サイプレスはインフィニオン テクノロジーズになりました

この表紙に続く文書には「サイプレス」と表記されていますが、これは同社が最初にこの製品を開発したからです。新規および既存のお客様いずれに対しても、引き続きインフィニオンがラインアップの一部として当該製品をご提供いたします。

文書の内容の継続性

下記製品がインフィニオンの製品ラインアップの一部として提供されたとしても、それを理由としてこの文書に変更が加わることはありません。今後も適宜改訂は行いますが、変更があった場合は文書の履歴ページでお知らせします。

注文時の部品番号の継続性

インフィニオンは既存の部品番号を引き続きサポートします。ご注文の際は、データシート記載の注文部品番号をこれまで通りご利用下さい。

9-Bit 電圧出力 DAC Data Sheet DAC9 V 2.2

Copyright © 2001-2014 Cypress Semiconductor Corporation. All Rights Reserved.

リソース	PSoC® ブロック			API メモリ（バイト）		ピン（外付け I/O あたり）
	デジタル	アナログ CT	アナログ SC	Flash	RAM	
CY8C29/27/24/22xxx, CY8C23x33, CY8CLED04/08/16, CY8CLED0xD, CY8CLED0xG, CY8CTST120, CY8CTMG120, CY8CTMA120, CY8C28x45, CY8CPLC20, CY8CLED16P01, CY8C28x43, CY8C28x52						
	0	0	2	151	0	1
CY8C26/25xxx	0	0	2	151	0	1

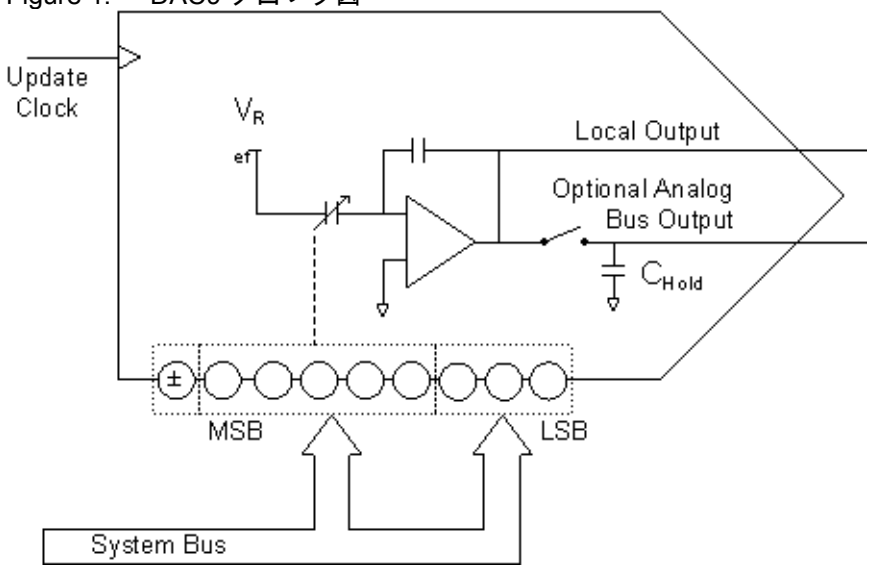
このユーザモジュールを使用したサンプルプロジェクトは、以下を参照してください。
www.cypress.com/psocexampleprojects。

特徴と概要

- 9-bit 分解能
- 電圧出力
- 2 の補数、オフセットバイナリ、及び記号 / サイズの入力データフォーマット
- アナログバスと外付け出力装置用 sample and hold 回路
- 125 ksps アップデートレート

DAC9 ユーザーのモジュールは、デジタルコードを出力電圧に変換します。DAC9 は毎秒 125k サンプルまでの更新レートでのデジタルコードを出力電圧に変換します。アプリケーションプログラミングインターフェイス (API) は、最大の流動性のためにオフセットーバイナリ、2 の補数、レジスタイメージ (符号とマグニチュード) のデータフォーマットをサポートしています。変換エラーを最小限にするためのオフセット補正機能が使用されています。

