に、タイミング ドリブン配置 (TDP) 機能を備えた PSoC Creator 2.1 に当てはまります。

デジタル設計とは?

MCU 分野だけの仕事をしていれば、低レベルのデジタル設計問題に直接対応する必要はほとんどありません。理由は、現在の MCU は高度に集積されている、すなわち、ほとんどのメモリとペリフェラルは CPU と共に単一のチップに内蔵されているためです。ユーザーは、CPU がメモリまたはペリフェラル レジスタ アドレスに正しい順序でアクセスするようコードを書くだけで十分です。

PSoC 3 および PSoC 5LP は従来の MCU と異なり、汎用デジタル (とアナログ) 構成要素からユーザー独自のカスタム ペリフェラルを構築することを可能にします。このためには、まずいくつかのデジタル設計の基本概念を紹介します。

本アプリケーション ノートの後半の説明はすべて、本節で紹介する概念に基づいてきます。

デジタル設計の基本概念

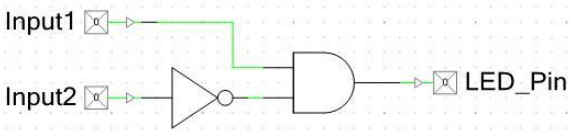
デジタル設計プロセスは実際的に CPU ファームウェアを書くことに似ていますが、デジタル デザインは CPU よりもはるかに速く実行できるという利点があります。たとえば、1 つの条件が真 (論理 HIGH または「1」)、もう 1 つが偽 (論理 LOW または「0」) になった時 LED 表示器を点灯させる場合を考えてみましょう。機能を実行するための C コードを以下のように書けます。

```
if (GetInput1() && !GetInput2())
{
    SetOutput(1); // turn on the LED
}
else
{
    SetOutput(0); // turn off the LED
}
```

この方法は、実行するのに多くの CPU サイクルを要します。

しかし、CPU コードを書く代わりに、図 3 に示すように単に低レベル論理ゲートを用いて機能を作成できます。

図 3. デジタル論理関数の例

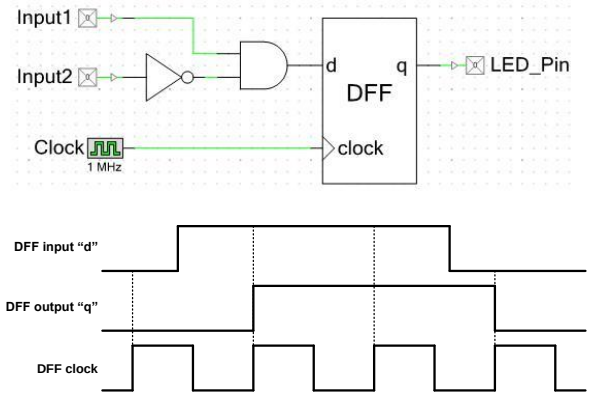


その他の MCU とは異なり、図 3 に示したデザインを PSoC Creator プロジェクト回路図に描き、その後プログラムして PSoC デバイスに直接実装することが可能です。唯一の遅延はピンやゲート、配線により発生し、ナノ秒のレンジです。もちろん、CPU サイクルはかかりません。

レジスタおよび DFF

図 3 に示したデザインには、入力が迅速にまたは同時に変化した場合、出力に望ましくないグリッチが発生してしまうという問題があります。この可能性を低下させるためには、図 4 に示すクロック制御レジスタ (「D フリップフロップ」、「DFF」とも呼ばれる) を使用します。

図 4. クロック制御のデジタル論理関数の例



レジスタの出力「q」は、「clock」入力が 0 から 1 に遷移する時、すなわち、「立ち上がりエッジ」でのみ変化します。出力は入力「d」と同じ値になり、クロックの次の立ち上がりエッジまでその値のままです。

クロックとは、固定周波数、50%デューティ比の連続駆動矩形波です。PSoC Creator では、デバイスの限界まで任意の望ましい周波数の、複数のクロックを PSoC に適用することができます。8 ページの「PSoC 3 および PSoC 5LP のクロック」を参照してください。

クロック制御レジスタを使うと、イベント検出と応答タイミングの制御がより効果的になります。これは、非同期入力を既存のクロック信号と「同期させる」こととも呼ばれています。

入力がクロック エッジと正確に同時に変化すると、出力はある期間不確定になる可能性があります。この状態はメタスタビリティと呼ばれています。このトピックの詳細情報は、6 ページの「メタスタビリティおよびレジスタ タイミング」を参照してください。

デジタル設計の方法

従来の MCU のユーザーは複数のレベルで MCU をプログラムします。

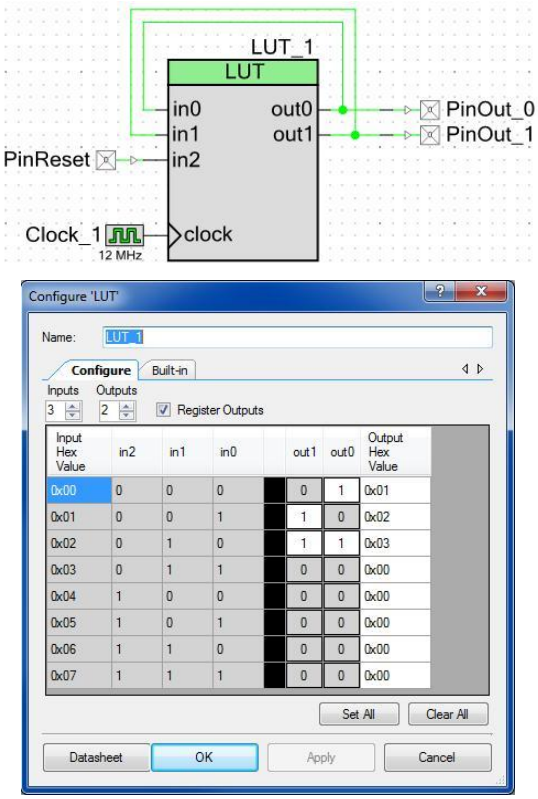
- **アセンブラ:** コードが高度に最適化されている一方、コーディング効率が低く、移植が不可能です。常に、特定のモジュールにのみ使用されています。
- **C:** コーディング効率および移植性が高いです。MCU のコーディングに最も一般的に使用される言語です。
- **オブジェクト指向言語 (Java、C++など):** カスタム オブジェクトの定義を可能にし、再利用性およびコーディング効率が高いです。常に、少なくとも 256K のメモリを備えた、32 ビットなどのハイエンド MCU に使用されています。

同様に、デジタル設計も複数のレベルで行われます。

- **ゲート レベル:** 個別のゲート (AND、OR、XOR、NOT) を DFF と接続して論理関数を実行します。PSoC Creator はすべての論理関数のゲート シンボルを用意しており、また、より少し高いレベルでルックアップ テーブル (LUT) コンポーネントを提供しています。

  LUT を使用すると、個別のゲートを互いに接続せずに複雑な論理関数を容易に設計することができます。また、MCU をプログラムせずにステートマシンを簡単に設計することもできます。図 5 は、例としてリセット機能付きの 2 ビット カウンターを示します。

図 5. LUT ベースのステートマシン デザイン



PSoC 3 および PSoC 5LP においてステートマシン設計の詳細については、「AN62510 – Implementing State Machines with PSoC 3 and PSoC 5LP」を参照してください。

- **コード レベル:** PSoC Creator は、C 言語と同様な構造と構文を持つ、Verilog というハードウェア設計言語をサポートしています。
- **データパス プログラミング:** ゲート、LUT および Verilog レベルのデザインは常に PSoC UDB 内の PLD に実装されています (図 2 を参照してください)。データパスなど UDB の他の部分をプログラムするために、PSoC Creator はデータパス コンフィギュレーション ツールを用意しています。ほとんどの UDB ベース デザインはデータパスと PLD 両方を使用しています。
- **コンポーネント:** オブジェクト指向プログラミングと同様に、PSoC Creator は実装と再利用の容易なカスタム コンポーネントを定義することを可能にします。コンポーネントは、UDB データパスおよび PLD を含む様々な PSoC リソースを使用します。

注 : Verilog、データパス プログラミングおよびコンポーネント開発の詳細な説明は本アプリケーション ノートの範囲を超えています。これらのトピックの詳細情報については、「関連アプリケーション ノート」を参照してください。

デジタル デザインのデバッグ

ファームウェア開発はデバッグのツールと技術を含んでいます。以下はファームウェア向けのいくつかの適切なデバッグ技術です。

- 各コード行をステップスルーし、あらゆる状態でその動作を少なくとも 1 回観察します。
- 開発しながらデバッグし、すなわち、小さいコード ブロックを書き、テストしてから次のブロックに進みます。

同様に、デジタル設計ではゲート、モジュール、またはコンポーネント レベルの性能を監視する必要があることがあります。これを PSoC で実施する最も簡単な方法は、信号をテスト ピンに配線して、オシロスコープまたはロジック アナライザを使ってピンの動作を観察します。ファームウェアと同様に、デジタル デザインをデバッグする際に以下のことを行う必要があります。

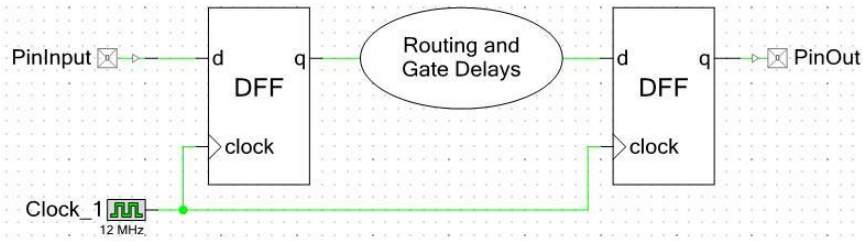
- 各信号をあらゆる状態で少なくとも 1 回観察します。
- 開発しながらデバッグします。この時、ロジックまたはコンポーネント レベルでデザインの小さい部分を実行し、テストしてからデザインの次の部分に進みます。

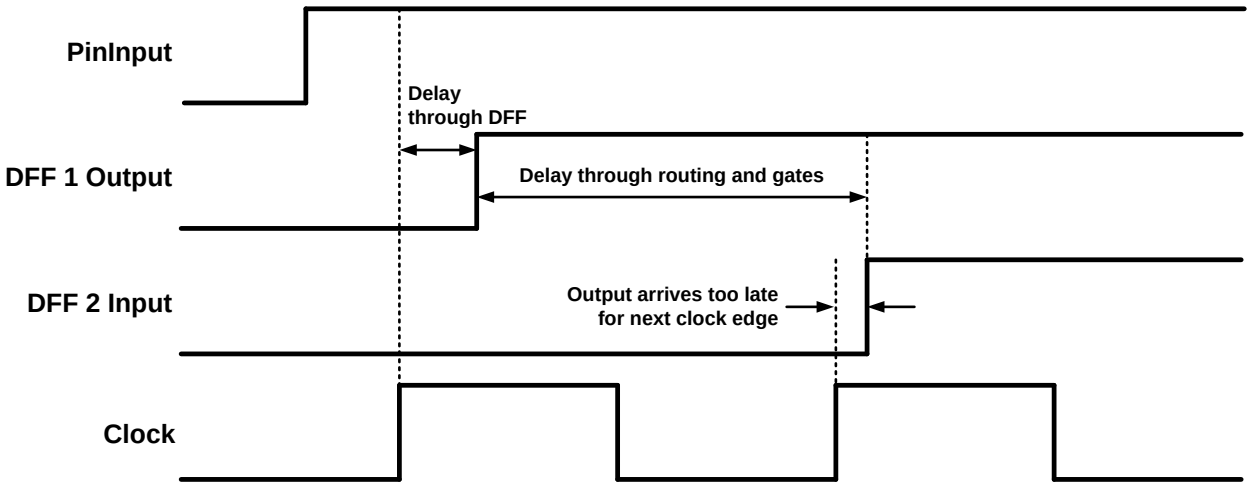
デジタル デザインにおけるタイミング問題

デジタル デザインにおいて最も一般的な問題はタイミングです。DFF やゲート、その他の論理要素により生じる遅延、およびピンと配線により生じる遅延が必ずあります。タイミング問題は、遅延が図 6 に示すように、正しい処理をするためのタイミングよりも、信号の到着が遅すぎる (早すぎることもある) 場合に現れます。

図 6 では、DFF 1 により生じる遅延およびロジックと DFF 間の配線により生じる遅延のため、信号が DFF 2 入力に遅れて到着しています。これを訂正するには、クロックを減速させる、または DFF 間の遅延を低減させる必要があります。

図 6. マルチレジスタ デザインにおける遅延



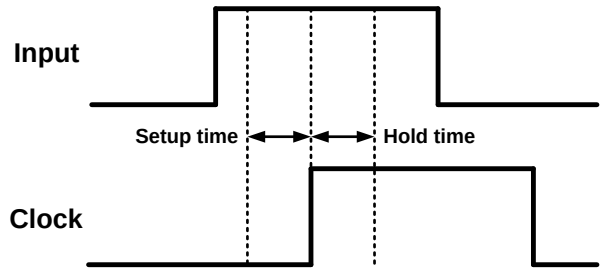


メタスタビリティおよびレジスタ タイミング

前述したように、レジスタ入力がクロック エッジと正確に同時に変化すると、出力はある期間不確定になる可能性があります。この条件をメタスタビリティ (metastability) と呼びます。

もっと正確に言えば、クロック エッジの前と後には、入力が安定することが必要となる期間があります。その期間は図 7 に示すように、レジスタの規定されたセットアップ時間とホールド時間で定義されます。

図 7. レジスタのセットアップ時間とホールド時間

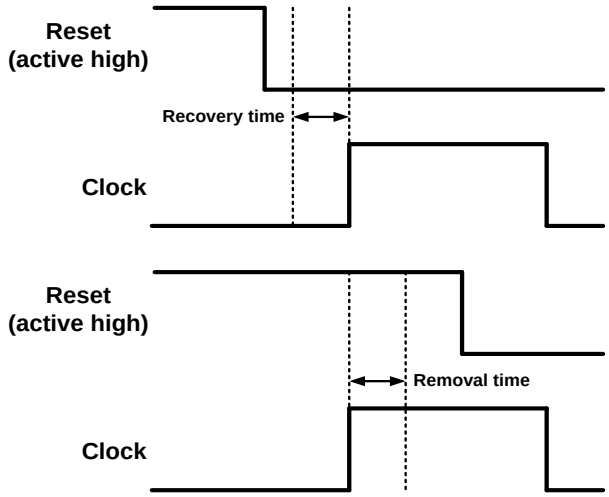


一般的には、入力はセットアップ時間とホールド時間中にその状態を維持する必要があります。そうしないと、出力は不確定、すなわち、メタステーブル (metastable) 状態に入ります。

回復時間 (recovery time) と除去時間 (removal time) はセットアップ時間とホールド時間に似ていますが、レジスタ入力の代わりにクロックを基準とした非同期制御のタイミングを示します。非同期制御の一例は DFF リセット入力です (図 30 および 18 ページの説明テキストを参照してください)。

回復時間とは、非同期制御がクロック エッジまで非アクティブになる必要がある最小時間です。除去時間とは、制御がクロック エッジからアクティブのままである必要がある最小時間です。図 8 を参照してください。

図 8. レジスタの回復時間と除去時間



静的タイミング解析 (STA)

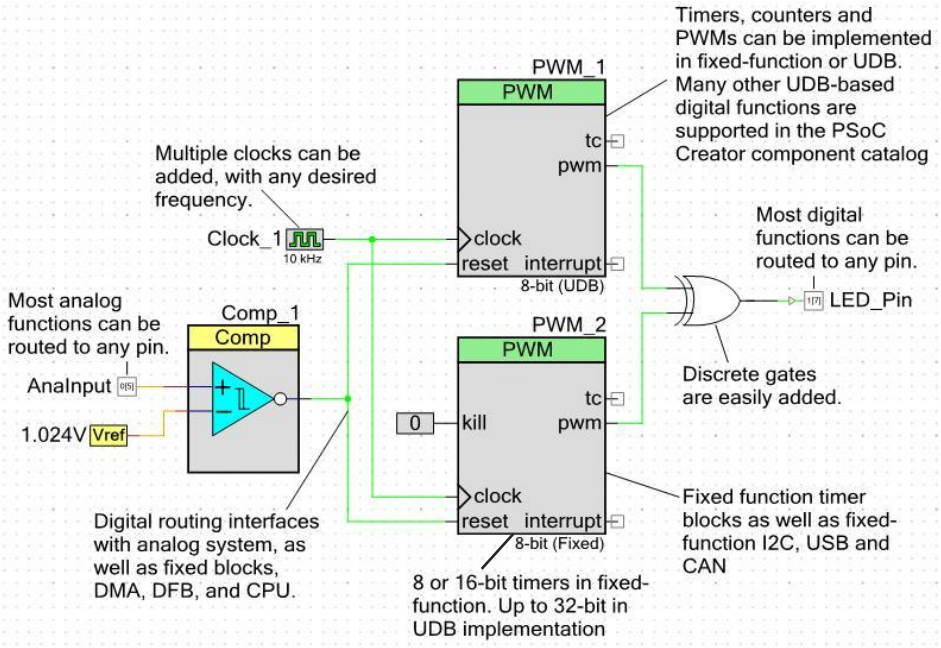
デジタル デザインのデバッグの重要な一部は、静的タイミング解析 (STA) です。STA はデジタル デザインを評価し、信号の入出力間の遅延を計算します。それらの遅延に基づいて、デザインで使用される各クロックの最大許容周波数を算出します。詳細については、「PSoC Creator 静的タイミング解析」を参照してください。

