

複数のAD9779 TxDAC®の同期化

Steve Reine、Gina Colangelo共著

はじめに

AD9779 TxDAC®は、1GSPSまでのDAC出力サンプリング・レートが可能です。ビーム・ステアリングが必要なアプリケーションなど、場合によっては複数のAD9779で同期をとることができます。したがって、AD9779を最高速度に近いレベルで動作させる場合、TxDACのタイミング仕様が非常に重要になります。

このアプリケーション・ノートでは、AD9779の動作をすべて詳細に説明するわけではありません。内部のデジタル・エンジンについて完全に理解するためには、AD9779のデータシートを読む必要があります。このアプリケーション・ノートでは、複数のAD9779間でREFCLK/DATACLKの同期をとるためにSYNC_Iの機能を拡張して使用しています。

従来のインターポレーション方式のTxDACでは、DAC出力サンプリング・レートのクロックからDACを駆動するときに2つの問題が生じます。1つは、入力データがラッチされているのがどのDACCLKエッジなのか判別しにくいという点です。この問題を解決するために、大部分のDACはDATACLK信号を出力して、入力レジスタのラッチング・エッジがどれかを示します。もう1つは、このアプリケーション・ノートの主要テーマである、複数のTxDACを同期させるときに生じる問題です。複数のデバイスから出力されるDATACLK信号が同期しているという保証はありません。パワーアップ時にデバイスが自動的に同期するというはまず考えられません。AD9779は、データ同期用の2番目のクロックを使用することでこの問題を解決しています。SYNC_Iと呼ばれるこのクロックはAD9779への入力クロックであり、複数のAD9779間で入力データのラッチを同期するために使用します。

このアプリケーション・ノートでは、複数のAD9779デバイス間でデジタル入力データを同期させる方法について詳細に説明します。DAC出力の位相アライメントは、1サイクルのDACCLK出力よりも短くなるように設計により保証されています。ただし、出力遅延のミスマッチ（周囲条件と温度による）のために、複数のDAC出力間で位相アライメントにわずかなミスマッチが発生することがあります。本アプリケーション・ノートでは、この問題については取り上げていません。

同期化の方法

複数のAD9779 DACの同期をとるには、2つの方法があります。1つは、デバイスの1つをマスターとして、その他のデバイスをスレーブとして使用する方法です。もう1つは、すべてのデバイスをスレーブとして使用する方法です。いずれの方法にもタイミングに関して同じ制約がありますが、性能上のトレードオフはありません。図1にマスター／スレーブ・モード、図2にスレーブ・モードのブロック図を示します。

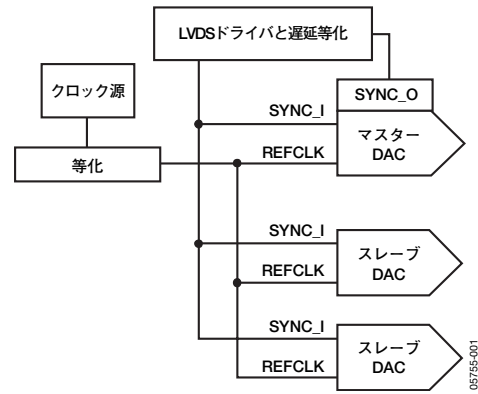


図1. マスター／スレーブ・モードのSYNC_I/O分配の処理系

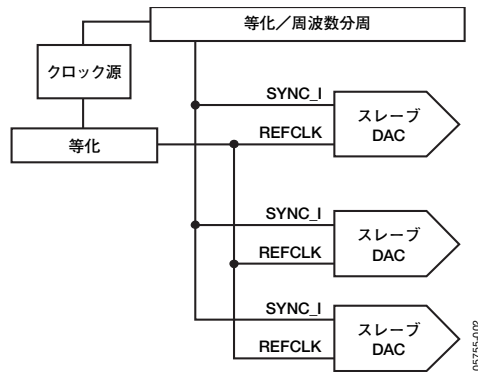


図2. スレーブ・モードのSYNC_I分配の処理系

REV. A

アナログ・デバイセズ株式会社

本 社 / 〒105-6891 東京都港区海岸1-16-1 ニューピア竹芝サウスタワービル
電話03(5402)8200
大阪営業所 / 〒532-0003 大阪府大阪市淀川区宮原3-5-36 新大阪MTビル2号
電話06(6350)6868