Figure 1. DAC9 ブロック図

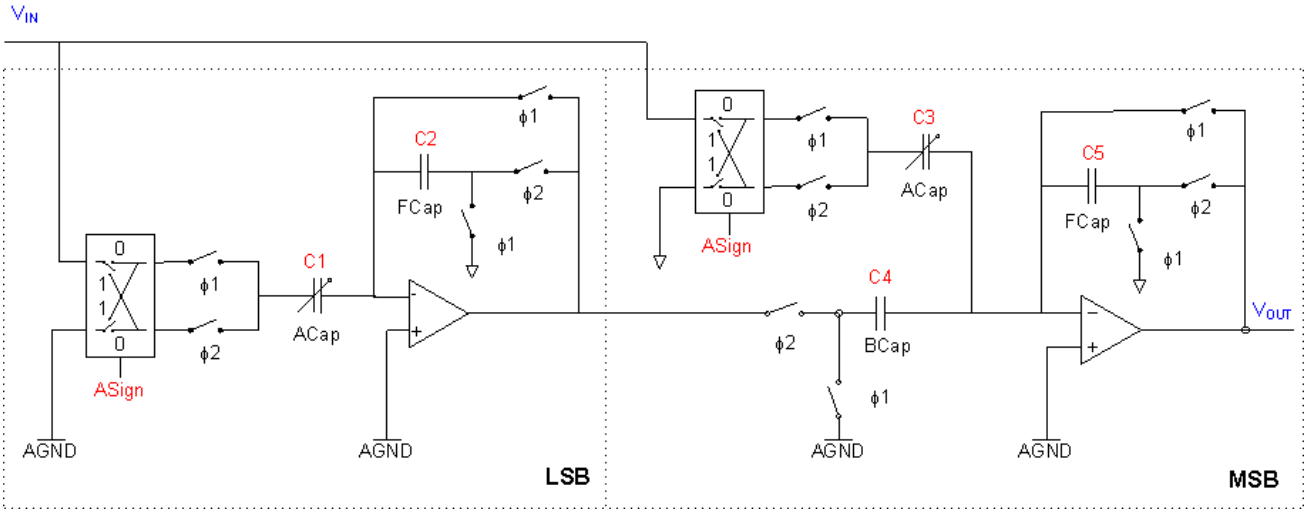


機能説明

DAC9 ユーザーのモジュールは、デジタルコードをアナログ出力電圧に変換します。デジタルコードは、-255 ~ +255 の範囲を持つ 2 の補数フォームに数値として表示されます。代替的な方法で、入力のコードを 0 ~ 510 までの範囲の数値でオフセットーバイナリ形式に表示することができます。システムレベルのパラメータ、RefMux のために選択した値に応じて、出力電圧範囲が異なる場合があります。

DAC9 ユーザ モジュールは、任意の 2 つのアナログ PSoC ブロックにマッピングします。これらのブロックは、指定された LSB と MSB です。LSB ブロック（ステージ）は、ブロックの “BCap" キャパシタ C_4 を介して MSB ステージと結合しています。内部的に見ると、これらの作業は、コード、及びサイズのフォーマットに基づいています。マグニチュード がもっとも高い 5 つのビットは、以下の単純な図式で提示されたバイナリ加重コンデンサアレイである C_3 値を設定します。マグニチュード がもっとも低い 2 つのビットは、C_1 値を設定します。C_3 にはゼロから 31 単位の値が入り、C_1 はキャパシタンス単位のセット {0, 4, 8, 12, 16, 20, 24, 28} からの値が入ります。ASign ビットによって反転できるリファレンス電圧は、各ステージで、マグニチュードキャパシタ C_1 および C_3 の割合によって、それぞれフィードバックキャパシタ C_2 および C_5 へスケール調整されます。それぞれ 32 単位の公称容量を持っています。さらに LSB ステージの出力は、結合キャパシタ C_4 によって、フィードバックキャパシタ C_5 へスケール調整されます。

Figure 2. DAC9 への単純な図式



ハードウェアがそれぞれの更新プログラムの周期でオフセット補正作業を実行します。Φ_1 と Φ_2 によって制御されるスイッチは、Φ_1 の間、反転入力の利点フォロワーとしてオペアンプを構成します。この構成では、オフセット電圧がサミングノードに表示され、ACaps、BCaps、FCaps を充電します。Φ_2 に再構成されたように、回路がこのようなコンデンサのオフセットの充電値を変換し、オフセット電圧を効果的にオフにします。

すべての更新プログラムのサイクルでは、オフセット補正の直接の結果として V_{out} が opamp オフセット電圧 (Φ_1 間) 、及び必要な電圧 (Φ_2 間、定着される) の間で変化されます。精度を向上させるために、これらの価格を軽減させることができる 1 つの方法は、出力バスに関連する sample-and-hold 回路を使用することです。V_{out} は、第 Φ_2 の後半、load 及び hold コンデンサ (DAC9 ブロック ダイアグラムの C_{Hold}) を充電します。C_{Hold} は、該当のサイクルが終了する時点で、opamp 出力装置から分離されます。個々のアナログ出力バスは、非常に高い入力インピーダンスを持つアナログ出力デバイスのバッファによって使用されます。

スケール調整されたレファレンス結果の出力を図に示します。

Equation 1

$$V_{Out} = (V_{REF} \pm AGND)\frac{C_1 C_4}{C_2 C_5} + (V_{REF} \pm AGND)\frac{C_3}{C_5} + AGND$$

グローバル パラメータ RefMux がデバイス エディタで (2 BandGap) ± BandGap に構成されているとき、AGND は 2.6V、リファレンス電圧は 1.3V です。出力は以下の通りです。

Equation 2

$$2.6Volts \pm 1.3Volts\left(\frac{C_1}{2^5 \cdot 2^5} + \frac{C_3}{2^5}\right)$$

結果として、式 2 が 10 ビットのマグニチュードを伴って表示されます。ただし、C_1 はマグニチュードがもっとも低い 3 つのビットを 4 倍にスケール調整することで得られる値に制限されています。C_1 とその分母からスケール係数を取り除いた場合、結果は次のようになります。

Equation 3

$$2.6\text{Volts} \pm 1.3\text{Volts}\left(\frac{C_1}{2^5 \cdot 2^5} + \frac{C_3}{2^5}\right), \qquad C_1 \in \{0, 1, 2, 3, 4, 5, 6, 7\}$$

例

上記の式 2 と 3 では、9bit DAC 入力コードが 2 の値と符号ビットに配置されていることが示されています。入力コードの MSB は、符号ビットとして使用されます。ビット 2 ~ 7 は、C_3 の 5 ビット値を指定するために使用され、3 つの LSB は C_1 を指定するために使用されます。入力コード値 +205 の符号は正であるため、C3 の値は 25、C1 の値は 5 です。式 2 にこれらを適用すると、式 4 が得られます。

Equation 4

$$V_{Out} = 2.6\text{Volts} + 1.3\text{Volts}\left(\frac{5}{2^5 \cdot 2^5} + \frac{25}{2^5}\right) = 3.63\text{Volts}$$