PSoC におけるデジタル設計

本アプリケーション ノートでは詳しく説明していませんが、PSoC Creator で利用できる PSoC デジタル サブシステムの柔軟性の高い機能がどのくらい多いか、また複雑なデザインを素早く設計することがどのくらい簡単であるかを認識することが重要です。

たとえば図 9 は、アナログ電圧レベルが「ブリージング」(呼吸イルミネーション) LED を有効／無効にする PSoC 3 と PSoC 5LP デザインです。2 個の PWM は少し異なる周期を持っており、LED が徐々に暗くなったり明るくなったりします。PSoC Creator に精通しなくても、半時間以内にそのようなデザインを PSoC デバイスに構築して実行できます。

図 9. PSoC 3 および PSoC 5LP における PSoC Creator デジタル デザイン例 (PSoC 4 は同様)



もう一つの例では、CPU を使用せずに多重化された ADC 入力および SRAM のデータを節約するための DMA とのインターフェースを制御して、高度なアナログ データ収集システムを作成するデジタル システムを設計します。

詳細については、PSoC Creator ウェブページ」や PSoC Creator 内の任意のサンプル デザインを参照してください。

PSoC デジタル サブシステム

前述したように、PSoC デジタル サブシステムの中心は UDB のアレイです (図 1 を参照してください)。デジタル コンポーネントを PSoC Creator 回路図に配置すると、一連のデータパス、PLD やその他の UDB レジスタ (図 2) が設定されて望ましい機能を実現します。

UDB の内部およびその周囲には DSI があります。DSI とは、UDB 内で、一対の UDB 間で、UDB アレイ全体にわたり、また UDB アレイとその他の様々な PSoC ブロック間で信号を接続するプログラム可能なスイッチの広範囲な構造です。これにより、PSoC がインテリジェントなカスタム ペリフェラルを構築する際に優れた柔軟性を得られます。

PSoC Creator 回路図でデジタル コンポーネント間のワイヤーを描くと、ソース点とデスティネーション点間の一連の DSI スイッチがオンにされてその接続を実現します。各スイッチは 1～2 ナノ秒の遅延を持っています。この遅延が短いですが、信号が複数のスイッチを介して配線されると、総遅延時間は問題になります。

これを管理するために、PSoC Creator は以下の機能を備えています。

- タイミング駆動配線 (TDR) : タイミング クリティカルな信号が、DSI スイッチ数とルーティング遅延を最小限にするように配線されます。
- タイミング駆動配置 (TDP) : UDB リソースが、コンポーネントのタイミングをさらに最適化するように選択されます。

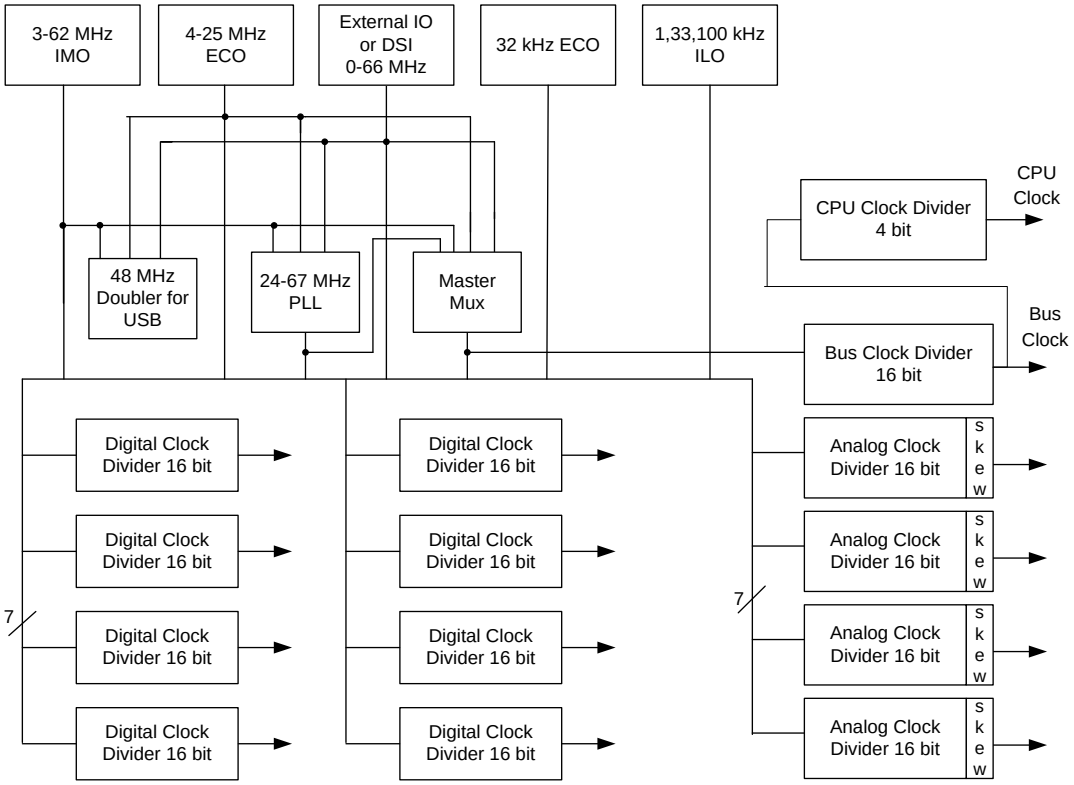
PSoC デジタル サブシステムの詳細情報については、いずれかのデバイス データシートまたはテクニカル リファレンス マニュアル (TRM) を参照してください。

PSoC 3 および PSoC 5LP のクロック

本アプリケーション ノートでは、信号のクロックとの同期化を中心とした説明が多いので、PSoC クロック システムおよびそれが PSoC Creator でどのように提供されるかを簡単に説明します。

PSoC 3 と PSoC 5LP は柔軟性の高いクロック生成と供給システムを備えています。図 10 は PSoC 3 を例として示します。PSoC 5LP は同様です。PSoC 4 のクロック システムはよりはるかに簡単で、詳細については PSoC 4 デバイス データシートを参照してください。

図 10. PSoC 3 のクロッキング システム

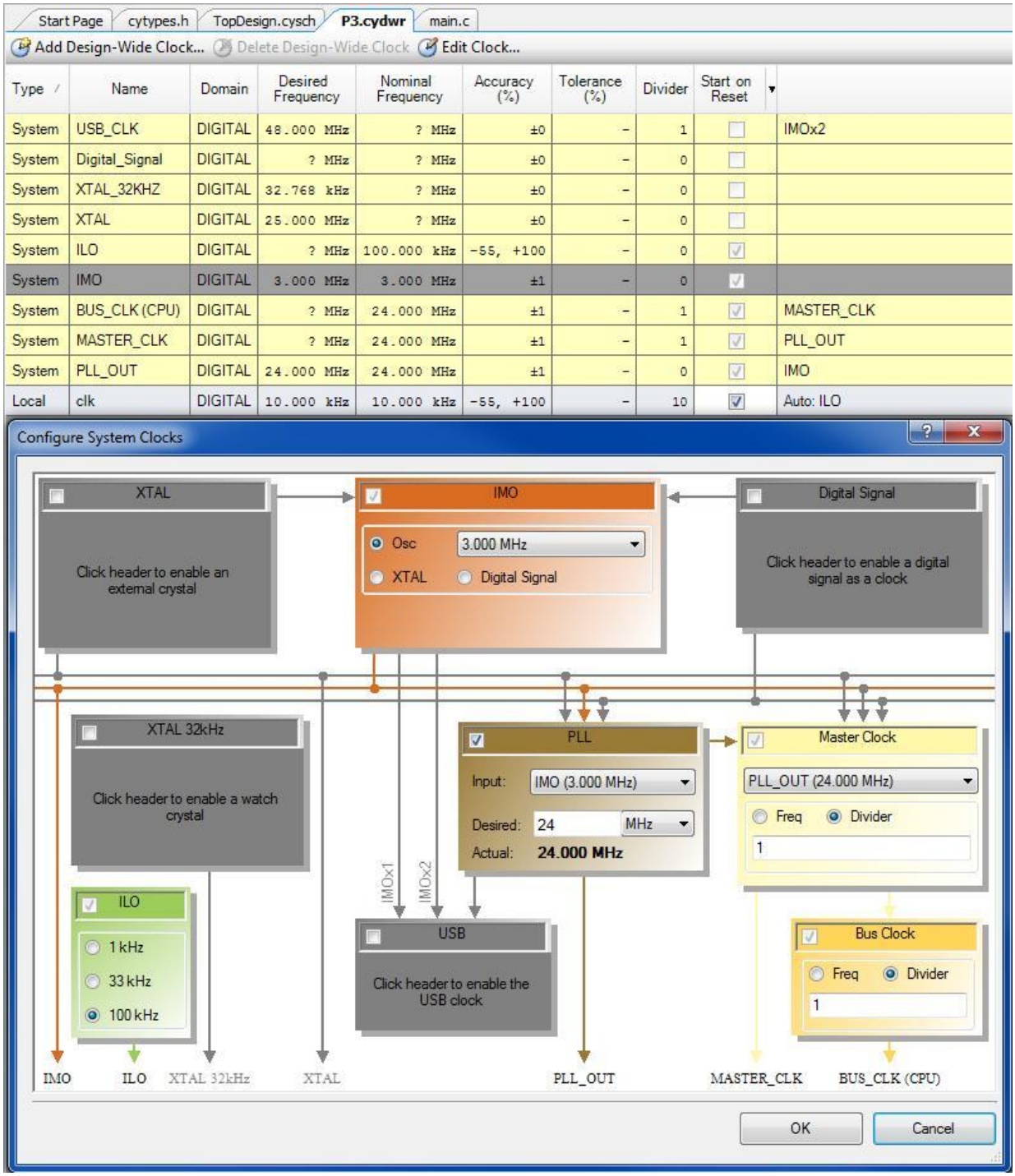


異なる周波数範囲、精度、電源要件を持つ様々なクロック ソースがあります。これはシステムの他の部分に送信され、バス クロック、デジタル クロックおよびアナログ クロックが生成されます。デジタル システムでは、8 個のクロック分周器が、必要とする任意の周波数のクロックを提供します。

クロッキング システムは PSoC Creator でセットアップされます。ユーザー定義のクロックは回路図ページに配置され (7 ページの図 9)、システム クロック設定はデザイン ワイド リソース (DWR) ページに示されます。9 ページの図 11 は、図 9 の例のクロック コンフィギュレーションを示します。

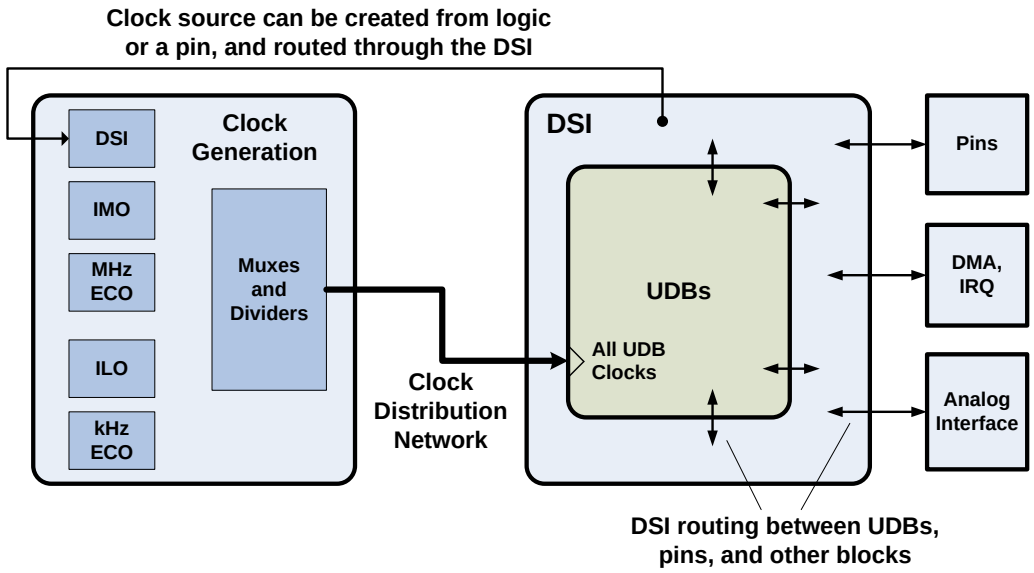
マスター クロックとバス クロックおよびそれらの使用に注意してください。バス クロックが常にマスター クロックと同じ周波数を持っているため、これらの 2 つの用語が区別せずに用いられることが多いです。しかし、それらの機能は異なります。マスター クロックは、バス クロックを含む他のすべての PSoC クロックのソースであり、図 10 に示した分周器を使用します。その周波数は、PSoC における他のすべてのクロックの周波数以上です。バス クロックは CPU、DMA、DFB およびその他の主なブロックのクロック ソースです。

図 11. PSoC 3 および PSoC 5LP における PSoC Creator クロック コンフィギュレーション例



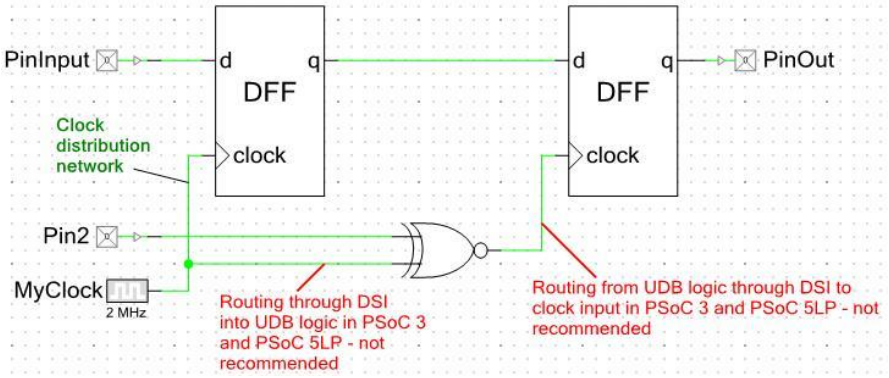
どのクロック ソースも UDB クロック入力に配線できます。クロックは図 12 に示すように、専用ネットワークを通して (DSI から独立している) UDB のクロック入力およびその他のブロックに供給されます。

図 12. PSoC 3 および PSoC 5LP におけるクロック供給および DSI



PSoC 3 および PSoC 5LP では、クロックは図 13 に示すように UDB 内のロジックに配線することもできます。ただし、一般的には、クロック コンポーネントなどのクロック ソースをクロック入力に直接接続したほうが良いです。これにより、クロック供給ネットワークの利点を生かせます。また、クロックのゲートまたはデータ入力への配線、ゲートまたはデータ出力のクロック入力への配線をしないほうが良いです。詳細については、13 ページの「トピック 2: クロックを使用」および 15 ページの「クロックをゲートで制御せず、イネーブル コンポーネントを使用」を参照してください。

図 13. クロックの配線経路



PSoC の詳細情報については、「AN60631, PSoC 3 and PSoC 5LP Clocking Resources」または PSoC Creator システム リファレンス ガイドの「クロッキング」章を参照してください。

PSoC における同期化

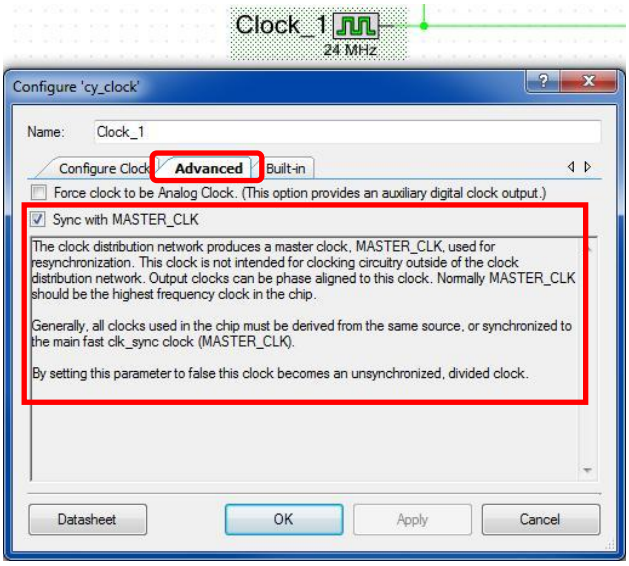
前述したように、非同期デジタル システムは信号に望ましくないグリッチを発生させることがあります。PSoC は、同期システムとして動作し、デジタル サブシステムと CPU および DMA 間の通信を可能にするよう設計されています。一般的には、CPU または DMA と通信しない PLD ベースのロジックを除き、非同期信号はサポートされていません。

以下に、PSoC において注意すべき同期化の例を示します。

PSoC 3 および PSoC 5LP のクロック

前述したように、PSoC 3 および PSoC 5LP には任意の望ましい周波数のクロックをデジタル サブシステムに提供する 8 個のクロック分周器があります。デフォルトでは、これらのクロックはマスター クロックと同期化されています。これは図 14 に示すように、PSoC Creator クロック コンポーネント コンフィギュレーション ダイアログで制御します。