AN-822

目次

はじめに	1	同期化の詳細	3
同期化の方法	1	タイミング仕様	7
改訂履歴	2		

改訂履歴

9/06—Rev. 0 to Rev. A

Replaced Introduction Section	1
Replaced Synchronization Details Section	3
Replaced Timing Specifications Section	7

2/06—Revision 0: Initial Version

同期化の詳細

動作中は、差動クロック信号でマスターとスレーブのすべてのデバイスのAD9779 REFCLK入力を駆動します。REFCLK入力レシーバは高ゲインの差動アンプで、各差動入力に400mVp-pの振幅と400mV近くのCOMMONモード入力レベルが必要です。

マスター・デバイスを選択する場合、マスター・デバイスで差動LVDS出力信号をイネーブルにできます。この信号はSYNC_O+およびSYNC_O-といいます。レジスタ0x07のビット5を使用して、DACCLKの立上がりまたは立下がりエッジでSYNC_Oがトリガされるように設定できます。レジスタ0x04のビット0 (MSB) とレジスタ0x05のビット[7:4] (LSB) を使用して、SYNC_Oの遅延時間を設定することも可能です。SYNC_Oの信号速度は、レジスタ0x04のビット[3:1]に基づいて、REFCLK速度の整数の約数とすることができます。マスター・デバイスのREFCLK入力信号とSYNC_O信号の予想タイミング図を図3に示します。

SYNC_OドライバとSYNC_Iレシーバは、LVDSレベルで規定されています (AD9779のデータシートを参照)。

複数のAD9779のCMOSデジタル・データ入力を駆動するパラレル・バスの遅延時間は等しくする必要があります。複数のデータ・バスが等化されていない場合は、DATA_CLOCK_DELAY (レジスタ0x04のビット[7:4]) の設定で、各AD9779のラッチング時間を約180psのインクリメントでオフセットできます。AD9779は、1つのデータ・バス内のビット・スキューを補償することはできません。

すべてのAD9779デバイスにおいて、SYNC_IとREFCLK入力の関係はセットアップとホールドになります。これらのタイミングの関係については、AD9779データシートの「タイミング情報」の項を参照してください。

SYNC_OとSYNC_O_DELAYによる推奨の利用方法は、SYNC_IとREFCLKのタイミングの等化にSYNC_O_DELAYを使用することで、これによりタイミングの関係を有効にできます。

SYNC_Iは、レジスタ0x05のビット0 (MSB) とレジスタ0x06のビット[7:4] (LSB) を使用して遅延を設定できます。遅延の等化が不完全なアプリケーションや、図2の回路 (スレーブ・モード) を選択するアプリケーションでは、SYNC_I_DELAYを使用することができます。SYNC_Iをイネーブルにするには、同期レシーバ・イネーブル・ビット (レジスタ0x07のビット7) を設定します。