計算された値は、実際的な値であり、これは、システムのノイズやチップのオフセット値に応じて大幅に異なることがあります。

DC および AC の電気的特徴

The following values are indicative of expected performance and based on initial characterization data. 次の表に明記されていない場合は、T_A = 25°C 及び Vdd = 5V 等式が成立します。また、f_{clock} = 125 kHz、外付け AGND 2.50V、外付け $_{Ref}$ 1.23V、REFPWR = HIGH, SCPOWER = ON の場合には、PSoC の電源ブロックは HIGH です。

Table 1. 5.0V DAC9 DC 及び AC 電気の特性 CY8C29/27/24/22xxx, CY8C23x33, CY8CLED04/08/16, CY8CLED0xD, CY8CLED0xG, CY8CTST120, CY8CTMG120, CY8CTMA120, CY8C28x45, CY8CPLC20, CY8CLED16P01, CY8C28x43, CY8C28x52 PSoC デバイスファミリ

パラメータ	典型	制限	単位	条件および注意
分解能	--	9	ビット	
リニア				
DNL	0.99	--	LSB	
INL	0.76	--	LSB	
単調	○	--		
ゲイン誤差				
リファレンス ゲイン誤差を含む	3.4		%FSR	
リファレンス ゲイン誤差を除く [3]	0.5		%FSR	
V_{OS}, オフセット電圧	±7.5	--	mV	
出力ノイズ	4.4	--	mV rms	0 ~ 300 kHz

パラメータ	典型	制限	単位	条件および注意
f_{clock}, アナログコラムクロック [1]				
低出力	8 ~ 500	--	kHz	
中出力	4 ~ 2000	--	kHz	
高出力	4 ~ 3200	--	kHz	
動作電流 [2]				
低出力	300	--	µA	
中出力	1130	--	µA	
高出力	4315	--	µA	

The following values are indicative of expected performance and based on initial characterization data. 次の表に特に明記されていない場合は、T_A = 25°C と Vdd = 3.3V 等式が成立します。また、f_{clock} = 125 kHz、外付け AGND 1.50V、外付け V_{Ref} 0.8V、REFPWR = HIGH、SCPOWER = ON の場合には、PSoC の電源ブロックは HIGH です。

Table 2. 3.3V DAC9 DC DC 及び AC 電気の特性 CY8C29/27/24/22xxx, CY8C23x33, CY8CLED04/08/16, CY8CLED0xD, CY8CLED0xG, CY8CTST120, CY8CTMG120, CY8CTMA120, CY8C28x45, CY8CPLC20, CY8CLED16P01, CY8C28x43, CY8C28x52 PSoC デバイスファミリ

パラメータ	典型	制限	単位	条件および注意
分解能	--	9	ビット	
リニア				
DNL	0.99	--	LSB	
INL	075	--	LSB	
単調	○	--		
ゲイン誤差				
リファレンス ゲイン誤差を含む	2.6		%FSR	
リファレンス ゲイン誤差を除く [3]	0.3		%FSR	
V_{OS}, オフセット電圧	±7.5	--	mV	
出力ノイズ	3	--	mV rms	0 ~ 300 kHz
f_{clock}, アナログコラムクロック [1]				
低出力	8 ~ 500	--	kHz	
中出力	4 ~ 2000	--	kHz	
高出力	4 ~ 3200	--	kHz	

パラメータ	典型	制限	単位	条件および注意
動作電流 [2]				
低出力	290	--	µA	
中出力	1090	--	µA	
高出力	4175	--	µA	

The following values are indicative of expected performance and based on initial characterization data. 次の表に特に明記されていない場合は、T_A = 25°C と Vdd = 2.7V 等式が成立します。また、f_{clock} = 125 kHz、外付け AGND 1.50V、外付け V_{Ref} 0.8V、REFPWR = HIGH、SCPOWER = ON の場合には、PSoC の電源ブロックは HIGH です。

Table 3. 2.7V DAC9 DC 及び AC 電気の特性 CY8C29/27/24/22xxx, CY8C23x33, CY8CLED04/08/16, CY8CLED0xD, CY8CLED0xG, CY8CTST120, CY8CTMG120, CY8CTMA120, CY8C28x45, CY8CPLC20, CY8CLED16P01, CY8C28x43, CY8C28x52 PSoC デバイスファミリ

パラメータ	典型	制限	単位	条件および注意
分解能	--	9	ビット	
リニア				
DNL	0.99	--	LSB	
INL	075	--	LSB	
単調	○	--		
ゲイン誤差				
リファレンス ゲイン誤差を含む	2.6		%FSR	
リファレンス ゲイン誤差を除く [3]	0.3		%FSR	
V_{OS}, オフセット電圧	±7.5	--	mV	
出力ノイズ	3	--	mV rms	0 ~ 300 kHz
f_{clock}, アナログコラムクロック [1]				
低出力	8 ~ 500	--	kHz	
中出力	4 ~ 2000	--	kHz	
高出力	4 ~ 3200	--	kHz	
動作電流 [2]				
低出力	290	--	µA	
中出力	1090	--	µA	
高出力	4175	--	µA	

電気的特性に関する注意

1. 広帯域ノイズの増加のために、指定された 3dB 範囲の上部の先端。ドループ < 1 LSB の下部の先端。DAC によって選択されたアナログコラムクロックは、更新プログラムを管轄する位相のクロックの 4 倍以上の高速です。タイミングの説明については下記を参照してください。
2. 基準電源ブロックのが含まれておらず、一般的なアナログブロックです (PSoC データシートを参照)。
3. リファレンス ゲイン誤差は、テスト マルチ プレクサによって経路指定され、ピンに戻される $V_{RefHigh}$ および V_{RefLow} と外付けリファレンスを比較して測定します。