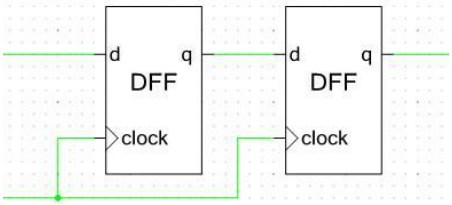
図 14. PSoC 3 および PSoC 5LP におけるクロック同期化設定



ピン (すべての PSoC デバイス)

入力ピンに送信される信号は非同期と見なされ、デフォルトでは、DSI に配線される前に二重同期化されています。二重同期化は図 15 に示すように、メタスタビリティおよび予期しないシステム動作の可能性をさらに低くします。

図 15. 二重同期化

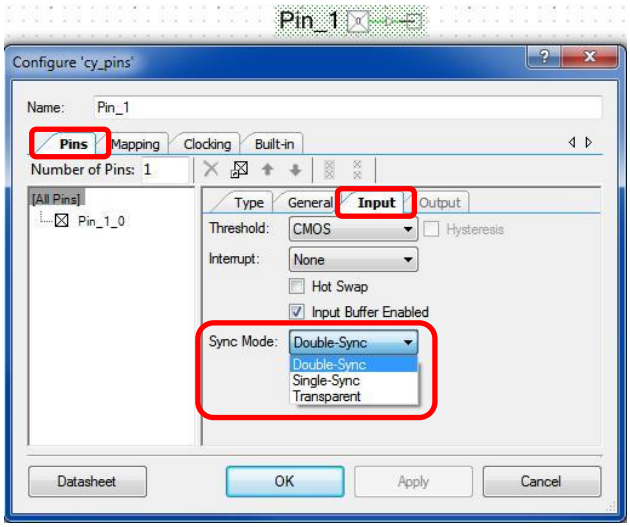


ダブルシンクロナイザーにより生じる遅延は同期化クロックの 1～2 サイクルであることに注意してください。PSoC 3 および PSoC 5LP では、同期化クロックはバス クロックであり、常にマスター クロックと同じ周波数を持っています。PSoC 4 では、同期化クロックは HFCLK です。

同期化クロックが 33MHz 以下の場合、ピン ブロック内の専用回路は二重同期化用に使用されます。バス クロックが 33MHz より大きい場合、ピン入力は DSI を介して UDB 回路に接続され、二重同期化は UDB 内で行われます。これにより、マスタークロック、バス クロックまたは HFCLK が 33MHz 以上または以下に変更されると、配線および STA 結果が変化します。たとえば、30 ページの図 50 前後の説明を参照してください。

前述したように、クロック ソースは PSoC の外部から、I/O ピンを介して配線されます。入力がバス クロックのソースである場合、入力同期化を無効にする必要があります。これは図 16 に示すように、PSoC Creator 内のピン コンポーネント コンフィギュレーション ダイアログで、「Sync Mode」を「Transparent」にセットすることで行います。

図 16. 入力ピンの同期化



一般的には、ピン入力は CPU または DMA によって正しく読み出されるように同期化する必要があります。ただし、PSoC の外部システムのタイミングが重要な場合があります。そのような場合には、遅延の同期化は許可されないことがあり、入力ピンの同期化は無効にする必要があります。詳細については、19 ページの「トピック 6: ピンとのインターフェース」を参照してください。

またピン出力は同期化オプションがあり、すなわち、バス クロックと (単一) 同期化することができます。 このオプションは、複数の出力間のスキューや相対遅延の減少を主な目的としています。このオプションはデフォルトでは無効です。

UDB の制御レジスタおよびステータス レジスタ

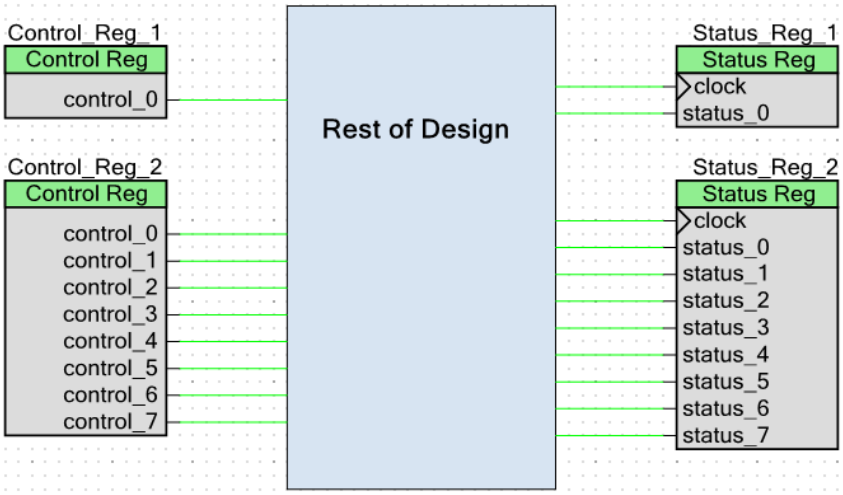
もう 2 つの主な UDB コンポーネントは制御レジスタとステータス レジスタです。各 UDB にはそれぞれ 1 つがあります。制御レジスタにより、CPU／DMA は信号をデジタル サブシステムに駆動し、ステータス レジスタにより、CPU／DMA はデジタル サブシステムから信号を読み出します。両方のレジスタに対応する PSoC Creator コンポーネントがあります。レジスタの詳細については、TRM を参照してください。

これらのレジスタは通常動作モードに加え、タイミング制約を緩和するための特殊モードもあります。たとえば、ステータス レジスタは 4 ビット ダブルシンクロナイザーにリコンフィギュレーションすることができます。この機能を利用するには、14 ページの図 22 に示すように PSoC Creator の Sync コンポーネントを使用します。

もう一つの例では、制御レジスタは 3 つのモードで動作できます。同期化とパルス モードでは、バス クロックと同期化された CPU／DMA からの直接、同期化、パルスの入力を、その他のクロックと再同期化することができます。詳細については、17 ページの「トピック 3: 制御レジスタ」を参照してください。

注 ：PSoC Creator の制御レジスタおよびステータス レジスタ コンポーネントは図 17 に示すように、8 本の制御出力またはステータス入力にコンフィギュレーションすることができます。複数のシングルワイヤー制御コンポーネントまたはステータス コンポーネントを作成することが可能ですが、各コンポーネントは完全な UDB レジスタを使用します。より良い方法は、複数の出力と入力を持つ制御およびステータス コンポーネントを使用することです。これは特に、共通のクロックを使用するステータス入力に当てはまります。

図 17. シングルおよびマルチワイヤーの制御レジスタおよびステータス レジスタ



PSoC Creator 静的タイミング解析 (STA)

PSoC Creator プロジェクトが構築されるたびに、STA が自動的に行われます。各クロックの周波数を制限するデザインでの重要なパスを示すレポートが生成されます。実際のクロック周波数が計算された最大周波数を超えると、デザインにタイミング違反が存在することを示す警告が生成されます。

次の節で説明するように、STA 警告を回避するために PSoC における同期化およびその他のデジタル機能を理解することが重要です。

PSoC デジタル設計上の注意事項

本節は、PSoC 3、PSoC 4 および PSoC 5LP の PSoC Creator を用いたデジタル設計の注意事項とベストプラクティスを説明するトピックを含んでいます。

トピック 1: コンポーネントのデータシート仕様

PSoC Creator コンポーネント カタログで用意されている、UDB から構築されたデジタル コンポーネント (カウンターや SPI など) を使用する際、コンポーネントのタイミング仕様を確認することが重要です。それらの仕様は図 18 に示すようにコンポーネント データシートに記載されています。

図 18. UDB カウンターのタイミング仕様

Table 2. AC Characteristics

Parameter	Description	Min	Typ	Max[2]	Units
f_{CLOCK}	Component clock frequency				
	8-bits UDB Up Counter	–	–	39	MHz
	8-bits UDB Counter with direction	–	–	39	MHz
	16-bits Up Counter	–	–	33	MHz
	16-bits Counter with direction	–	–	33	MHz
	24-bits UDB Up Counter	–	–	29	MHz
	32-bits UDB Up Counter	–	–	26	MHz

コンポーネント タイプに応じて、UDB ベース コンポーネントのクロック周波数は、前述した配線による遅延のため制限されます。クロックの周波数はデータシート記載の最大値を超えないことが重要です。

詳細については、いずれかのデジタル コンポーネントのデータシートを参照してください。

トピック 2: クロックを使用

可能な限り遅いクロックを使用

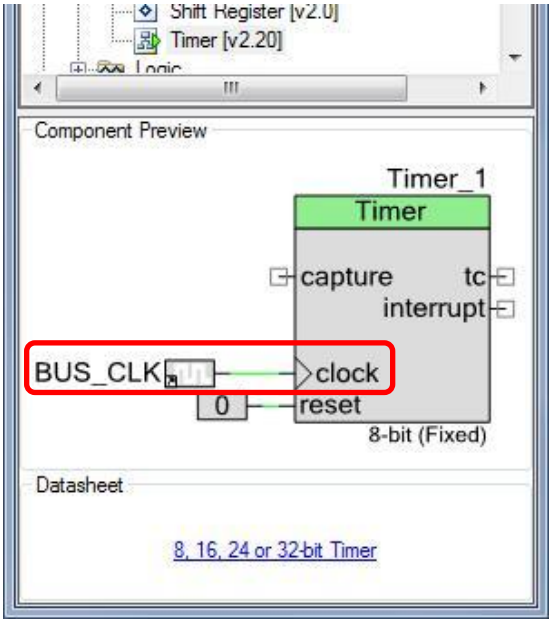
前述したように、PSoC Creator コンポーネントのクロック周波数はコンポーネントのデータシート仕様に応じて制限する必要があります。

一般的には、すべてのクロック周波数はアプリケーション要件を満たしながら、可能な限り遅くする必要があります。これにより、PSoC Creator の TDR システムはデザインの重要な部分の配線を最適化する余裕ができます。また、システム ノイズおよび消費電力も減少します。

マクロでのクロック設定

図 19 に示すように、高周波数クロックをすぐに確定することができない場合があります。PSoC Creator コンポーネント カタログ内のいくつかのマクロは、デフォルトとしてバス クロックにセットされているクロック コンポーネントを含んでいます。バス クロックが必要でない場合は、クロック周波数を変更する必要があります。

図 19. タイマー マクロはバス クロックを含む



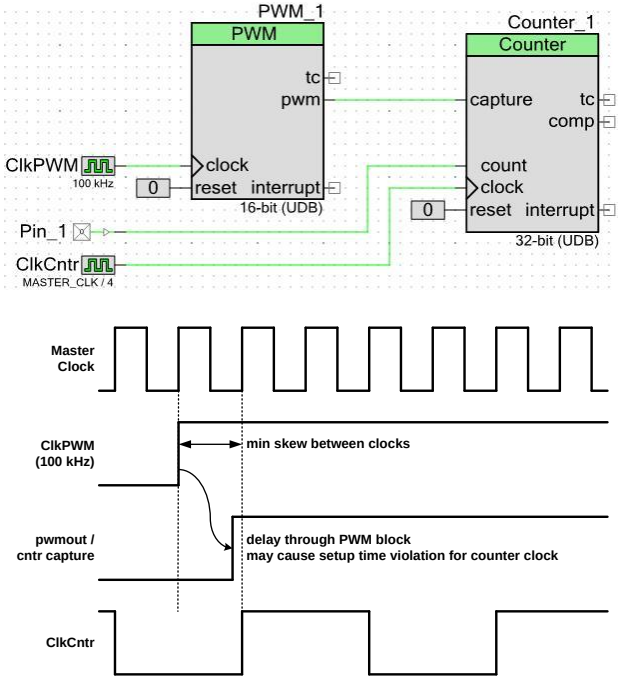
複数のクロック

前述したように、PSoC 内のほとんどのデジタル信号は同期化されています。ほとんどのクロックでもその他の周波数の高いクロックと同期化されています。クロックはマスター クロックに同期化されている場合が多いです (この場合に限りません) (図 10 を参照してください)。

複数のクロックを同じソースと同期化することができますが、図 20 に示すように、それらは互いに同期する必要がありません。この例は PWM からの周期的なキャプチャ時間を持つ周波数カウンターを示します。この例では、2 つのクロックはマスター クロック (PSoC 4 では HFCLK) と同期化されていますが、互いに同期する必要がありません。そのため、マスター クロック周波数が十分高ければ、PWM コンポーネントによる遅延はカウンター コンポーネントにセットアップ時間違反を発生させる可能性があります。

注 本節およびその他の節に示した例は、ほとんどの条件下で指定された STA 警告を生成するために設計されています。ただし、配線や配置、クロック周波数によっては、警告を生成できない場合もあります。

図 20. マルチクロックの一例



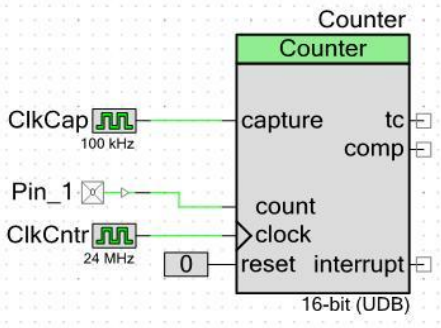

```
Warning-1366: Setup time violation found in
a path from clock (ClkPWM) to clock
(ClkCntr).
```

この場合、クロック間の最小時間は 1 マスター クロック周期であるため、ClkCntr 周波数を低下させても問題が修正されません。問題を修正するために、次の節で説明するようにマスター クロック周波数を低下させることも、PWM 出力を ClkCntr と同期させることもできます。28 ページの STA レポートを使用して警告を解除を参照してください。

Sync コンポーネントによりクロックを同期化

図 21 に示す簡単な例では、低周波数クロックを使ってカウンターのキャプチャ周波数を直接制御します。これで、STA から同期化の警告が発生する可能性があります。

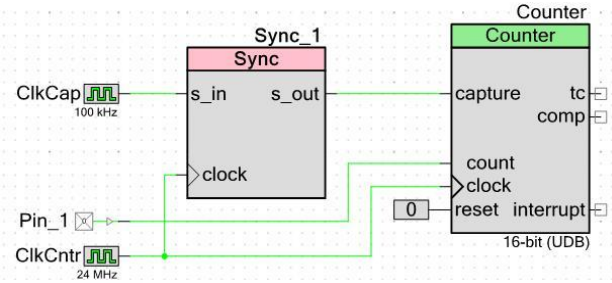
図 21. マルチクロックのもう一例




```
Warning-1350: Path(s) exist between clocks
ClkCap(routed) and ClkCntr, but the clocks
are not synchronous to each other
```

この警告を解除するために、図 22 に示すように PSoC Creator の Sync コンポーネントを使って低周波数クロックをカウンター クロックと同期させます。

図 22. クロックを同期化



Sync コンポーネントが二重同期化をするため (11 ページの図 15 および 12 ページの「UDB の制御レジスタおよびステータス レジスタ」を参照)、同期化出力のデューティ比は正確に 50% になることはありません。これは、同期化クロックが同期元のクロックよりはるかに速い場合に許可されます。しかし、2 つのクロックがほぼ同じ周波数である場合、同期化出力はどのクロックも生成せず、単に DC レベルを維持することがあります。同期化クロックは同期元のクロックより 2 倍以上速い必要があります。4 倍が推奨されています。

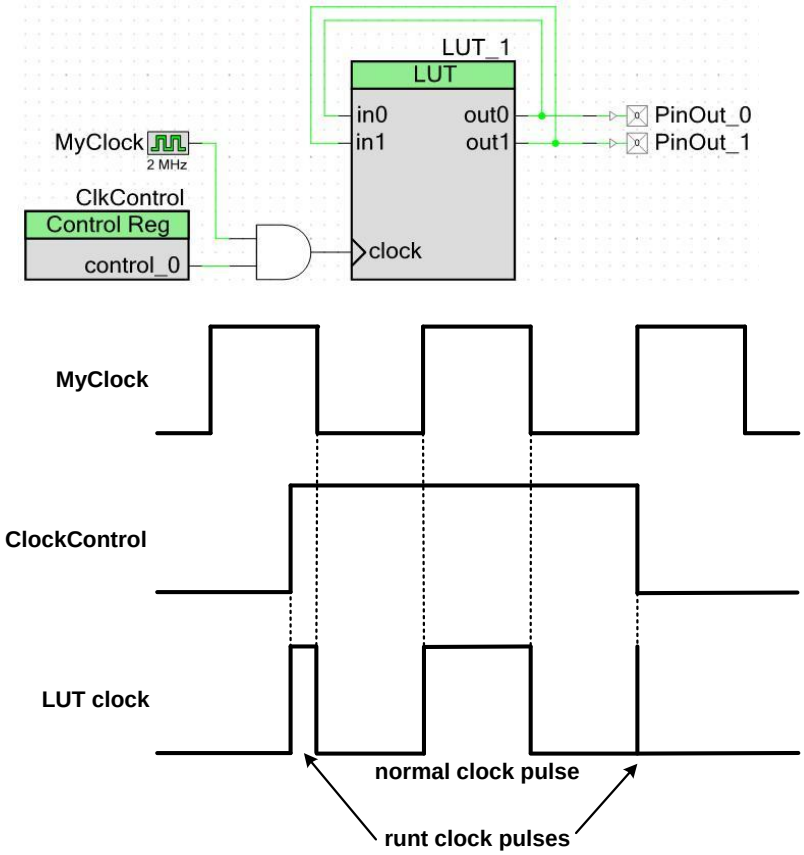
Sync コンポーネントの詳細情報については、Sync コンポーネントのデータシートを参照してください。