表1に、SPIレジスタで設定できるSYNC_O_DELAYとSYNC_I_DELAYのインクリメントごとの遅延を示します。

表1

温度	SYNC_I/O_DELAY (インクリメントごとの概算遅延)
-40°C	72ps
+25°C	78ps
+85°C	83ps

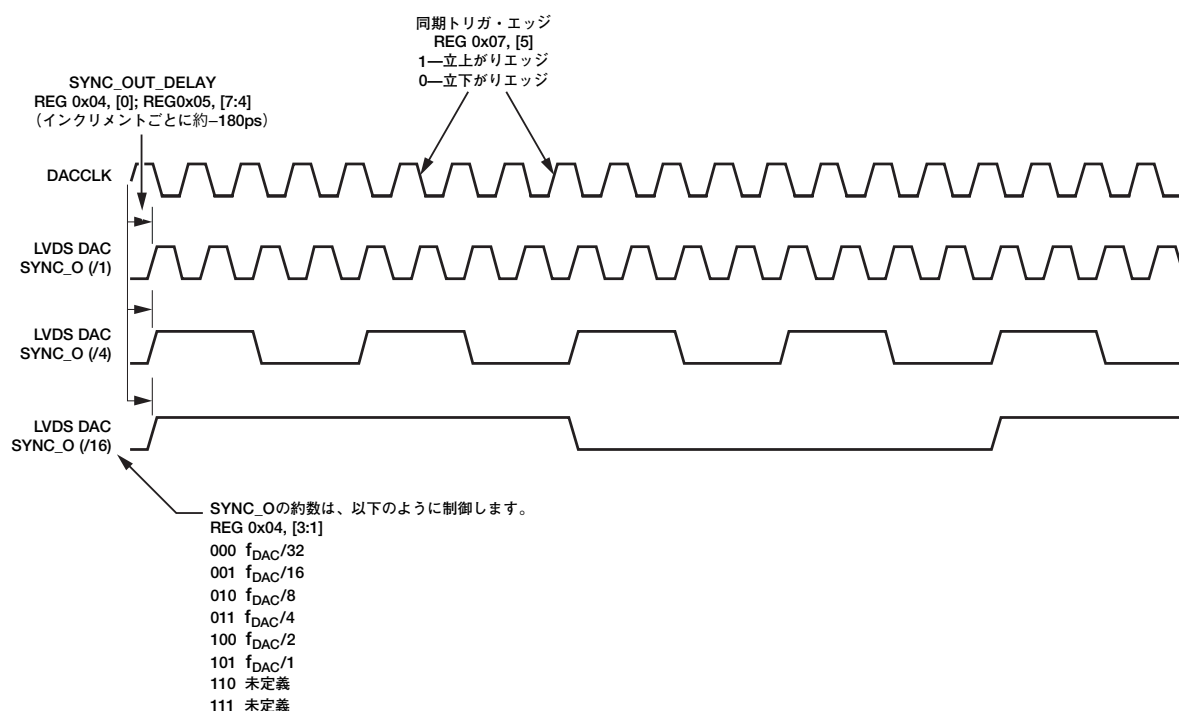


図3. DACCLKSYNC_Oのタイミング

05755-003

図4に、複数のAD9779の同期化に使用する内部回路のブロック図を示します。プログラマブル遅延の後、SYNC_I信号はSYNC_Iの立上りエッジごとにDACCLK周期の単一パルス長だけが残るように処理されます。この場合、DACCLKはAD9779 DACの内部サンプル・レートのクロックとなることに注意してください。これはAD9779の設定にもよりますが、REFCLKの場合も同様です。DACCLK周期のこの単一パルス長は、図4に示す5ビット・デバイダのロード信号を駆動します。デバイダの遅延ロジックから出力される5つの信号は、ゼロ充填がイネーブになっている場合も含め、インターポレーション・レートのすべてにおいて考えられるDATACLK信号となります。DACCLKオフセット・レジスタの設定によって、図4に示すビット1からビット4までの遅延がDACCLKサイクルのインクリメントごとに可能になります。図6に、5ビット・デバイダの内部タイミング、ロード信号の効果、DACCLKオフセット値を示します。

エッジ検出器は誤差検出回路も駆動します（図5の詳細な回路図を参照）。このプログラマブルな誤差検出回路ではタイミング・マージンを計測し、このマージンを越えた場合に割込みを発生させることができます。

図4の点線で囲んだ部分の詳細を図5に示します。内部では、FF5に入力される信号が相互にセットアップとホールドのタイミング条件に適合しなければなりません。FF5の入力信号のタイミングが無効になれば、REFCLKとデジタル入力データ間の同期が失われる可能性があります。このポイントのタイミング不良は、一般にDAC出力ノイズ・フロアの増加によってわかります。FF5の入力信号のタイミング条件をDACCLKとSYNC_Iの入力に適用すると、この2つの入力に対するセットアップとホールドのタイミング条件になります。