次の表に特に明記されていない場合は、すべての限界値が T_A = 25°C と Vdd = 5V 等式を保証します。また、f_{clock} = 125 kHz、外付け AGND 2.50V、外付け V_{Ref} 1.23V、REFPWR = HIGH、SCPOWER = ON の場合には、PSoC が電源 HIGH を遮断します。

Table 4. 5.0V DAC9 DC 及び AC 電気の特性 CY8C26/25xxx デバイス

パラメータ	典型 [1]	限界 [2]	単位	条件および注意
分解能	—	9	ビット	
直線性 [6]				
DNL	.10	.25	LSB	
INL	.15	.40	LSB	
単調性 [6]	—	○	ビット	
ゲイン誤差	1.0	2.5	%FSR	
V_{OS}, オフセット電圧 [3]	8	43	mV	
出力ノイズ				
帯域限界	.3	1	mV rms	0 ~ 10 kHz
広帯域	7	10	mV rms	0 ~ 300 kHz
f_{clock}, アナログコラムクロック [4]	—	128 ~ 1332	kHz	
動作電流 [5]				
低出力	250	—	µA	
中出力	560	—	µA	
高出力	1560	2000	µA	

次の表に特に明記されていない場合は、すべての限界値が T_A = 25°C と Vdd = 3.3V 等式を保証します。また、f_{clock} = 125 kHz、外付け AGND 1.50V、外付け V_{Ref} 0.80V、REFPWR = HIGH, SCPOWER = ON の場合には、PSoC 電源ブロックは HIGH です。

Table 5. 3.3V DAC9 DC 及び AC 電気の特性

パラメータ	典型 [1]	限界 [2]	単位	条件および注意 [6]
分解能	—	9	ビット	
直線性 [6]				
DNL	.25	.25	LSB	
INL	.20	.45	LSB	
単調性 [6]	—	○	—	
ゲイン誤差	1.0	2.5	%FSR	
V_{OS}, オフセット電圧 [3]	7	31	mV	
出力ノイズ				
帯域限界	.3	1	mV rms	0 ~ 10 kHz
広帯域	7	10	mV rms	0 ~ 300 kHz
f_{clock}, アナログコラムクロック [4]	—	128 ~ 1332	kHz	
動作電流 [5]				
低出力	200	—	µA	
中出力	500	—	µA	
高出力	1280	1800	µA	

電気的特性に関する注意

1. 一般的な値は、統計平均値 + 1 を意味します σ。限界値は、テストまたは統計分析により保証されます。
2. 外部 AGND に対する 2 の補数ゼロスケールオフセット値には、アナログ出力バッファのオフセットエラーが含まれていません。
3. 広帯域ノイズの増加のために、指定された 3dB 範囲の上部の先端。ドループ < 1 LSB の下部の先端。DAC によって選択されたアナログコラムクロックは、更新プログラムを管轄する位相のクロックの 4 倍以上の高速です。タイミングの説明については下記を参照してください。
4. 基準電源ブロックのが含まれておらず、一般的なアナログブロックです (PSoC データシートを参照)。

本ユーザ モジュールの 「配置」セクションに、この直線性と単調性要件に適合する配置が一覧表示されています。

配置

DAC9 ユーザ モジュールは、指定された LSB と MSB という 2 つの PSoC ブロックにマッピングします。LSB ブロックの出力は MSB ブロックの入力に入ります。そのため、これらのブロックは相互に隣接した位置に配置されます。CY8C26/25xxx デバイスファミリでは、MSB ブロックは “ タイプ A" スイッチド キャパシタの PSoC ブロックにのみマッピングします。CY8C29/27/24/22xxx, CY8C23x33, CY8CLED04/08/16, CY8CLED0xD, CY8CLED0xG, CY8CTST120, CY8CTMG120, CY8CTMA120, CY8C28x45, CY8CPLC20, CY8CLED16P01, CY8C28x43, CY8C28x52 デバイスファミリでは、MSB ブロックは “ タイプ C" スイッチド キャパシタの PSoC ブロックにのみマッピングします。これにより直線性誤差が軽減されます。その理由は、これらのタイプのブロックが、LSB ブロックと MSB ブロックを結合する “BCap" キャパシタ（上図の C_4）のオフセット誤差を無効にする自動ゼロ化プロセスを許可するためです。

PSoC デバイスの CY8C26/25xxx ファミリで直線性と単調性の仕様に適合するには、MSB/LSB を ASA10/ASB20、ASA21/ASB11、あるいは ASA23/ASB13 に配置します。その他の配置は、直線性が低く、単調性はまったく得られない場合もあります。これは、デバイスリビジョン AC と AD による SC ブロック間のオンチップ信号配線が異なるためです。

配置場所の選定でもう一つ重要なことは、MSB クロックと LSB クロックを同じソースから抽出しなければならないということです。両クロックがアナログアレイの同じコラムに配置されていれば、これは自動的に実行されます。別々のコラムに配置されている場合、両コラムのマルチプレクサを同じソースに設定しなければなりません。