注 : 図 21 と図 22 では、PSoC 4 であれば、ClkCap はカウンターの「capture」端子または Sync の「s_in」端子に直接接続することができません。良い回避方法は、ClkCap の周波数 (200kHz) を倍増させ、まずトグル フリップフロップ (TFF) コンポーネントを通って走らせることです。

クロックをゲートで制御せず、イネーブル コンポーネントを使用

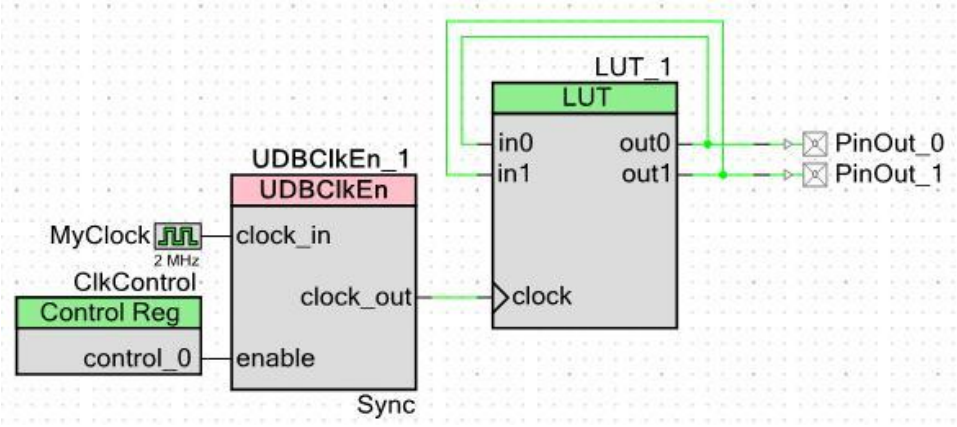
多くのデザインでは、クロックをゲートで制御して単にクロックをオンやオフに切り替えることで機能を制御します。しかし図 23 の例は、非同期化制御を使ってクロックをゲートで制御することが、どのように狭いクロック パルスを発生させ、予期できない結果をもたらすかを示します。

図 23. クロックのゲーティング例



機能が UDB をベースとする場合 (図 23 の例と同様)、図 24 に示すように UDBClkEn コンポーネントを使用したほうが良いです。このコンポーネントは、制御された、レベルに伴って変化するイネーブル信号をクロックに提供し、オプションとして入力クロックをバス クロックと同期化させることができます。UDBClkEn コンポーネントの詳細情報については、UDBClkEn コンポーネントのデータシートを参照してください。PSoC 4 では、クロックは UDBClkEn の「clock_in」端子に直接接続できます。

図 24. UDBClkEn コンポーネントを使用



注 : すべてのデジタル機能コンポーネント (18 ページの「トピック 5: デジタル機能コンポーネントを使用」を参照) は UDBClkEn コンポーネントを組み込んでいます。そのため、UDBClkEn コンポーネントは常に DFF または LUT コンポーネントと併用すれば良いです。UDBClkEn コンポーネントは別の UDBClkEn コンポーネントを駆動することができません。そうしようとすると、以下のエラーの 1 つ以上の例が発生します。

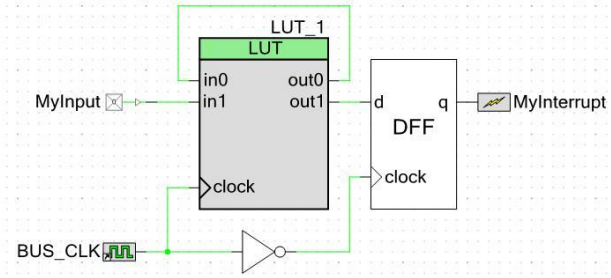
```
Error: mpr.M0096: The UDB Clock/Enable components may only drive clocks in UDB content and
\UDBClkEn_1:udbclkenable\ is connected to clock_in on ... which is not a clock input to the
UDB. (App=cydsfit)
```

反転されたクロックは可能です。

このトピックは通常、図 25 に示す反転クロックの使用に適用されません。なぜかというと、各 UDB には、非反転または反転クロックの使用を可能にするクロック制御モジュールがあり、インバーター シンボルは単に UDB にその反転クロック入力を使用させます。

反転クロックを使用すると、クロックの両エッジが使用されるため、クロックが減速することがあります。しかし一方では、クロッキングが 2 倍の周波数で行われるため、タイミング問題が多くなります。

図 25. 反転クロックによるデザイン例

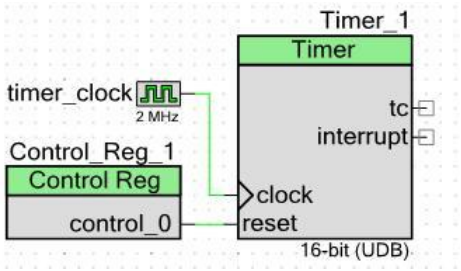


トピック 3: 制御レジスタ

12 ページの「UDB の制御レジスタおよびステータス レジスタ」で説明したように、ファームウェアがデジタル システムと通信する標準的な方法は制御レジスタを使用することです。各 UDB には 8 ビットの制御レジスタがあります。24 の UDB では 192 ビットの制御が使用可能です。制御レジスタ出力は、DSI を介してデジタル システムの任意の望ましい点に接続することができます。

ただし、STA 警告につながる制御レジスタのクロッキングの注意事項がいくつかあります。たとえば図 26 では、制御レジスタは UDB ベース タイマーのリセット入力を駆動します。すぐに分からないのは、制御レジスタはデフォルトでバス クロックによってクロック供給されることです。高周波数バス クロックのため、バス クロックとタイマー クロック間にセットアップ時間違反が発生することがあります。

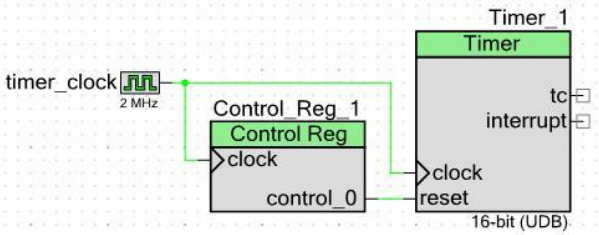
図 26. 制御レジスタの STA エラー




```
Warning-1366: Setup time violation found in
a path from clock (CyBUS_CLK) to clock
(timer_clock).
```

これは、図 27 に示すように制御レジスタを Sync モードに移行させることで簡単に修正できます。

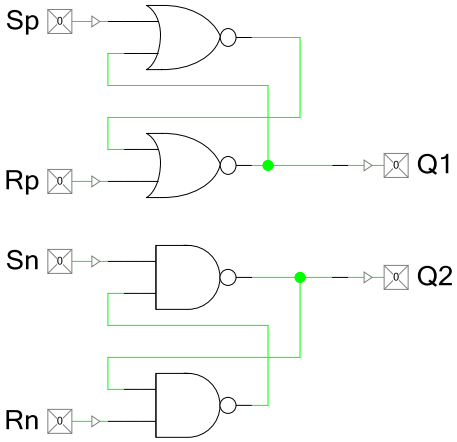
図 27. 制御レジスタの Sync モード



トピック 4: ラッチを作らない

図 28 は典型的なセット-リセット (S-R) ラッチ デザインを示します。(上の回路では、S と R はアクティブ HIGH で、下の回路では、S と R はアクティブ LOW です。) ラッチはレジスタや DFF に似ていますが、より不安定で、すなわち、入力上のグリッチパルスで出力が予想外に変化することがあります。また、両入力が同時にアクティブになると、ラッチが不安定な状態に入ります。ラッチの代わりにレジスタを使用したほうが良いです。

図 28. 典型的な S-R ラッチ デザイン



上記のデザインを実現しようとすると、PSoC Creator は以下の警告を生成します。

```
Warning-1361: The design contains a
combinational loop. Check the design for
unintentional latches. Breaking the loop
at ...
```

つまり、STA はその解析にループを含まないということです。

図 28 に示したデザインは別の問題があります。これらのデザインは同期化されず (図 3 と図 4 前後の説明を参照)、ラッチごとに 2 つの UDB PLD マクロセルを使用します。より良い方法は、図 29 に示すように PSoC Creator の SRFF コンポーネントを使用します。このデザインでは、ラッチごとに 1 つのマクロセルのみを使用します。

図 29. 同期化された S-R ラッチ コンポーネント

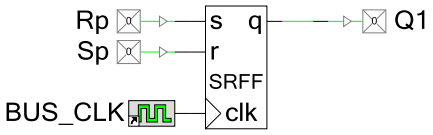
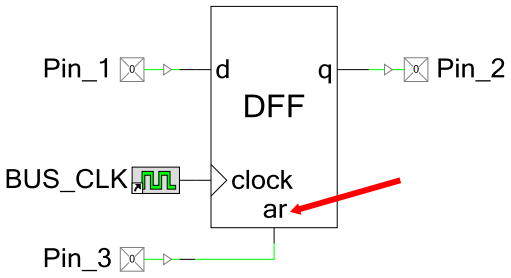


図 29 に示したラッチは LUT コンポーネントを用いて設計することができます (図 5 を参照してください)。これはまた、DFF コンポーネントを用いて設計することもできます。PSoC Creator の DFF コンポーネントは図 30 に示すようにリセットまたはプリセットに設定できます。

図 30. リセットに設定された DFF

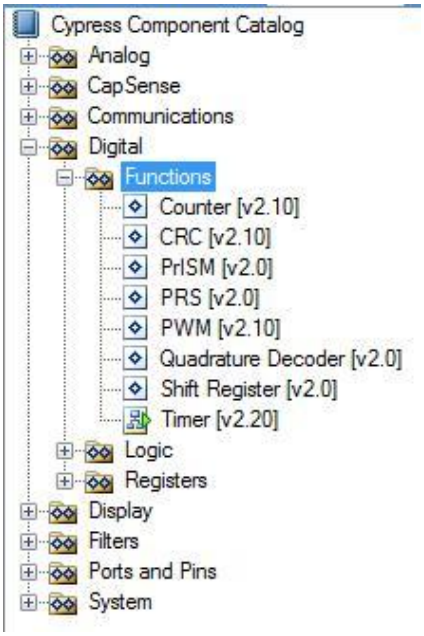


コンフィギュレーション ダイアログ ボックスには、非同期と同期、プリセットとリセットのオプションがあります。非同期入力はアサートされるとすぐに有効になり、メタスタビリティ問題を発生させることがあります。6 ページの「メタスタビリティおよびレジスタ タイミング」の回復時間と除去時間の説明を参照してください。同期入力は次のクロック エッジで有効になるため、その動作が予測可能です。詳細については、DFF コンポーネント データシートを参照してください。

トピック 5: デジタル機能コンポーネントを使用

PSoC Creator によって生成されたほとんどの STA 警告は、図 31 に示すコンポーネント カタログのデジタル機能のフォルダー内のコンポーネントの使用に関連しています。これらのコンポーネントはすべてがクロック入力を持ってるというユニークさを持っています。

図 31. デジタル機能コンポーネント



UDB のデータパスや PLD およびその他のブロックは複数のソースからクロック供給されます。デジタル機能コンポーネントはこの機能を利用し、バス クロックの反転であるそれらのクロック入力と同期化されています。これは「Sync コンポーネントにより」で説明したように、非同期化信号が他の入力に接続すると、STA 警告を発生させることがあります 。14 ページの図 21 と図 22 を参照してください。

固定機能ブロックがクロックを持ち、カウンター、タイマー、PWM コンポーネントでサポートされていますが、このトピックは固定機能タイマー ブロックに適用しません。詳細については、これらのコンポーネントの TRM またはデータシートを参照してください。

トピック 6: ピンとのインターフェース

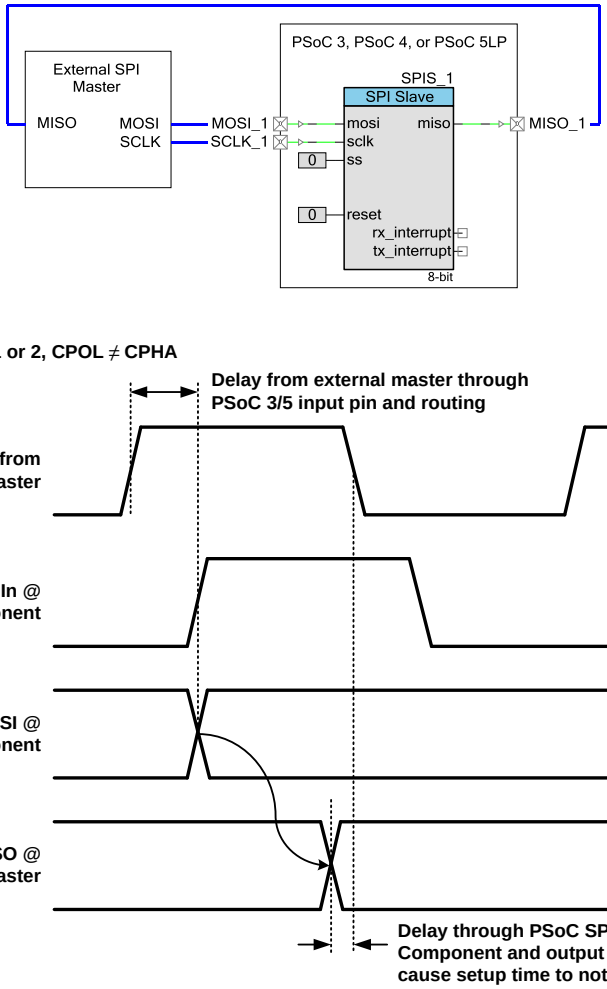
PSoC 3 および PSoC 5LP の I/O ピンは、CPU、DMA、アナログ、LCD など PSoC 内の多くのソース ／シンクに接続できます。またこれらは DSI にも接続できるため、信号はデジタル構造にわたって配線することができます。これにより、ピンは他のすべてのコンポーネントと同じように同期化問題を抱えています。ピン入力と出力は互いに独立しており、同期化される場合もされない場合もあります。

11 ページの「ピン」で説明したように、ピン入力信号はバス クロックと二重同期化することができます。デフォルトでは、同期化が有効になっていますが、同期化を無効にする必要がある特別な場合もあります。

これを最もよく示している例は、図 32 に示す PSoC Creator の SPI スレーブ (SPIS) コンポーネントです。プリント基板の配線および PSoC ピン、DSI ルーティング、UDB により生じる遅延はマスターの SCLK と MOSI 出力からその MISO 入力までの全体的な遅延を発生させます。これで、マスターの MISO 入力が正しく読み出されない可能性があります。

PSoC Creator の SPI マスター (SPIM) コンポーネントにも同様な問題が存在しています。

図 32. SPIS コンポーネントの SPI タイミング



PSoC Creator の SPIS コンポーネントでは遅延の問題はあまりありません。一方、MISO、SCLK および MOSI ピンを同期化すると、かなり多い遅延が追加されます。これは特に入力ピンに当てはまります 。11 ページの「ピン」で説明したように、入力ピンは二重同期化されており、多くの遅延を発生させることがあります。一般的には、SPI ピンの同期化を無効にする必要があります。

すべての PSoC Creator SPI コンポーネントは、同期化がすでに無効にされたピン コンポーネントを含むマクロとして用意されています。マクロを使用しない、すなわち、SPIM または SPIS コンポーネントを直接使用する場合、これらのコンポーネントに接続されたピンは同期化が無効になっていることを確認してください。

トピック 7: 固定機能ブロックとのインターフェース

図 1 に示すように、PSoC 3 および PSoC 5LP は、UDB システムと固定機能デジタルとアナログ サブシステム間のインターフェースを備えています。図 33 は、固定機能ブロックと UDB ベースのコンポーネント (この場合はカウンター) 間のインタフェースのいくつかの例を示します。タイマー コンポーネントは固定機能タイマー ／カウンター ／PWM (TCPWM) ブロックをベースとしていることに注意してください。

この場合、PSoC Creator の STA 機能はアナログとデジタルサブシステム間で配線される信号を含みません。更に、発生する可能性のあるタイミング問題で、コンパレータと ADC 出力はカウンター入力で適切に取り込まれないことがあります。

また、以下のような、固定機能ブロック ブロックから UDB クロックまでの非同期パスの STA 警告が発生する可能性があります。

```
Asynchronous path(s) exist from
"Clock_1(fixed-function)" to "Clock_1".
```

理由は、クロッキング ネットワークと DSI 間のタイミング差異のためです。詳細については、図 12 および 10 ページの説明を参照してください。一般的には、図 34 に示すように、UDB ベース コンポーネントに接続された固定機能ブロック出力に Sync コンポーネントを追加する必要があります。