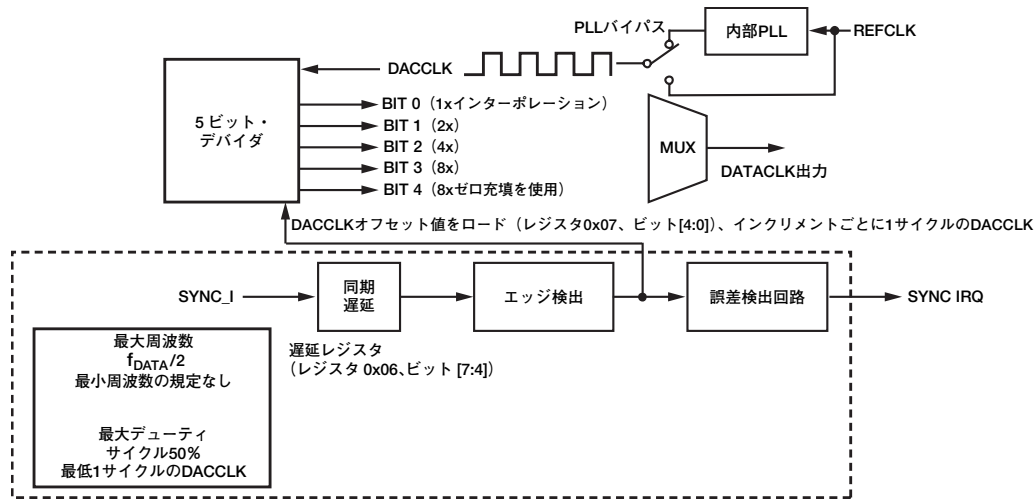


図4. AD9779マルチDAC同期回路のブロック図

05755-004

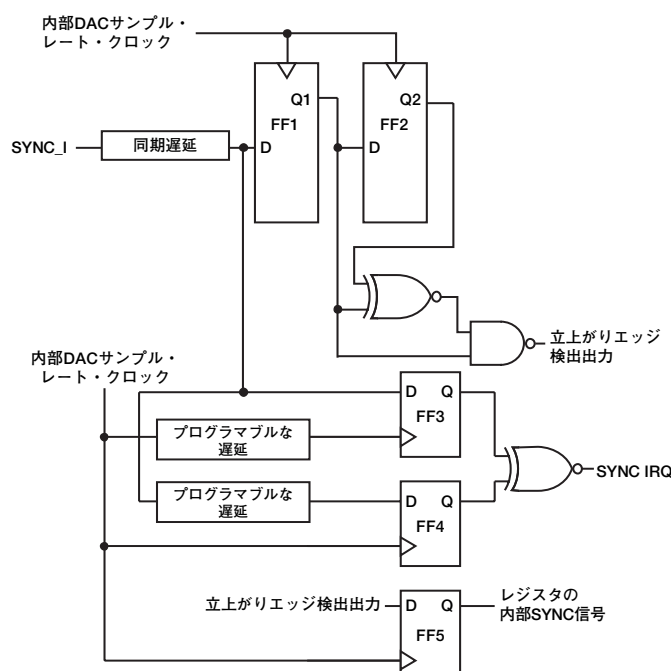


図5. プログラマブルなタイミング・マージンとロード信号生成の詳細

同期入力遅延を変化させることで、REFCLK/SYNC_Iの有効なタイミング・ウィンドウを効果的にシフトさせることができます。実際のアプリケーションでは、所定の同期入力遅延があれば、所定の幅の有効なREFCLK/SYNC_Iタイミング・ウィンドウが得られます。タイミング・マージンを1つずつインクリメントする場合、タイミング・マージンの値はSYNC IRQが設定される値に設定することができます。この値にタイミング・マージンを設定すると、実際にはSYNC IRQの設定は0のマージンになります。SYNC IRQは、セットアップ違反によるタイミング・エラーとホールド違反によるタイミング・エラーを区別しません。ただし、設計により、プログラマブルなタイミング・マージンがセットアップまたはホールドのいずれか小さい方のマージンを超えると、SYNC IRQが設定されます。タイミング・マージンは、レジスタ0x06のビット[3:0]の値を増加させることで大きくすることができます。マージンが0のときは、注意すべき仕様（セットアップまたはホールド）の方向に少しでもドリフトが生じると、SYNC IRQが設定されます。