パラメータおよびリソース

DAC9 のインスタンスを生成するには、デバイスのエディタで、ユーザーのモジュールを選択し、必要な場合には名前を変更し、デバイスにマッピングします。配置時に考慮すべき注意点としては、信号がオフのチップに駆動される場合、そしてコラムクロックリソースの他のユーザーのモジュールとの相互依存的な場合には、アナログコラム出力バスの可能性を含むことができます。配置されると、ユーザ モジュールがパラメータを表示します。

DataFormat（データフォーマット）

DAC9 ユーザーモジュールの API は、3 種類の異なるデータ形式、つまり、オフセットバイナリ、2 の補数と符号とサイズのフォーマットを扱っています。API セクション (下) の WriteBlind エントリポイントには、このような習慣と、それぞれに関連する値の範囲が説明されています。

AnalogBus

DAC ブロックは、隣接している、アナログ PSoC ブロック側の出力デバイスをブロードキャストしています。アナログバスのオプションのいずれかを選択する場合は、アナログ出力バッファのいずれかを使用して、DAC 出力のデバイスが外部と接続されています。特定のコラムでバスを選択した場合、アレイの上部にある PSoC ブロックとの追加的な接続機能が提供されます。スイッチドキャパシタ PSoC ブロックには、Φ_2 の半分の間 DAC 出力デバイスを見本として抽出する一つの sample and hold 回路が組み込まれています。これは、自動的にゼロ作業の期間中発生する電圧のスイングから外付け出力デバイスを分離します。

ClockPhase（クロック位相）

このパラメータは、次のクロックとタイミングのセクションに議論されているコラムクリック分周器によって生成された位相のクロック、Φ_1 と Φ_2 の役割を判別します。*標準*が選択された場合、Φ_1 の間の自動ゼロサイクルが発生し、DAC 出力デバイスが Φ_2 で有効に機能します。クロック位相を *Swapped* に設定する場合は、これらの役割が逆になります。これは、DAC を Φ_1 からの入力デバイスを見本抽出する別の周辺機器に接続するときに便利です。

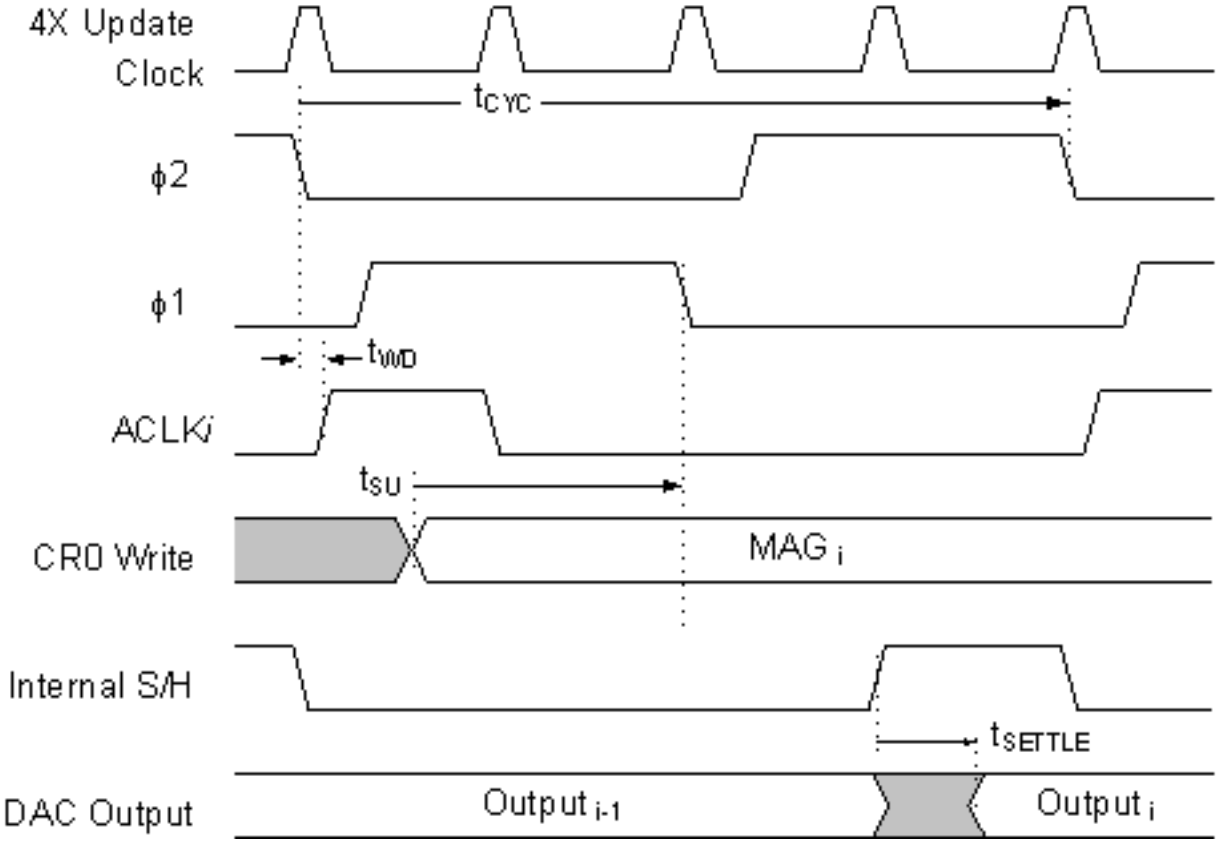
Note スワップにクロックフェーズを設定するサンプルを無効にし、アナログ出力バスの機能を保持します。アナログバスのパラメータを有効に設定する場合には、バスの出力がローカル PSoC ブロックの出力を表示し、Φ_1 の間 AGND (+ オフセット電圧) と Φ_2 の間、必要な出力を変更します。