図 33. 固定機能／UDB インターフェース

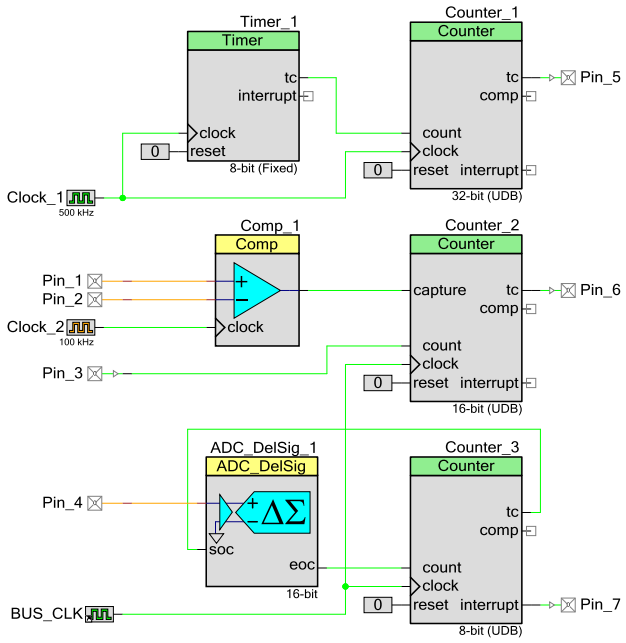
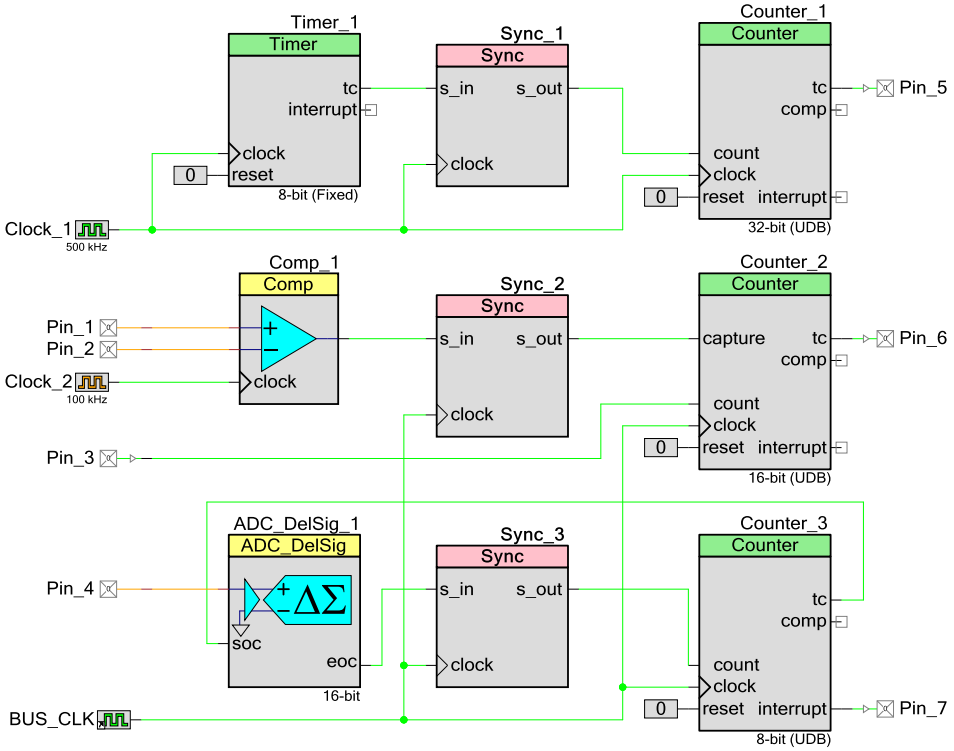


図 34. 固定機能ブロックが Sync コンポーネントとインターフェース



PSoC Creator STA レポートの使用

前述したように、PSoC Creator の STA 機能はデジタル デザインを評価し、各クロックの最大許容周波数および潜在的なタイミング問題が存在しているかについてのレポートを生成します。プロジェクトの構築中に STA 警告を受けた場合、警告を理解して解除するために STA レポートをレビューする必要があることがあります。

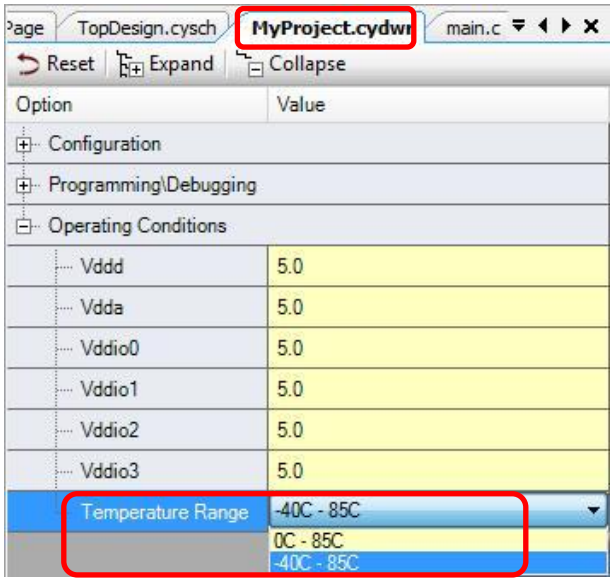
本節では、STA レポート使用の方法およびベストプラクティスについて説明します。

注 : PSoC 4 ベースのデザインで STA 警告を生成する可能性もありますが、PSoC 3 および PSoC 5LP は複雑なデジタルとクロッキング システムのため、STA 警告を生成する可能性がより高いです。そのため、本節は PSoC 3 および PSoC 5LP ベースのデザインに向いています。

STA の温度条件を設定

PSoC デバイスは広範囲の電源電圧と温度にわたって動作し、TDR 制約は全温度範囲の仕様を考慮に入れて選択されます。しかし、デザインが制御された温度範囲にわてって実行される場合、TDR 制約は満たしやすくなり、柔軟な配線を実現することが容易になります。図 35 に示すように、デザインの温度範囲は PSoC Creator プロジェクト システムの 1 つの設定です。

図 35. デバイスの温度範囲の設定

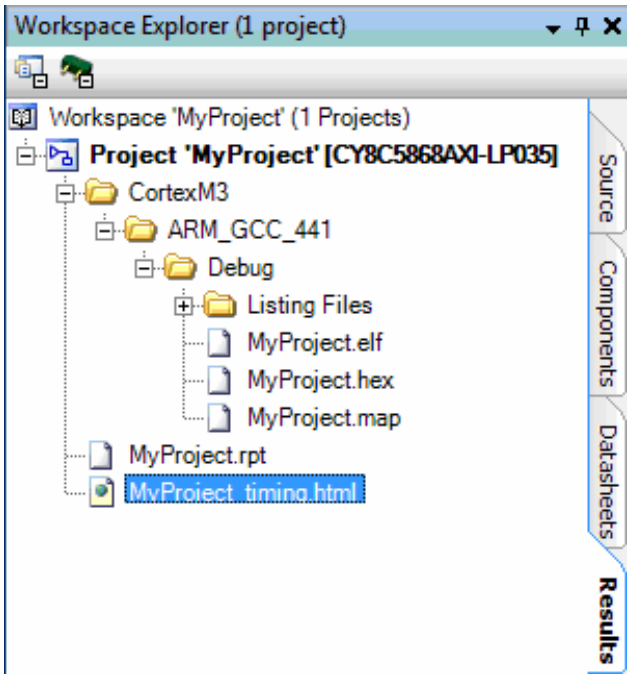


プロジェクトに狭い温度範囲を選択すれば、TDR ツールはタイミング準拠の配線ソリューションを見出すことが容易になります。これは、プロジェクトの構築中に STA 警告を受ける可能性に影響します。

STA レポートを探す

STA レポートは自動生成された HTML ファイルであり、図 36 に示すように PSoC Creator プロジェクトの「Results」タブで見つけます。

図 36. PSoC Creator プロジェクトの STA レポート ファイル



STA レポートの要素

タイミング違反およびクロックのまとめのほか、STA レポートにはクロックとその他の信号間のセットアップ時間、ホールド時間、回復時間、除去時間の詳細な解析を示すいくつかのセクションがあります。STA レポートのセクションの詳細情報については、PSoC Creator のヘルプ記事「静的タイミング解析」を参照してください。

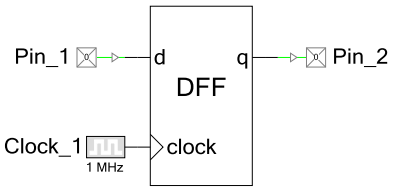
以下の例は、デザインと STA レポート データとの関係を詳しく示します。

例 1

まず、22 ページの図 37 の簡単な例で始めましょう。この例では、すべてのコンポーネントとクロック設定は PSoC 3 および PSoC 5LP のデフォルトです。

- マスター クロックとバス クロックは 24MHz
- Pin_1 の入力同期化がオン (11 ページの「ピン」を参照)
- Pin_2 の出力同期化がオフ
- Clock_1 はマスター クロックと同期

図 37. 例 1 の STA レポート



Static Timing Analysis

Project : Fig36
Build Time : 07/22/13 17:10:53
Device : CY8C3866AXI-040
Temperature : -40C - 85/125C
Vdda : 5.00
Vddd : 5.00
Vio0 : 5.00
Vio1 : 5.00
Vio2 : 5.00
Vio3 : 5.00
Voltage : 5.0
Vusb : 5.00

Expand All | Collapse All | Show All Paths | Hide All Paths

+ Timing Violation Section

No Timing Violations

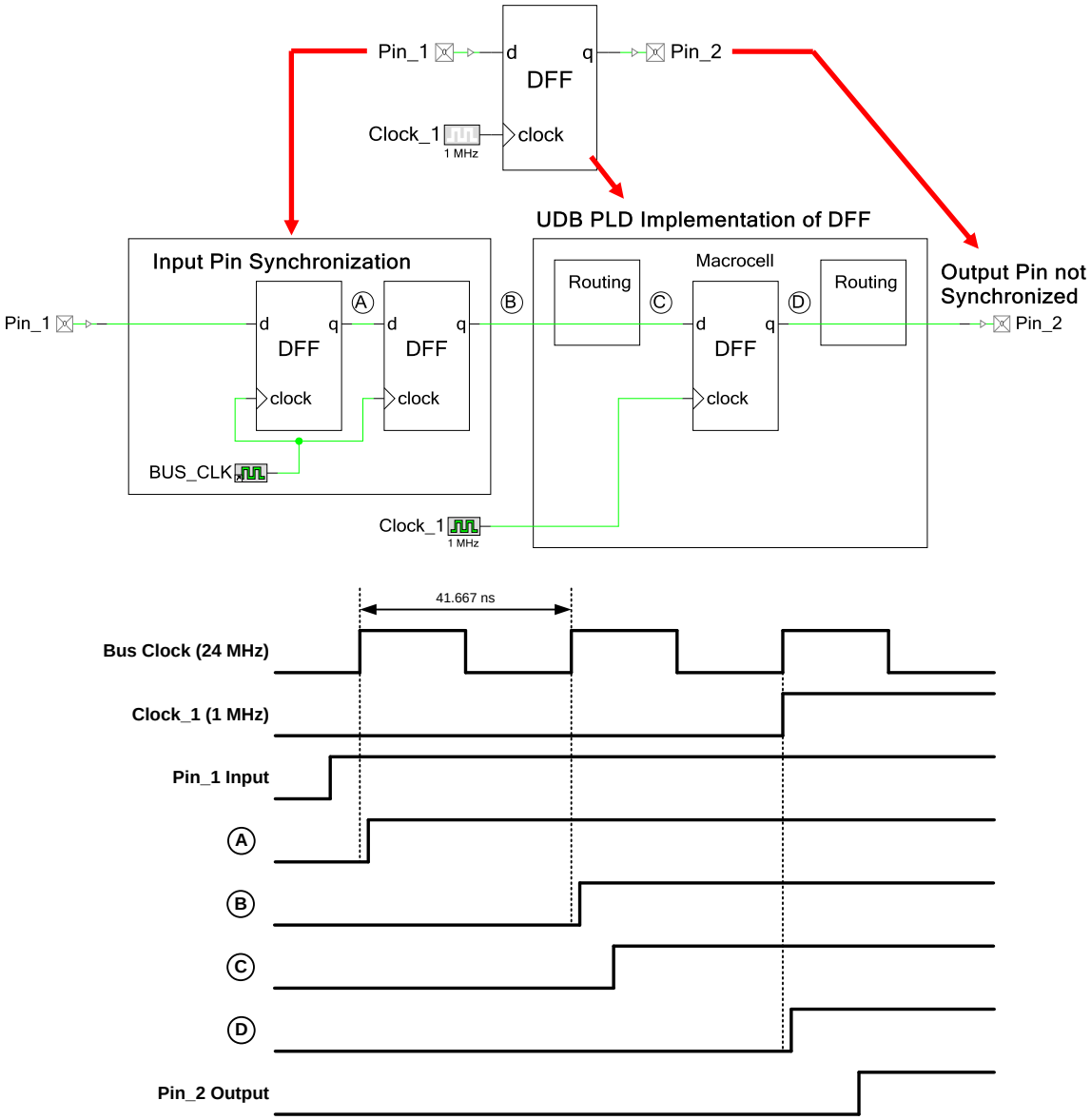
+ Clock Summary Section

Clock	Domain	Nominal Frequency	Required Frequency	Maximum Frequency	Violation
CyILO	CyILO	1.000 kHz	1.000 kHz	N/A	
CyIMO	CyIMO	3.000 MHz	3.000 MHz	N/A	
CyMASTER_CLK	CyMASTER_CLK	24.000 MHz	24.000 MHz	N/A	
Clock_1	CyMASTER_CLK	1.000 MHz	1.000 MHz	69.677 MHz	
CyBUS_CLK	CyMASTER_CLK	24.000 MHz	24.000 MHz	69.677 MHz	
CyPLL_OUT	CyPLL_OUT	24.000 MHz	24.000 MHz	N/A	

STA レポートが開かれると、タイミング違反、クロックの一覧およびクロック速度違反のまとめを示します。この例は違反がありません。レポートに示すように、バス クロックと Clock_1 はクロック速度違反なしに最大 69.677 MHz の周波数 (デバイスの周波数限界よりも高いことがある) を達成できます。

STA レポートのタイミング詳細を検討する前に、回路図上のデザインと PSoC デバイス内で実際に実行されるものとの関係を理解することが重要です。23 ページの図 38 では、入力ピン同期化ブロックは、入力ピン同期化がオンになっているため、入力ピン信号はバス クロックによってクロック供給される 2 本のレジスタを通過することを示します。その後、信号はピンから DSI を介して、回路図で DFF を実装する UDB マクロセルに配線されます。最後に、マクロセル出力は DSI を介して、同期化されていない出力ピンに配線されます。

図 38. 例 1 のハードウェア実装



STA レポートの遅延の詳細については、「Expand All」をクリックしてください。 (図 37 を参照)。24 ページの図 39 は拡張されたレポートおよび詳細なサブセクションの数字の重要性を示します。

- **Setup Subsection** の「Path Delay Requirement」: 41.6667ns。24MHz バス クロックに対応します。
- 同じサブセクションの表の「Delay」: 14.352ns。これは、配線およびゲートに起因した遅延により生じる Pin_1 出力からマクロセル入力までの計算された遅延です。
- 同表の「Slack」: 27.315ns、すなわち[41.667－14.352]の結果です。これはマクロセルの計算されたセットアップ時間です。
- **Clock to Output Section** の「Delay」: 22.883ns。これは、配線に起因した遅延により生じるマクロセル出力から Pin_2 までの計算された遅延です。

「Show All Paths」をクリックすると、さらなる拡張が表示されますが、通常は不要です。図 39 では、**Hold Subsection** は折りたたまれています 。ホールド時間違反の詳細については、35 ページの「ホールド時間、回復時間および除去時間の違反」を参照してください。

図 39. 拡張された STA レポート 1 および対応するタイミング

Static Timing Analysis

Project : MyProject
Build Time : 07/23/13 15:55:08
Device : CY8C3866AXI-040
Temperature : -40C - 85/125C
Vdda : 5.00
Vddd : 5.00
Vio0 : 5.00
Vio1 : 5.00
Vio2 : 5.00
Vio3 : 5.00
Voltage : 5.0
Vusb : 5.00

Expand All | Collapse All | Show All Paths | Hide All Paths

– Timing Violation Section

No Timing Violations

– Clock Summary Section

Clock	Domain	Nominal Frequency	Required Frequency	Maximum Frequency	Violation
CyILO	CyILO	1.000 kHz	1.000 kHz	N/A	
CyIMO	CyIMO	3.000 MHz	3.000 MHz	N/A	
CyMASTER_CLK	CyMASTER_CLK	24.000 MHz	24.000 MHz	N/A	
Clock_1	CyMASTER_CLK	1.000 MHz	1.000 MHz	69.677 MHz	
CyBUS_CLK	CyMASTER_CLK	24.000 MHz	24.000 MHz	69.677 MHz	
CyPLL_OUT	CyPLL_OUT	24.000 MHz	24.000 MHz	N/A	

– Register to Register Section

– Setup Subsection

– Source Clock : CyBUS_CLK : Positive edge(Required Frequency 24 MHz)

– Destination Clock : Clock_1 : Positive edge(Required Frequency 1 MHz)