実際、DACCLKはエッジ検出器の出力をサンプリングしています。エッジ検出器の出力は単一パルスであり、ロジック・ハイレベル幅は1サイクルのDACCLKと同じです。ロード信号が有効になるには、内部DACCLK信号の立ち上がりエッジ周辺の所定のタイミング・ウィンドウ内で、エッジ検出器の出力が一定（ハイレベルまたはローレベル）でなければなりません。

プログラマブルなタイミング・マージンの設定が0で、FF5の入力のタイミングが有効とすれば、FF3とFF4のQ出力は同じになり、SYNC IRQはリセット状態にとどまります。これと同じ条件で、FF5の入力のタイミングが無効であれば、FF3とFF4の出力が異なり、SYNC IRQが設定されます。FF5の入力に有効なタイミング条件が存在する場合は、プログラマブルなタイミング・マージンを0より大きい値に設定してタイミング・マージンを決める必要があります。

マスター/スレーブの同期構成でAD9779を使用するシステムを設計する場合は、SYNC IRQを設定する前にSYNC_O DELAYの値（プログラマブル・タイミング・マージンを可能な限り大きい値に設定できる値）を決めることを推奨します。これが、最適なタイミングで最大のタイミング・マージンになります。その後で、プログラマブル・タイミング・マージンの数値を低くすることができます。プログラマブル・タイミング・マージンをどのくらい低下させるかによって、ドリフトに対するSYNC IRQの感度が決まります。

AD9779が受信できる高いDACCLK周波数では、DACCLKとSYNC_Iの有効なタイミング域がDACCLKサイクルの大部分になることがあります。ただし、もっと遅いDACCLK周波数の場合、プログラマブル・タイミング・マージンの範囲では無効なタイミング・ウィンドウが見つからないことがあります。このような場合、通常のドリフト特性では、AD9779が温度変化に伴ってドリフトして無効なタイミング条件になることはありません。

確実に同期させるために、SYNC_Iの最大レートはDATACLK/2になっています（DATACLKはAD9779の入力データ・レートです。DACCLKではありません）。図6に、SYNC_Iを加える2つの例を紹介します。いずれの例でも、AD9779は4×インターポレーション・モードであり、SYNC_IはDACCLK/8の速度で動作します。このため、4×ラインもDATACLK出力信号です。図6のSYNC_I_DELAYED (a) では、DACCLKのオフセット値を0000に設定しています。内部のSYNC_I_DELAYED (a) 信号の立ち上がりエッジで、DACCLKのエッジが立ち上がるとDATACLK出力ビットがすべて0にリセットされます。ここでは、4×ラインを時間(X)で設定するためには、DACCLKを基準とするウィンドウ(Y)でSYNC_I_DELAYED信号が発生しなければなりません。SYNC_I_DELAYED (a) 信号がこのウィンドウよりわずかでも前か後に発生すると、4×ラインの立ち上がりエッジがDACCLKの1サイクル分進むか、遅延します。

DACCLKのオフセット値を00000に設定すると、SYNC_I_DELAYED (a) 信号を加えてから4×ラインの立上がりエッジの間にDACCLKの1サイクル分の遅延が生じます。

図6のSYNC_I_DELAYED (b) では、DACCLKのオフセット値を時間 (Z) で00010に設定しています。これにより、8×、4×、2×のビットが010 (DACCLKオフセット・ビットと一致) に設定されています。したがって、4×ラインの次の立上がりエッジ (DATACLK出力) は、DACCLKの3サイクル後に発生します。

このように、複数のDACが特定の時間ウィンドウ内でSYNC_Iパルスを受信し、すべてのDACのDACCLKオフセット値が同一であれば、DATACLK信号が同期します。これにより、複数のAD9779デバイスのデータ・ラッチ動作が同時に発生します。

最初の同期化では、2×、4×、8×のカウンタ・ビットが不連続になる場合があります。すなわち、SYNC_Iの立上がりエッジを初めて加えると、同期によってカウンタの値が倍数で変化する状態になることがあります。しかし、最初の同期が完了した後は、SYNC_Iの速度がDATACLK/2またはそれよりも遅い速度のままであれば、2×、4×、8×のビットが0にリセットされる時のみ同期パルスが発生します (これは冗長かもしれませんが、同期が実現していれば、SYNC_Iパルスを加える必要がなくなります)。しかし、初期パルスの中でSYNC_Iパルスを周期的に発生させる主な理由は、ごくまれに複数のAD9779デバイスが非同期になる事態に対処するためです。これは、電源のグリッチや、場合によってはシステム内のAD9779デバイスの一部だけをトリガする不良クロック・パルスのために起こります。