アナログコラムのクロック

DAC は、適切な WriteBlind と WriteStall API 機能を呼び出すことにより、新しい値を " 作成 " するよう命令を受け取ったかどうかに関係なく、出力デバイスを連続的に更新します。アナログコラムクロックマルチプレクサは、これらの更新作業を制御する位相クロック、Φ_1 と Φ_2 を生成するために使用するソースクロックを選択します。位相クロック発生器は、Φ_1 と Φ_2 と生成するために、コラムのクロックを 4 つまで分割するのでコラムのクロック周波数は、実際のアナログ出力更新レートよりも 4 倍以上高速です。2 つのレベルの多重化を介して、すべてのデジタルブロックと、システムクロック分周期が含まれているコラムのクロックを選択することができます。上記の電気特性のセクションには、コラムのクロック周波数の下部と上部の限界値が指定されています。

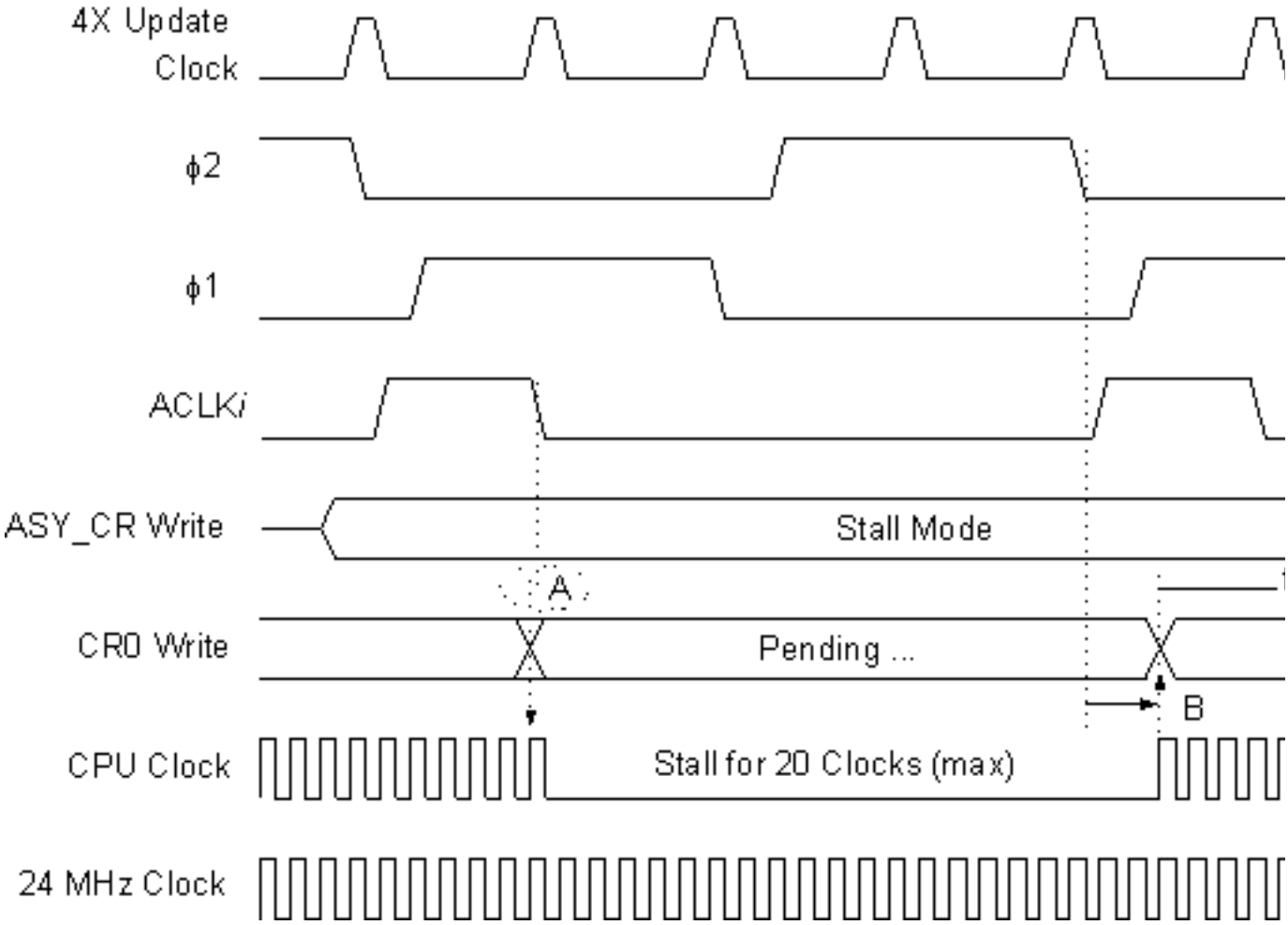
正の出力電圧 (Vout > AGND) と一般的な位相の場合、上記の単純なグラフに示されたように Φ_1 の間に基準電圧が A Cap に保存されます。A Cap を完全に充電するには、値の変化がセットアップタイム t_{SU} で Φ_1 下がり エッジまでの要件を満たしている必要があります。次の図に示されているこれらの設定時間は基準の電源レベルに応じて異なります。セットアップタイムが特徴化されていないにもかかわらず、ハードウェアのストールの構造を使用して、これらの要求を満たすように保証することができます。不適切なセットアップタイムの入力により、A Cap が完全に充電されていない場合、この設定時間が校正されている f1 全体のサイクルの次の位相クロックサイクルまで出力デバイスが校正されません。f2 の立下り エッジに関連する同様の設定時間は、負の出力電圧 (Vout < AGND) の動作を管轄しています。