Path Delay Requirement : 41.6667ns(24 MHz)
Affects clock : CyMASTER_CLK

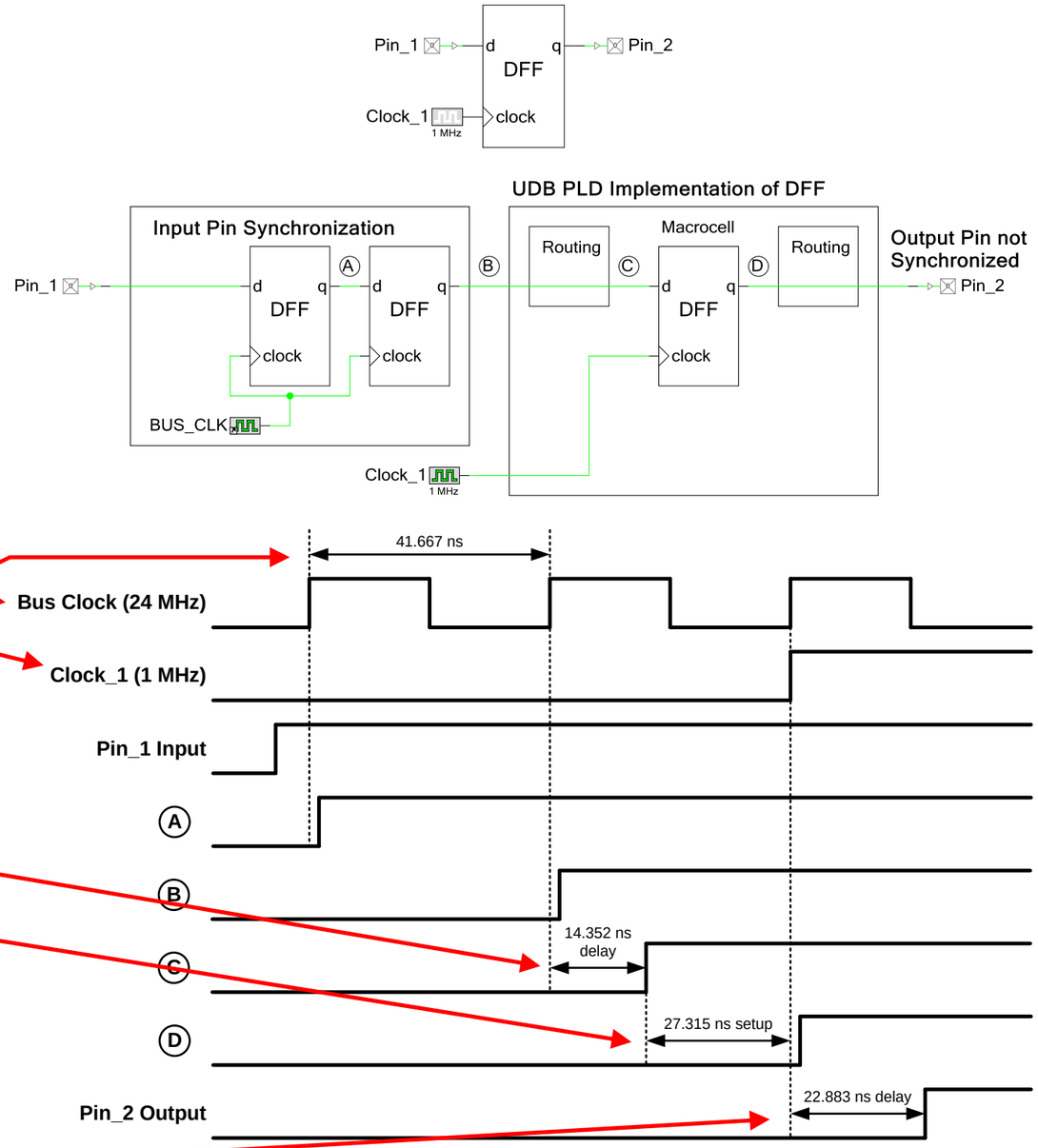
Source	Destination	FMax	Delay (ns)	Slack (ns)	Violation
Pin_1(0)/fb	cydff_1/main_0	69.677 MHz	14.352	27.315	

+ Hold Subsection

– Clock To Output Section

– Clock_1

Source	Destination	Delay (ns)
cydff_1/q	Pin_2(0)_PAD	22.883



レポートの表の任意の行をクリックすると、その行は拡張されて詳細情報を表示します。図 40 では、**Clock to Output Section** は拡張されて Clock_1 から DFF を介して Pin_2 までの遅延の各部を示しています。

拡張されたレポートは遅延の大半 (16.2ns) はピンそれ自体により生ずることを示します。遅延の小さい量 (5.4ns) はマクロセル出力から DSI を介してピンに配線されることにより生じ、残りの非常に小さい量は DFF それ自体により生じます。

図 40. 例 1 の拡張された「Clock to Output Section」

- Clock_1

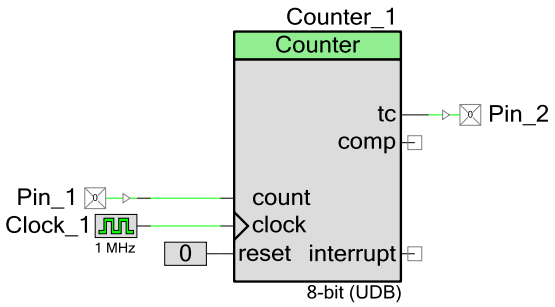
Source	Destination	Delay (ns)
cydff_1/q	Pin_2(0)_PAD	22.883

Type	Location	Fanout	Instance/Net	Source	Dest	Delay (ns)
macrocell1	U(3,4)	1	cydff_1	cydff_1/clock_0	cydff_1/q	1.250
Route		1	cydff_1	cydff_1/q	Pin_2(0)/pin_input	5.433
iocell	P0[1]	1	Pin_2(0)	Pin_2(0)/pin_input	Pin_2(0)/pad_out	16.200
Route		1	Pin_2(0)_PAD	Pin_2(0)/pad_out	Pin_2(0)_PAD	0.000
Clock					Clock path delay	0.000

例 2

図 41 に示すように、例 1 の DFF を UDB ベースの 8 ビット カウンターに置き換えます。

図 41. 例 2 の STA レポート



生成された STA レポート ファイルの一部を図 42 に示します。まだタイミング違反が発生していませんが、**Clock Summary Section** はバス クロックのマージンが小さくなり、最大周波数が約 35MHz に低減されたことを示します。

図 42. 例 2 の STA レポート

Static Timing Analysis

Project : MyProject
Build Time : 07/23/13 16:15:15
Device : CY8C3866AXI-040
Temperature : -40C - 85/125C
Vdda : 5.00
Vddd : 5.00
Vio0 : 5.00
Vio1 : 5.00
Vio2 : 5.00
Vio3 : 5.00
Voltage : 5.0
Vusb : 5.00

Expand All | Collapse All | Show All Paths | Hide All Paths

- Timing Violation Section

No Timing Violations

- Clock Summary Section

Clock	Domain	Nominal Frequency	Required Frequency	Maximum Frequency	Violation
CyILO	CyILO	1.000 kHz	1.000 kHz	N/A	
CyIMO	CyIMO	3.000 MHz	3.000 MHz	N/A	
CyMASTER_CLK	CyMASTER_CLK	24.000 MHz	24.000 MHz	N/A	
Clock_1	CyMASTER_CLK	1.000 MHz	1.000 MHz	35.688 MHz	
CyBUS_CLK	CyMASTER_CLK	24.000 MHz	24.000 MHz	35.688 MHz	
CyPLL_OUT	CyPLL_OUT	24.000 MHz	24.000 MHz	N/A	

+ Register to Register Section

+ Clock To Output Section

理由を理解するために、26 ページの図 43 内の **Register to Register Section、Setup Subsection** の詳細レポートを参照してください。

図 43. 例 2 の拡張された「Setup Subsection」

- **Register to Register Section**

  - **Setup Subsection**

    - **Source Clock : Clock_1 : Positive edge(Required Frequency 1 MHz)**

      - **Destination Clock : Clock_1 : Positive edge(Required Frequency 1 MHz)**

Path Delay Requirement : 1000ns(1 MHz)

Source	Destination	FMax	Delay (ns)	Slack (ns)	Violation
\Counter_1:CounterUDB:sCTRLReg:SyncCtl:ctrlreg\/control_7	\Counter_1:CounterUDB:sC8:counterdp:u0\/cs_addr_1	45.304 MHz	22.073	977.927	
\Counter_1:CounterUDB:count_stored_i\/q	\Counter_1:CounterUDB:sC8:counterdp:u0\/cs_addr_1	48.256 MHz	20.723	979.277	
\Counter_1:CounterUDB:sC8:counterdp:u0\/z0_comb	\Counter_1:CounterUDB:sC8:counterdp:u0\/cs_addr_0	51.454 MHz	19.435	980.565	
\Counter_1:CounterUDB:sC8:counterdp:u0\/z0_comb	\Counter_1:CounterUDB:sSTSReg:rstSts:stsreg\/status_3	61.501 MHz	16.260	983.740	
Net_22/q	\Counter_1:CounterUDB:sSTSReg:rstSts:stsreg\/status_3	70.507 MHz	14.183	985.817	
\Counter_1:CounterUDB:prevCompare\/q	\Counter_1:CounterUDB:sSTSReg:rstSts:stsreg\/status_0	92.902 MHz	10.764	989.236	
\Counter_1:CounterUDB:sC8:counterdp:u0\/z0_comb	Net_22/main_0	94.073 MHz	10.630	989.370	
\Counter_1:CounterUDB:sC8:counterdp:u0\/z0_comb	\Counter_1:CounterUDB:sSTSReg:rstSts:stsreg\/status_1	99.661 MHz	10.034	989.966	

    - **Source Clock : CyBUS_CLK : Positive edge(Required Frequency 24 MHz)**

      - **Destination Clock : Clock_1 : Positive edge(Required Frequency 1 MHz)**

Path Delay Requirement : 41.6667ns(24 MHz)
Affects clock : CyMASTER_CLK

Source	Destination	FMax	Delay (ns)	Slack (ns)	Violation
Pin_1(0)/fb	\Counter_1:CounterUDB:sC8:counterdp:u0\/cs_addr_1	35.688 MHz	28.021	13.646	
Pin_1(0)/fb	\Counter_1:CounterUDB:count_stored_i\/main_0	69.677 MHz	14.352	27.315	

カウンター コンポーネントは UDB データパスと PLD を組み合わせて構築されています。図 43 では、**Source Clock: Clock_1** 表は、カウンター内の UDB 要素がカウンターのクロック入力 (この場合は Clock_1) によってクロック供給されることを示します。また、Clock_1 は 1MHz のみであるため、1µs のスラック、すなわち本当に十分なマージンがあります。

詳細情報は **Source Clock: CyBUS_CLK** 表に記載されています。この表の「Path Delay Requirement」は 24MHz バス クロックの周期であることに注意してください。理由は、14 ページの「複数のクロック」を参照してください。

この表は、バス クロックと同期化されている入力ピンはカウンター内の 2 つの UDB 要素に接続されることを示します。その中の 1 つは、27ns 以上の遅延を持って、スラックを 14ns 以下に低減します。これは、バス クロックが 35MHz に制限される理由です。

最後に、カウンター コンポーネントを 8 ビットから 32 ビットに変更し、デモの目的のためにバス クロック周波数を 28MHz に増加させることができます。再構築後に、図 44 に示すようにセットアップ時間違反が発生します。

図 44. 32 ビット カウンターのタイミング違反

- **Timing Violation Section**

Note: If your design will only ever run at typical room temperatures, selecting the narrower temperatur
range in the system DWR for your application helps the tool to find timing-compliant routing solutions.

Violation	Source Clock	Destination Clock	Slack(ns)
Setup			
	CyBUS_CLK	Clock_1	-2.705

- **Clock Summary Section**

Clock	Domain	Nominal Frequency	Required Frequency	Maximum Frequency	Violation
CyILO	CyILO	1.000 kHz	1.000 kHz	N/A	
CyIMO	CyIMO	3.000 MHz	3.000 MHz	N/A	
CyMASTER_CLK	CyMASTER_CLK	28.000 MHz	28.000 MHz	N/A	
Clock_1	CyMASTER_CLK	1.000 MHz	1.000 MHz	26.029 MHz	
CyBUS_CLK	CyMASTER_CLK	28.000 MHz	28.000 MHz	26.029 MHz	Frequency
CyPLL_OUT	CyPLL_OUT	28.000 MHz	28.000 MHz	N/A	

違反の理由は **Clock Summary Section** でより明らかになります。ここには、コンポーネントのバス クロック周波数 28MHz が高すぎることが示されます。

違反の詳細は 27 ページの PAGEREF _Ref362333928 \h 図 45 に示されます。32 ビット カウンターは、チェーンにされた 4 個の UDB データパスからなります。したがって、バス クロックと同期化された入力ピンは UDB 要素内の多くのデスティネーション点に接続されています。デスティネーション点の中の 1 つは配線およびその他の遅延が 38.419ns であり、バス クロックの周期 35.7143ns より長いため、セットアップ時間違反を発生させます。Clock_1 周波数を変更することは役に立ちません。この問題を修正するために、バス クロック周波数を低下させるか、またはピン 1 を Clock_1 と同期化させます。

図 45. 32 ビット カウンターのタイミング違反の詳細

– Source Clock : CyBUS_CLK : Positive edge(Required Frequency 28 MHz)

– Destination Clock : Clock_1 : Positive edge(Required Frequency 1 MHz)

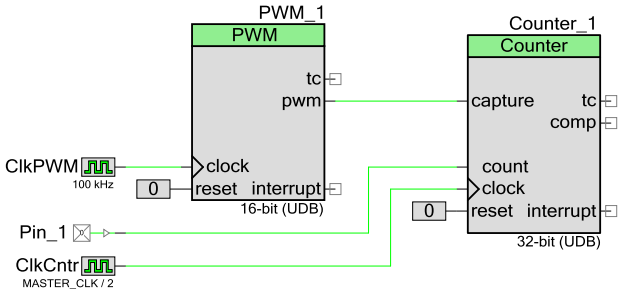
Path Delay Requirement : 35.7143ns(28 MHz)
Affects clock : CyMASTER_CLK

Source	Destination	FMax	Delay (ns)	Slack (ns)	Violation
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	26.029 MHz	38.419	-2.705	SETUP
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u2\/ci	28.483 MHz	35.109	0.605	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	28.484 MHz	35.108	0.606	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	30.431 MHz	32.861	2.853	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u1\/ci	31.448 MHz	31.799	3.915	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u2\/ci	31.449 MHz	31.798	3.916	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u2\/cs_addr_1	33.805 MHz	29.581	6.133	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u3\/cs_addr_1	33.807 MHz	29.580	6.134	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u0\/cs_addr_1	35.064 MHz	28.519	7.195	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u1\/cs_addr_1	35.066 MHz	28.518	7.196	

STA レポートを使用して警告を解除

最後の演習では、上記の節からの技術を使用して適度に複雑な例で一連の STA 警告を理解し、解決します。まず、図 20 を検討しましょう。これは図 46 とほぼ同じですが、ClkCntr はマスター クロックを 2 で割った値です 。

図 46. 図 20 からのマルチクロックの例



このデザインでは、両クロックがマスター クロックと同期化されており、カウンター クロック ClkCntr がマスター クロックを 2 で割った値にセットされたことに注意してください。マスター クロック周波数が十分高ければ (この場合は 60 MHz)、2 つの STA セットアップ時間違反が検出されます。

```
Warning-1366: Setup time violation found in
a path from clock (ClkCntr) to
clock(ClkCntr)

Warning-1366: Setup time violation found in
a path from clock (CyBUS_CLK) to clock
(ClkCntr)
```

図 47 は STA レポートの「Timing Violation Section」に同じ警告を示します。

図 47. 図 46 の STA レポート

Static Timing Analysis

Project : Fig19
Build Time : 07/23/13 09:27:41
Device : CY8C3866AXI-040
Temperature : -40C - 85/125C
Vdda : 5.00
Vddd : 5.00
Vio0 : 5.00
Vio1 : 5.00
Vio2 : 5.00
Vio3 : 5.00
Voltage : 5.0
Vusb : 5.00

Expand All | Collapse All | Show All Paths | Hide All Paths

- Timing Violation Section

Note: If your design will only ever run at typical room temperatures, selecting the narrower temperature range in the system DWR for your application helps the tool to find timing-compliant routing solutions.