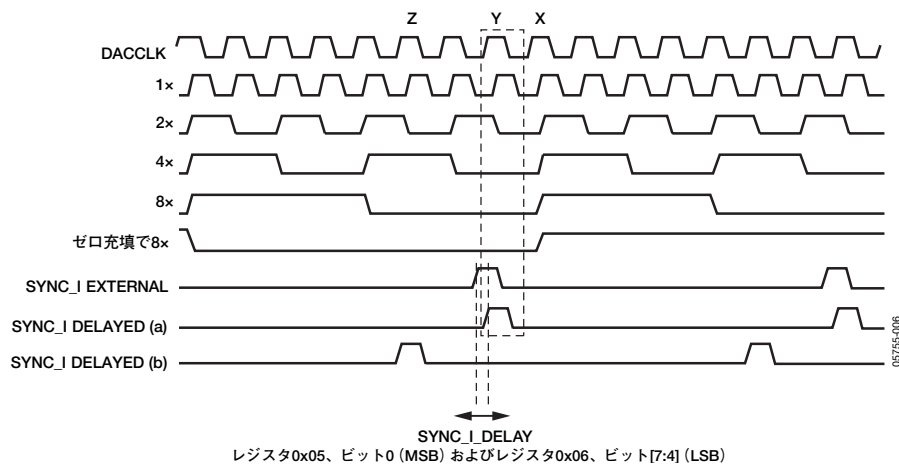


図6. SYNC_I、DACCLK、DATACLKの内部タイミング

タイミング仕様

最初に留意すべきタイミング仕様は、図7に示すように SYNC_I と REFCLK の間で必要とされる関係です。AD9779 のデータシートによると、必要なタイミング仕様は $t_S = -0.2\text{ns}$ 、 $t_H = 1.0\text{ns}$ です。

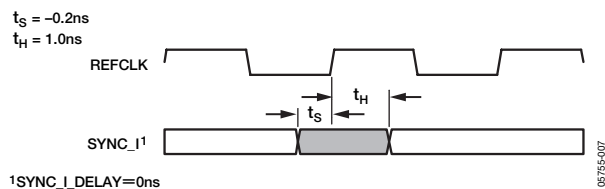


図7. REFCLKとSYNC_Iのタイミングの関係

DACCLK OFFSETを0以外の値に設定すれば、図7に示す DACCLK信号はDACCLKの1サイクル分、左側に移動します。これと同様に、SYNC_I_DELAYを0以外の値に設定すると、SYNC_I_DELAYがインクリメントされるたびに、図7に示す SYNC_I信号はデータシートに記載されている SYNC_I_DELAY のインクリメント値だけ左側に移動します。

図8に、このタイミング情報を示します。これらの値は DATACLK_DELAY_ENABLEのリセットで有効です。DATACLK_DELAY_ENABLEが設定されていれば、DATACLKが遅延（図8で右側に移動）しますが、デジタル入力データのサンプリング・ポイントは移動しません。このため、 t_S と t_H のキープアウト・ウィンドウはDATACLKに対して左側に移動します。DATACLK_DELAY_ENABLEが設定されているときのインクリメントごとの平均遅延およびDATACLK_DELAYのインクリメント値に対応する平均遅延については、AD9779のデータシートに記載されています。

セットアップ/ホールド・データとREFCLKとの関係については、AD9779のデータシートを参照してください。アプリケーションによっては、このデータが必要になることがあります。

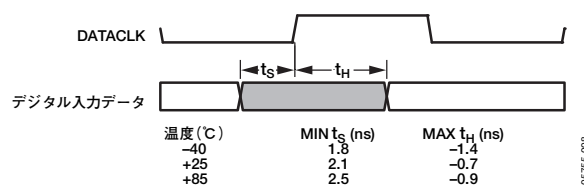


図8. DATACLKと入力データの間のセットアップ&ホールド時間