Figure 3. V_{OUT} > AGND の更新タイミング



より大きいアプリケーションにおいて、瞬間的な (1 アップデートサイクル) 偏差は許容されています。他のアプリケーションは、より厳しい要求条件が付与されることができます。ハードウェアの同期機能は、A Cap 値を変更するレジスタをライトするタイミングを制御するために使用することができます。この機能は、入力内容によって WriteStall API 内で直接サポートされます。呼び出された場合、該当のハードウェアが PSoC ブロックのレジスタの作成の命令を認識し、CPU クロックを停止させ、Φ_1 の立ち上がりエッジまでのライト命令が完了されるのを留保します。
ASY_CR レジスタは、次の操作を制御します。

Figure 4. ハードウェアの同期機能、CPU クロックのストール



CPU ストール時に、すべてのアナログ及びデジタル PSoC ブロックが正常に機能します。この周期の間 DAC の CR0 レジスタをライトする MOV 命令が単純に停止され、割り込みされるかペンディングされている状態のままになります。同じストール時に CPU 周期数が失われるため、このとき、次の関係を使用して、周期を計算することができます。

Equation 5

$$\mathrm{CPU\,Cycles} \le \frac{F_{CPU}}{F_{\phi_1}} = \frac{4 \times F_{CPU}}{F_{ColClock}}$$

当然、ストール側の CPU サイクルの損失を最小限に抑えるには、コラムのクロックを実用的に最も高い周波数で駆動する必要があります。外付け出力の周期が、以前の出力周期を単純に繰り返すため、出力電圧が実際に変更するときに必要な値よりもはるか高くなることがあります。より迅速なコラムクロックは、出力機能の呼び出す時間から出力を変更する時間までの遅れを最小限に抑えます。

しかし、コラムのクロック周波数の実際的な限界が存在します。DAC は、AGND の出力電圧の各位相のクロック周期で変化するため、コラムクロックが出力デバイスが安定するように許可される周波数に制限されています。アナログ出力バスの Sample and hold 機能を使用する場合、オペアンプ出力デバイスが Φ_2 の周期後半で標本抽出のウインドウが表示されている、バスを駆動します。オペアンプが出力電圧に変化し安定化されるに十分にコラムクロックが速い場合は、これは、出力信号のノイズで観察することができます。極端な例では、出力デバイスは、見本抽出ウィンドウが閉まる前に最終的に出力電圧の一部としてのみ変化します。これは、出力の全範囲の終わりで最も明らかにされる深刻な非直線入力の圧縮値を観察することができます。これらの現象が発生することを防止するアナログコラムクロックのハイレベルの限界は、上記の電気特性表に示されています。

アプリケーション プログラミング インタフェース (API)

アプリケーション プログラミング インタフェース (API) ルーチンは、設計者が高級言語でモジュールを操作できるようにユーザ モジュールをコンポーネントとして提供します。このセクションでは、各機能に対するインタフェースを 「include」ファイルによって提供される関連定数とともに示します。

Note すべてのユーザ モジュール API の場合と同じように、API 関数を呼び出すことで A と X レジスタの値が変更されることがあります。A と X の値が呼び出し後に必要な場合は、呼び出し元関数で A と X の値を保存してください。PSoC Designer のバージョン 1.0 以降では、効率を高めるために、この 「registers are volatile (レジスタの揮発性)」ポリシーが選択され、実施されています。C コンパイラは、自動的にこの条件で処理されています。アセンブラ言語のプログラマは、コードがこのポリシーを遵守していることも確認する必要があります。一部のユーザ モジュール API 関数では A と X は変更されないこともありますが、将来も変更されないという保証はありません。

DAC9 ユーザーモジュールを初期化して、更新プログラムの値を作成し、ユーザーのモジュールを無効にするために、入力内容が提供されています。

DAC9_Start

説明：

このユーザ モジュールで必要なすべての初期化を実行し、スイッチド キャパシタ PSoC ブロックの出力レベルを設定します。

C プロトタイプ：

```
void  DAC9_Start(BYTE bPowerSetting)
```

アセンブラ：

```
mov   A, bPowerSetting
lcall  DAC9_Start
```

パラメータ：

bPowerSetting: 出力レベルを指定する 1 バイト。リセット及び構成されてから、DAC ブロックに割り当てられている PSoC ブロックの電源が遮断されています。C およびアセンブリで容易されたシンボル名、およびそれらに関連付けられた値は、以下の表で示されます。

記号名	値
DAC9_OFF	0
DAC9_LOWPOWER	1
DAC9_MEDPOWER	2
DAC9_FULLPOWER	3

戻り値：

なし

副作用：

DAC 出力デバイスが駆動されます。デフォルトで、初期値は AGND です。電源が供給されるとき、一部の他の出力値が必要な場合は、" スタート " を起動する前に、作動ルーチンのいずれかを呼び出します。A および X レジスタがこの機能により変更される場合があります。

DAC9_Stop

説明：

ユーザーモジュールの電源を遮断します。

C プロトタイプ：

```
void  DAC9_Stop(void)
```

アセンブリ：

```
lcall  DAC9_Stop
```

パラメータ：

なし

戻り値：

なし

副作用：

出力は駆動されません。A および X レジスタがこの機能により変更される場合があります。

DAC9_SetPower

説明：

DAC の 変換されたコンデンサブロックの PSoC ブロックの電源レベルを設定します。ブロックを遮断するときとつけるときに使用することができます。（これは Start 関数と同じです。）