Violation	Source Clock	Destination Clock	Slack(ns)
Setup			
	ClkCntr	ClkCntr	-2.533
	CyBUS_CLK	ClkCntr	-14.795

- Clock Summary Section

Clock	Domain	Nominal Frequency	Required Frequency	Maximum Frequency	Violation
CyILO	CyILO	1.000 kHz	1.000 kHz	N/A	
CyIMO	CyIMO	3.000 MHz	3.000 MHz	N/A	
CyMASTER_CLK	CyMASTER_CLK	60.000 MHz	60.000 MHz	70.952 MHz	
ClkCntr	CyMASTER_CLK	30.000 MHz	30.000 MHz	27.882 MHz	Frequency
Clk_PWM	CyMASTER_CLK	100.000 kHz	100.000 kHz	42.411 MHz	
CyBUS_CLK	CyMASTER_CLK	60.000 MHz	60.000 MHz	31.784 MHz	Frequency
CyPLL_OUT	CyPLL_OUT	60.000 MHz	60.000 MHz	N/A	

警告を解除するために、まず (ソース クロックとデスティネーション クロック両方が ClkCntr である) カウンター クロックのセットアップ違反を検討します。このためには、**Timing Violation Section** で ClkCntr から ClkCntr までの行をクリックします。すると、図 48 に示すようにレポートのさらなる詳細が表示されます。

図 48. ClkCntr から ClkCntr までの STA セットアップ時間違反

- **Register to Register Section**
  - **Setup Subsection**
    - **Source Clock : ClkCntr : Positive edge(Required Frequency 30 MHz)**
      - **Destination Clock : ClkCntr : Positive edge(Required Frequency 30 MHz)**

Path Delay Requirement : 33.3333ns(30 MHz)

Source	Destination	FMax	Delay (ns)	Slack (ns)	Violation
\Counter_1:CounterUDB:sC32:counterdp:u0\/z0	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	27.882 MHz	35.866	-2.533	SETUP
\Counter_1:CounterUDB:sC32:counterdp:u1\/z0	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	29.039 MHz	34.436	-1.103	SETUP
\Counter_1:CounterUDB:sC32:counterdp:u2\/z0	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	30.298 MHz	33.006	0.327	
\Counter_1:CounterUDB:sCTRLReg:SyncCtl:ctrlreg\/control_7	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	30.714 MHz	32.558	0.775	
\Counter_1:CounterUDB:sC32:counterdp:u0\/z0	\Counter_1:CounterUDB:sC32:counterdp:u2\/ci	30.716 MHz	32.556	0.777	
\Counter_1:CounterUDB:sC32:counterdp:u3\/z0_comb	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	31.670 MHz	31.576	1.757	
\Counter_1:CounterUDB:sC32:counterdp:u0\/z0	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	31.742 MHz	31.504	1.829	
\Counter_1:CounterUDB:count_stored_i\/q	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	32.050 MHz	31.201	2.132	
\Counter_1:CounterUDB:sC32:counterdp:u1\/z0	\Counter_1:CounterUDB:sC32:counterdp:u2\/ci	32.127 MHz	31.126	2.207	
\Counter_1:CounterUDB:sC32:counterdp:u1\/z0	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	33.251 MHz	30.074	3.259	

カウンターが 32 ビットであるため、実装するには 4 個の UDB データパスが必要です。図 48 は、いくつかの UDB による遅延は 30MHz カウンター クロック (ClkCntr) に対しては長すぎ、ClkCntr を 27.882MHz 以下に低下させる必要があることを示します。ClkCntr がマスター クロックの半分であるため、マスター クロックを 55MHz に低下させることで、ClkCntr は 27.5MHz となります。

これにより最初の警告を解除しますが、もう 1 つのセットアップ時間違反があります。

```
Warning-1366: Setup time violation found in a path from clock (CyBUS_CLK) to clock (ClkCntr)
```

この警告を解除するために、STA レポートの CyBUS_CLK から ClkCntr までのセクションを検討します。**Timing Violation Section** 内の対応する行をクリックします。詳細は図 49 に示されます。

図 49. 55MHz マスター クロックの STA セットアップ時間違反

- **Register to Register Section**
  - **Setup Subsection**
    - **+ Source Clock : ClkCntr : Positive edge(Required Frequency 27.5 MHz)**
    - **+ Source Clock : Clk_PWM : Positive edge(Required Frequency 100 kHz)**
    - **- Source Clock : CyBUS_CLK : Positive edge(Required Frequency 55 MHz)**
      - **Destination Clock : ClkCntr : Positive edge(Required Frequency 27.5 MHz)**

Path Delay Requirement : 18.1818ns(55 MHz)
Affects clock : CyMASTER_CLK

Source	Destination	FMax	Delay (ns)	Slack (ns)	Violation
Pin_1(0)_SYNC/out	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	31.758 MHz	31.488	-13.306	SETUP
Pin_1(0)_SYNC/out	\Counter_1:CounterUDB:sC32:counterdp:u2\/ci	35.489 MHz	28.178	-9.996	SETUP
Pin_1(0)_SYNC/out	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	35.528 MHz	28.147	-9.965	SETUP
Pin_1(0)_SYNC/out	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	39.005 MHz	25.638	-7.456	SETUP
Pin_1(0)_SYNC/out	\Counter_1:CounterUDB:sC32:counterdp:u1\/ci	40.212 MHz	24.868	-6.686	SETUP
Pin_1(0)_SYNC/out	\Counter_1:CounterUDB:sC32:counterdp:u2\/ci	40.263 MHz	24.837	-6.655	SETUP
Pin_1(0)_SYNC/out	\Counter_1:CounterUDB:sC32:counterdp:u3\/cs_addr_1	44.431 MHz	22.507	-4.325	SETUP
Pin_1(0)_SYNC/out	\Counter_1:CounterUDB:sC32:counterdp:u2\/cs_addr_1	44.727 MHz	22.358	-4.176	SETUP
Pin_1(0)_SYNC/out	\Counter_1:CounterUDB:sC32:counterdp:u0\/cs_addr_1	46.322 MHz	21.588	-3.406	SETUP
Pin_1(0)_SYNC/out	\Counter_1:CounterUDB:sC32:counterdp:u1\/cs_addr_1	46.389 MHz	21.557	-3.375	SETUP

まず、「Path Delay Requirement」は ClkCntr ではなく、55MHz マスター クロックの周期であることに注意してください。理由は、14 ページの「複数のクロック」を参照してください。カウンターUDB までの配線により生じる遅延は Pin_1 に対しては長すぎます。2 つの解決策があります。

1. マスター クロック周波数を低下させます。全体的なシステム タイミングには高速なマスター クロックが必要ない場合、この方法は、ノイズおよび消費電力を減少できるため、推奨されています。図 49 は、マスター クロックは FMax の最小値 (31.758MHz) よりも低くする必要があることを示します。しかし、バスクロックを 33MHz より低くすると、異なる方法が Pin_1 に適用されます(11 ページの「ピン」を参照してください)。この方法は配線を変化させ、その結果、別のタイミング違反を発生させます。図 50 は、実際にマスター クロック周波数をさらに低下する (26.036MHz よりも低くする) 必要があることを示します。

図 50. 31MHz マスター クロックのセットアップ時間違反

- Register to Register Section

- Setup Subsection

+ Source Clock : ClkCntr : Positive edge(Required Frequency 15.5 MHz)

+ Source Clock : Clk_PWM : Positive edge(Required Frequency 100 kHz)

- Source Clock : CyBUS_CLK : Positive edge(Required Frequency 31 MHz)

- Destination Clock : ClkCntr : Positive edge(Required Frequency 15.5 MHz)

Path Delay Requirement : 32.2581ns(31 MHz)
Affects clock : CyMASTER_CLK

Source	Destination	FMax	Delay (ns)	Slack (ns)	Violation
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	26.036 MHz	38.409	-6.151	SETUP
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u2\/ci	28.491 MHz	35.099	-2.841	SETUP
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	28.492 MHz	35.098	-2.840	SETUP
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u3\/ci	30.440 MHz	32.851	-0.593	SETUP
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u1\/ci	31.457 MHz	31.789	0.469	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u2\/ci	31.458 MHz	31.788	0.470	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u2\/cs_addr_1	33.817 MHz	29.571	2.687	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u3\/cs_addr_1	33.818 MHz	29.570	2.688	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u0\/cs_addr_1	35.077 MHz	28.509	3.749	
Pin_1(0)/fb	\Counter_1:CounterUDB:sC32:counterdp:u1\/cs_addr_1	35.078 MHz	28.508	3.750	

その後 (PLL を 52MHz にセットし、2 で割って 26MHz マスター クロックを取得することで)、図 51 に示すようにどの STA も発生しません。

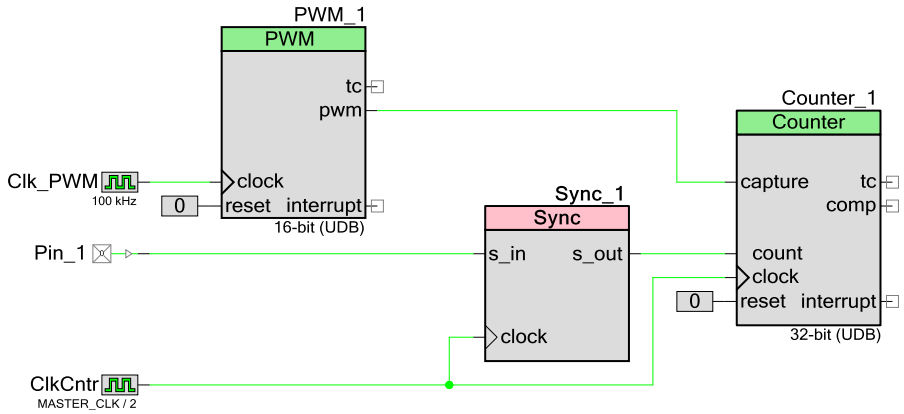
図 51. 23MHz マスター クロックでのクロックのまとめ

- Clock Summary Section

Clock	Domain	Nominal Frequency	Required Frequency	Maximum Frequency	Violation
CyILO	CyILO	1.000 kHz	1.000 kHz	N/A	
CyIMO	CyIMO	3.000 MHz	3.000 MHz	N/A	
CyMASTER_CLK	CyMASTER_CLK	26.000 MHz	26.000 MHz	76.225 MHz	
ClkCntr	CyMASTER_CLK	13.000 MHz	13.000 MHz	26.036 MHz	
Clk_PWM	CyMASTER_CLK	100.000 kHz	100.386 kHz	40.125 MHz	
CyBUS_CLK	CyMASTER_CLK	26.000 MHz	26.000 MHz	26.036 MHz	
CyPLL_OUT	CyPLL_OUT	52.000 MHz	52.000 MHz	N/A	

2. いつでもマスター クロック周波数を十分低い値に低下させることですべての STA 警告を解除することができます。ただし、高速なマスター クロックまたはバス クロックが必要な場合は、より良い解決策は PSoC Creator の Sync コンポーネント (この場合は Pin_1 を ClkCntr と同期化するため) を追加することです。図 52 を参照してください。

図 52. Sync コンポーネントが追加された、図 20 からのマルチクロックの例



これにより、STA レポートは図 53 に示すように、ClkCntr の最大周波数が 28MHz に達したことを示します。

図 53. Sync コンポーネントが追加された、26MHz マスター クロックでのクロックのまとめ

– Clock Summary Section

Clock	Domain	Nominal Frequency	Required Frequency	Maximum Frequency	Violation
CyILO	CyILO	1.000 kHz	1.000 kHz	N/A	
CyIMO	CyIMO	3.000 MHz	3.000 MHz	N/A	
CyMASTER_CLK	CyMASTER_CLK	26.000 MHz	26.000 MHz	77.149 MHz	
ClkCntr	CyMASTER_CLK	13.000 MHz	13.000 MHz	28.988 MHz	
Clk_PWM	CyMASTER_CLK	100.000 kHz	100.386 kHz	42.411 MHz	
CyBUS_CLK	CyMASTER_CLK	26.000 MHz	26.000 MHz	N/A	
CyPLL_OUT	CyPLL_OUT	52.000 MHz	52.000 MHz	N/A	

ClkCntr の最大周波数が 28MHz になるため、図 54 に示すようにマスター クロックの最大周波数は 56MHz になります。

図 54. Sync コンポーネントが追加された、56MHz マスター クロックでのクロックのまとめ

– Clock Summary Section

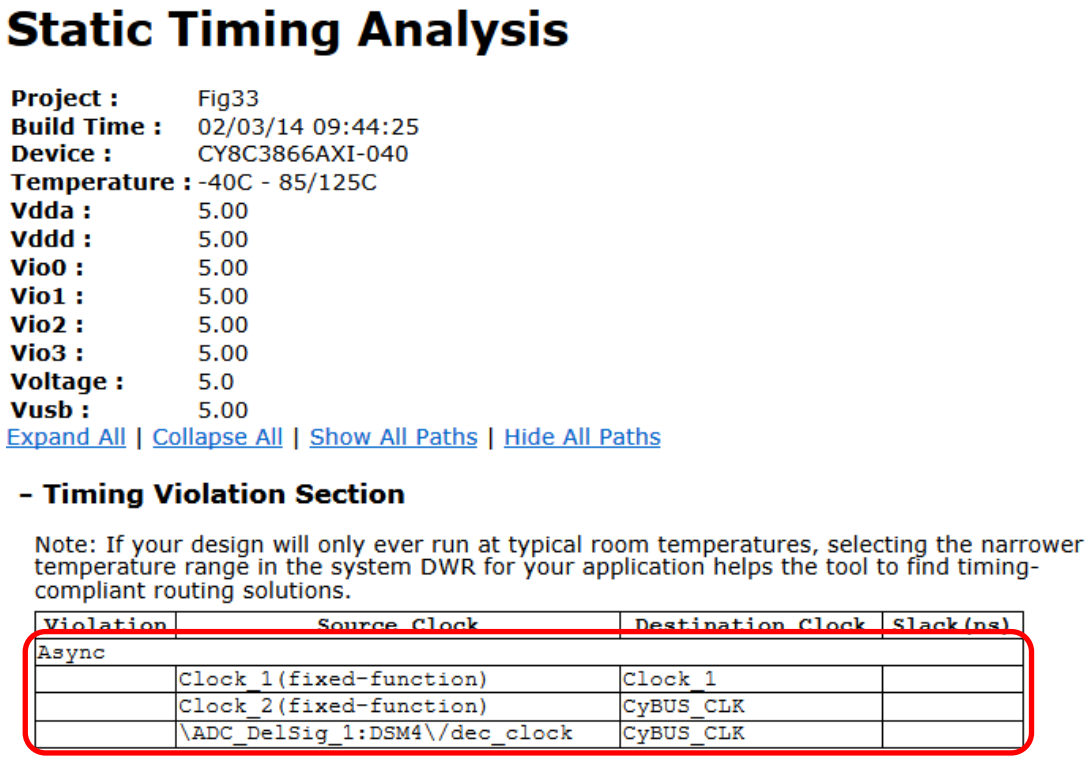
Clock	Domain	Nominal Frequency	Required Frequency	Maximum Frequency	Violation
CyILO	CyILO	1.000 kHz	1.000 kHz	N/A	
CyIMO	CyIMO	3.000 MHz	3.000 MHz	N/A	
CyMASTER_CLK	CyMASTER_CLK	56.000 MHz	56.000 MHz	75.228 MHz	
ClkCntr	CyMASTER_CLK	28.000 MHz	28.000 MHz	28.988 MHz	
Clk_PWM	CyMASTER_CLK	100.000 kHz	100.000 kHz	42.411 MHz	
CyBUS_CLK	CyMASTER_CLK	56.000 MHz	56.000 MHz	N/A	
CyPLL_OUT	CyPLL_OUT	56.000 MHz	56.000 MHz	N/A	

また、最速のマスター クロックとバス クロック (66MHz) が必要な場合は、いつでも ClkCntr をマスター クロックを 2 の代わりに 3 で割った値にセットして、最大周波数要件より小さい 22MHz 周波数を取得することができます。

非同期クロックの交差

STA レポートが生成するもう一つの違反の種類は、図 55 に示す「Async」です。これは常に、固定機能ブロックの出力を UDB ベースコンポーネントの入力に接続することに起因しています。この問題を処理する最も簡単な方法は、トピック 7 で説明したように Sync コンポーネントをデザインに追加することです。

図 55. 非同期のタイミング違反



Static Timing Analysis

Project : Fig33
Build Time : 02/03/14 09:44:25
Device : CY8C3866AXI-040
Temperature : -40C - 85/125C
Vdda : 5.00
Vddd : 5.00
Vio0 : 5.00
Vio1 : 5.00
Vio2 : 5.00
Vio3 : 5.00
Voltage : 5.0
Vusb : 5.00

Expand All | Collapse All | Show All Paths | Hide All Paths

– Timing Violation Section

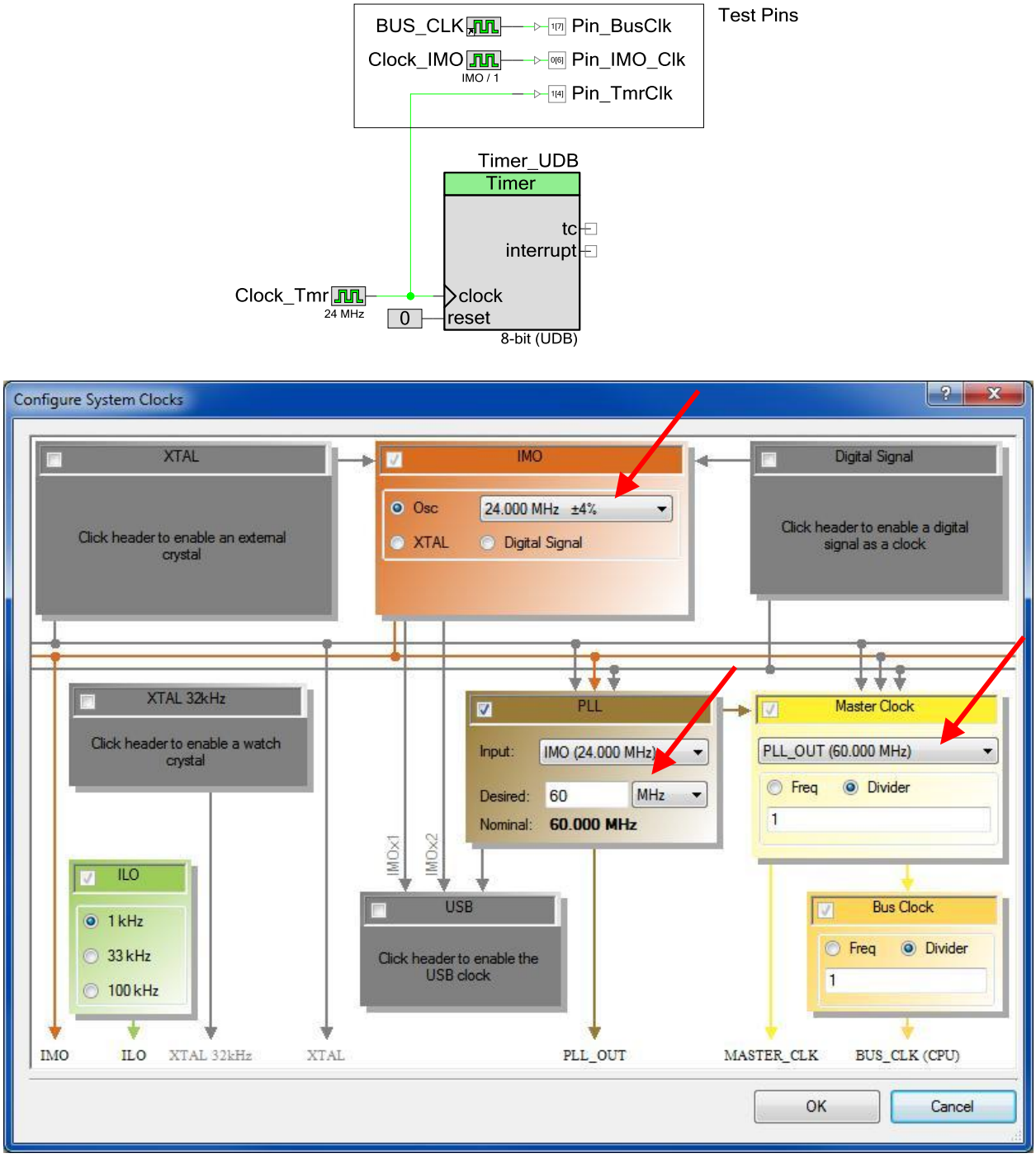
Note: If your design will only ever run at typical room temperatures, selecting the narrower temperature range in the system DWR for your application helps the tool to find timing-compliant routing solutions.

Violation	Source Clock	Destination Clock	Slack(ns)
Async			
	Clock_1(fixed-function)	Clock_1	
	Clock_2(fixed-function)	CyBUS_CLK	
	\ADC_DelSig_1:DSM4\/dec_clock	CyBUS_CLK	

クロックの公称周波数および必要な周波数

上の図を検討して、「STA Clock Summary Section」での「Nominal Frequency」と「Required Frequency」両方のコラムですべてのクロック周波数が同じであることに注意してください。これは一般的な場合ですが、周波数が異なる場合もあります。そのような場合は、図 56 に示すようにマスター クロックが他のクロックの整数倍ではないことに起因しています。Clock_Tmr は IMO をソースとし、マスター クロックは 60MHz です。マスター クロックの周波数はタイマー クロックの 2.5 倍です。

図 56. 公称周波数と必要なクロック周波数が異なるデザイン



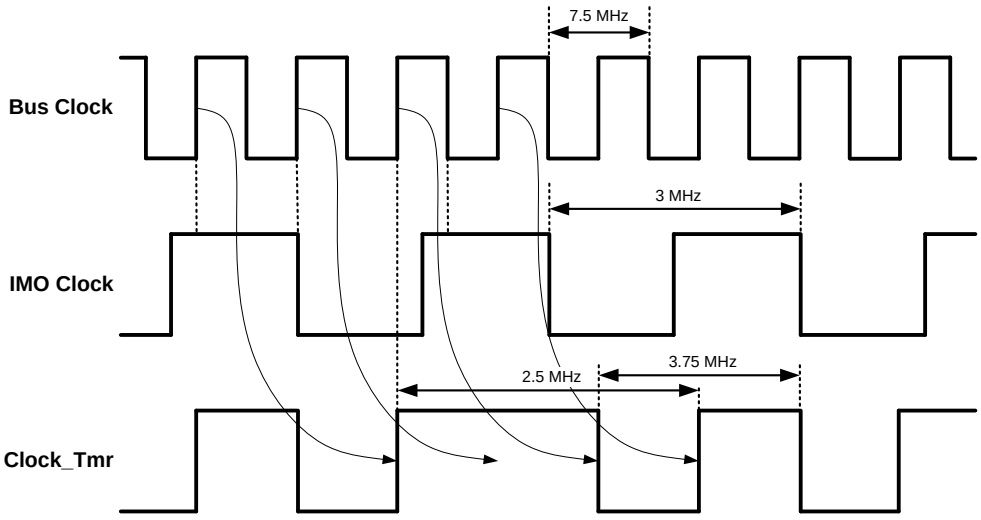
より低い周波数でも同じ影響が出て、33 ページの図 56 に示したテスト ピンでオシロスコープを使って表示することが容易になります。図 57 はほぼ同じ例ですが、Clock_Tmr は 3MHz で、マスター クロックは 7.5MHz (60MHz を 8 で割った値) です。

図 57. 異なる公称周波数と必要な周波数、例 2

Type	Name	Domain	Desired Frequency	Nominal Frequency	Accuracy (%)	Tolerance (%)	Divider	Start on Reset	
System	USB_CLK	DIGITAL	48.000 MHz	? MHz	±0	-	1	☐	IMOx2
System	Digital Signal	DIGITAL	? MHz	? MHz	±0	-	0	☐	
System	XTAL 32kHz	DIGITAL	32.768 kHz	? MHz	±0	-	0	☐	
System	XTAL	DIGITAL	25.000 MHz	? MHz	±0	-	0	☐	
System	ILO	DIGITAL	? MHz	1.000 kHz	-50, +100	-	0	☑	
System	IMO	DIGITAL	3.000 MHz	3.000 MHz	±1	-	0	☑	
System	BUS_CLK (CPU)	DIGITAL	? MHz	7.500 MHz	±1	-	1	☑	MASTER_CLK
System	MASTER_CLK	DIGITAL	? MHz	7.500 MHz	±1	-	8	☑	PLL_OUT
System	PLL_OUT	DIGITAL	60.000 MHz	60.000 MHz	±1	-	0	☑	IMO
Local	Clock_IMO	DIGITAL	? MHz	3.000 MHz	±1	-	1	☑	IMO
Local	Clock_Tmr	DIGITAL	3.000 MHz	3.000 MHz	±1	-	1	☑	Auto: IMO
Local	Clock_UDB_1	DIGITAL	? MHz	7.500 MHz	±1	-	0	☑	BUS_CLK

どちらの場合でも、タイマー クロックは IMO をソースとしていますが、一般的には UDB で使用するにはバス クロックと同期化する必要があります。図 58 はテスト ピンのタイミング図を示します。Clock_Tmr はバス クロックと二重同期化されるため、右へ 2 バス クロック サイクルシフトされることに注意してください。

図 58. タイミング図、異なる公称周波数と必要な周波数



IMO をソースとした Clock_Tmr は、IMO の整数倍でない周波数を持つバス クロックと同期化されるため、Clock_Tmr 出力は 2 つの周波数 (2.5MHz と 3.75MHz) を持ち、どちらの周波数も IMO の周波数に等しくないです。この影響は**同期化ジッタ**と呼ばれています。

これは、図 59 に示すように STA レポートの「Clock Summary Section」に示されます。PSoC Creator の「Clocks」タブで用語「Desired Frequency」と「Nominal Frequency"」(34 ページの図 57)、STA レポートで「Nominal Frequency」と「Required Frequency」(図 59) を区別することが重要です。Nominal Frequency (公称周波数) は 2 つのディスプレイで同じです。Required Frequency (必要な周波数) は、クロックを使用するパスがタイミングを満たすのに必要な周波数です。これはクロックの最高周波数です (この場合は 3.75MHz です)。公称周波数が最大周波数よりも低く、必要な周波数が最大周波数よりも高い可能性があります。この場合、明らかな解決策がないタイミング違反が発生します。

図 59. 異なる公称周波数と必要なクロック周波数を示す STA

+ Clock Summary Section

Clock	Domain	Nominal Frequency	Required Frequency	Maximum Frequency	Violation
Clock_IMO	Clock_IMO	3.000 MHz	3.000 MHz	N/A	
Clock_IMO (routed)	Clock_IMO (routed)	3.000 MHz	3.000 MHz	N/A	
Clock_Tmr (routed)	Clock_Tmr (routed)	3.000 MHz	3.000 MHz	N/A	
CyBUS_CLK (routed)	CyBUS_CLK (routed)	7.500 MHz	7.500 MHz	N/A	
CyILO	CyILO	1.000 kHz	1.000 kHz	N/A	
CyIMO	CyIMO	3.000 MHz	3.000 MHz	N/A	
CyMASTER_CLK	CyMASTER_CLK	7.500 MHz	7.500 MHz	N/A	
Clock_Tmr	CyMASTER_CLK	3.000 MHz	3.750 MHz	55.723 MHz	
CyBUS_CLK	CyMASTER_CLK	7.500 MHz	7.500 MHz	N/A	
CyPLL_OUT	CyPLL_OUT	60.000 MHz	60.000 MHz	N/A	

一般的には、タイマー、カウンター、PWM やその他のタイミング依存の回路を、同期化ジッタを持つクロックで駆動することは良い方法ではありません。デザインの構築後、STA クロックのまとめをレビューしてこの状況が発生しないことを確認してください。この状況が発生した場合、2 つの解決策があります。

- マスター クロックとバス クロック周波数を、他のクロック ソースの周波数の整数倍になるように調整します。ベストプラクティスは、他のクロックの周波数の少なくとも 4 倍にすることです (14 ページの説明を参照してください)。上記の場合では、マスター クロックを 12MHz、すなわち、Clock_Tmr の周波数の 4 倍に変更することができます。
- マスター クロックである、またはマスター クロックと同期するクロック ソースを選択します。この場合、Clock_Tmr のソースは IMO の代わりにマスター クロックです。ただし、これはクロック周波数を変化させ、その結果、デザインの他の部分を変更する必要があることに注意してください。

ホールド時間、回復時間および除去時間の違反

上記の例は、セットアップ時間の違反が最も一般的であるため、これに集中しました。その他の種類の違反は 6 ページの「メタスタビリティおよびレジスタ タイミング」で説明したように、以下の状況で発生します。

- ホールド時間違反は常に、DSI から配線されて、遅れさせられる、またはスキューが発生するクロックに関連しています (図 23 を参照してください)。これで、クロックが遅延させられ、ホールド時間違反が発生します (図 7 を参照してください)。
- 回復時間および除去時間の違反は非同期リセットまたはプリセットに関連しています (図 30 を参照してください)。これらを同期に変更すると、警告が解除されます。

まとめ

本アプリケーション ノートはデジタル設計のいくつかの一般的な原則、およびデザインを PSoC 3、PSoC 4 および PSoC 5LP に実装する方法を示しました。その後、一連のベストプラクティスを説明しました。

最後に、STA 警告の詳細説明および解除する方法を示す詳しい例を提供しました。

本アプリケーション ノートで提供した情報は、PSoC に堅牢かつ高性能のデジタル デザインを実装する手助けになります。Verilog、データパス プログラミングおよびコンポーネント開発の詳細な説明は本アプリケーション ノートの範囲を超えています。それらのトピックの詳細情報は、PSoC Creator に同梱されているコンポーネント開発キット ツールおよび資料を参照してください。また、PSoC Creator デザイン チュトリアルを参照してください。

関連アプリケーション ノート

- AN82250 - Implementing Programmable Logic Designs – An Introduction
- AN82156 - PSoC 3 and PSoC 5LP Designing PSoC Creator Components with UDB Datapaths
- AN54181 - Getting Started with PSoC 3
- AN79953 – Getting Started with PSoC 4
- AN77759 - Getting Started with PSoC 5LP
- AN60631 - PSoC 3 and PSoC 5LP Clocking Resources
- AN72382 - Using PSoC 3 and PSoC 5LP GPIO Pins
- AN62510 – Implementing State Machines with PSoC 3 and PSoC 5LP

著者について

氏名:	Todd Dust
役職:	シニア アプリケーション エンジニア
経歴:	シアトル パシフィック大学、電気工学理学士
氏名:	Mark Ainsworth
役職:	アプリケーション エンジニア主任
経歴:	シラキュース大学、コンピューター エンジニアリング理学士 ; ワシントン大学、電気工学理学修士

改訂履歴

文書名: PSoC® 3、PSoC 4、PSoC 5LP デジタル設計のベストプラクティス - AN81623

文書番号 001-97885

版	ECN	変更者	発行日	変更内容
**	4802473	HZEN	07/14/2015	これは英語版 001-81623 Rev. *D を翻訳した日本語版 001-97885 Rev. **です。
*A	5827507	AESATP12	07/21/2017	更新されたロゴと著作権。

ワールドワイド販売と設計サポート

サイプレスは、事業所、ソリューション センター、メーカー代理店、および販売代理店の世界的なネットワークを保持しています。お客様の最寄りのオフィスについては、サイプレスのロケーション ページをご覧ください。

製品

ARM® Cortex® Microcontrollers	cypress.com/arm
車載用	cypress.com/automotive
クロック&バッファ	cypress.com/clocks
インターフェース	cypress.com/interface
IoT (モノのインターネット)	cypress.com/iot
メモリ	cypress.com/memory
マイクロコントローラ	cypress.com/mcu
PSoC	cypress.com/psoc
電源用 IC	cypress.com/pmic
タッチ センシング	cypress.com/touch
USB コントローラー	cypress.com/usb
ワイヤレス / RF	cypress.com/wireless

PSoC® ソリューション

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6

サイプレス開発者コミュニティ

フォーラム | WICED IOT Forums | Projects | ビデオ | ブログ | トレーニング | Components

テクニカルサポート

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

© Cypress Semiconductor Corporation, 2012-2017. 本書面は、Cypress Semiconductor Corporation 及び Spansion LLC を含むその子会社（以下、「Cypress」という。）に帰属する財産である。本書面（本書面に含まれ又は言及されているあらゆるソフトウェア又はファームウェア（以下、「本ソフトウェア」という。）を含む）は、アメリカ合衆国及び世界のその他の国における知的財産法令及び条約に基づき、Cypress が所有する。Cypress はこれらの法令及び条約に基づく全ての権利を留保し、また、本段落で特に記載されているものを除き、Cypress の特許権、著作権、商標権又はその他の知的財産権のライセンスを一切許諾していない。本ソフトウェアにライセンス契約書が伴っておらず、かつ、あなたが Cypress との間で別途本ソフトウェアの使用方法を定める書面による合意をしていない場合、Cypress は、あなたに対して、（1）本ソフトウェアの著作権に基づき、（a）ソースコード形式で提供されている本ソフトウェアについて、Cypress ハードウェア製品と共に用いるためにのみ、組織内部でのみ、本ソフトウェアの修正及び複製を行うこと、並びに（b）Cypress のハードウェア製品ユニットに用いるためにのみ、（直接又は再販売者及び販売代理店を介して間接のいずれかで）エンドユーザーに対して、バイナリーコード形式で本ソフトウェアを外部に配布すること、並びに（2）本ソフトウェア（Cypress により提供され、修正がなされていないもの）に抵触する Cypress の特許権のクレームに基づき、Cypress ハードウェア製品と共に用いるためにのみ、本ソフトウェアの作成、利用、配布及び輸入を行うことについての非独占的で譲渡不能な一身専属的ライセンス（サブライセンスの権利を除く）を付与する。本ソフトウェアのその他の使用、複製、修正、変換又はコンパイルを禁止する。

適用される法律により許される範囲内で、Cypress は、本書面又はいかなる本ソフトウェアに関しても、明示又は黙示をとわず、いかなる保証（商品性及び特定の目的への適合性の黙示の保証を含むがこれらに限られない）も行わない。適用される法律により許される範囲内で、Cypress は、別途通知することなく、本書面を変更する権利を留保する。Cypress は、本書面に記載のあるいかなる製品又は回路の適用又は使用から生じる一切の責任を負わない。本書面で提供されたあらゆる情報（あらゆるサンプルデザイン情報又はプログラムコードを含む）は、参照目的のためのみに提供されたものである。この情報で構成するあらゆるアプリケーション及びその結果としてのあらゆる製品の機能性及び安全性を適切に設計し、プログラムし、かつテストすることは、本書面のユーザーの責任において行われるものとする。Cypress 製品は、兵器、兵器システム、原子力施設、生命維持装置若しくは生命維持システム、蘇生用の設備及び外科的移植を含むその他の医療機器若しくは医療システム、汚染管理若しくは有害物質管理の運用のために設計され若しくは意図されたシステムの重要な構成部分として用いるため、又はシステムの不具合が人身傷害、死亡若しくは物的損害を生じさせることになるその他の使用（以下、「本目的外使用」という。）のためには、設計、意図又は承認されていない。重要な構成部分とは、装置又はシステムのその構成部分の不具合が、その装置若しくはシステムの不具合を生じさせるか又はその安全性若しくは実効性に影響すると合理的に予想できる、機器又はシステムのあらゆる構成部分をいう。Cypress 製品のあらゆる本目的外使用から生じ、若しくは本目的外使用に関連するいかなる請求、損害又はその他の責任についても、Cypress はその全部又は一部をとわず一切の責任を負わず、かつ、あなたは Cypress をそれら一切から免除するものとし、本書により免除する。あなたは、Cypress 製品の本目的外使用から生じ又は本目的外使用に関連するあらゆる請求、費用、損害及びその他の責任（人身傷害又は死亡に基づく請求を含む）から Cypress を免責補償する。

Cypress、Cypress のロゴ、Spansion、Spansion のロゴ及びこれらの組み合わせ、WICED、PSoC、CapsSense、EZ-USB、F-RAM、及び Traveo は、米国及びその他の国における Cypress の商標又は登録商標である。Cypress の商標のより完全なリストは、cypress.com を参照のこと。その他の名称及びブランドは、それぞれの権利者の財産として権利主張がなされている可能性がある。