C プロトタイプ：

```
void  DAC9_SetPower(BYTE bPowerSetting)
```

アセンブラ：

```
mov   A, bPowerSetting
lcall  DAC9_SetPower
```

パラメータ：

bPowerSetting: スタート入力内容のために使用される PowerSetting パラメータと同じです。

戻り値：

なし

副作用：

DAC 出力デバイスが駆動されます。デフォルトで、初期値は AGND です。電源が供給されるとき、一部の他の出力値が必要な場合は、" スタート " を起動する前に、作動ルーチンのいずれかを呼び出します。A および X レジスタがこの機能により変更される場合があります。

DAC9_WriteBlind

説明：

出力電圧を指示された値を直ちに更新します。

C プロトタイプ：

```
// For OffsetBinary:
void  DAC9_WriteBlind(INT iOutputValue)

// For TwosComplement:
void  DAC9_WriteBlind(INT iOutputValue)

// For TwoByteSignAndMagnitude in register format:
void  DAC9_WriteBlind2B(byte bLSB, byte bMSB)
```

アセンブラ：

```
; for OffsetBinary:
mov   X, >iOutputValue            ; upper byte
mov   A, <iOutputValue            ; lower byte
lcall  DAC9_WriteBlind
;or
mov   X, [piOutputValue]          ; upper byte
mov   A, [piOutputValue+1]        ; lower byte
lcall  DAC9_WriteBlind

; for TwosComplement:
mov   X, >iOutputValue            ; upper byte
mov   A, <iOutputValue            ; lower byte
lcall  DAC9_WriteBlind

; for TwoByteSignAndMagnitude format:
mov   X, bMSB                     ; MSB block CR0 register value
mov   A, bLSB                     ; LSB block CR0 register value
lcall  DAC9_WriteBlind
```

パラメータ：

iOutputValue: 出力電圧を指定する 2 バイト（それぞれ）。容認できる値は、下記に示す Data-Format の指定値に対応する範囲にあります。

データフォーマット	最小値	最大値
OffsetBinary	0	510
2 の補数	-255 (FF01h)	255 (00FFh)
TwoByteSignAndMagnitude	3F1Ch	1F3Ch

オフセットーバイナリ値は 0 まで表示される最小の出力電圧および 510 まで表示される最大の出力電圧を持つ正の数です。9-bit の 2 の補数では、上位バイトが正の値に対応する 00h、あるいは負の値に対応する FFh です。TwoByteSignAndMagnitude のフォーマットでは、上位バイトが $00smmmmm_2$ の形式を、下位バイトが $00tmmm00_2$ の形式となります。ここで 's' は符号、't' は反転符号、'm' はマグニチュードビットを表します。正の値では、s=0、t=1 です。

戻り値：

なし

副作用：

このユーザーのモジュールのタイミングのセクションで説明している理由のため、出力デバイスに障害が発生する可能性があります。A および X レジスタがこの機能により変更される場合があります。

注 : OFFSET_BINARY を選択する場合は、範囲外の入力値が API 内の 2 の補数データに自動的に変換されます。これは、可能な限り許可されたオフセットバイナリ値を超える範囲の値が小さい出力 (Agnd 付近) の値に変換されることを意味します。

DAC9_WriteStall

説明：

Φ_1 が起動されるまでのマイクロプロセッサーをストールした後、出力電圧を指示された値で更新します API は、割り込みが無効になったり、最大割り込み遅延が ACLK*i* 以下という事実を前提としています (図 「高速更新クロックを用いた強制同期化」を参照)。

C プロトタイプ：

```
// For OffsetBinary:
void  DAC9_WriteStall(INT iOutputValue)

// For TwosComplement:
void  DAC9_WriteStall(INT iOutputValue)

// For TwoByteSignAndMagnitude in register format:
void  DAC9_WriteStall2B(byte bLSB, byte bMSB)
```

アセンブリ：

```
; for OffsetBinary:
mov   X, >iOutputValue                ; upper byte
mov   A, <iOutputValue                ; lower byte
lcall  DAC9_WriteStall
;or:
mov   X, [piOutputValue]              ; upper byte
mov   A, [piOutputValue+1]            ; lower byte
lcall  DAC9_WriteStall
```

```
; for TwosComplement:
mov   X, >iOutputValue                ; upper byte
mov   A, <iOutputValue                ; lower byte
lcall  DAC9_WriteStall

; for TwoByteSignAndMagnitude format:
mov   X, bMSB                          ; MSB block CR0 register value
mov   A, bLSB                          ; LSB block CR0 register value
lcall  DAC9_WriteStall
```

パラメータ：

wOutputValue または iOutputValue: 形式と値の範囲が WriteBlind エントリポイント用に指定されているパラメータと同一。bMSB または bLSB: 形式と値の範囲が WriteBlind エントリポイント用に指定されているパラメータと同一。

戻り値：

なし

副作用：

ACLKi が無効 (‘i’ は、アナログ PSoC ブロックがマップされる列にある) になる場合、Φ_2 が無効になるまで、つまり、3/4 の更新プログラムのサイクルの間 (+ 2 つの CPU クロック) マイクロプロセッサーの CPU クロックが不能の状態がされています。ストールの間隔の間の割り込みが確認されないことに注意してください。A および X レジスタがこの機能により変更される場合があります。

ファームウェア ソースコードの例

次のサンプルコードは、一つのサイクルを発生させ、のこぎり波をつくります。

```
;;-------------------------------------------------------------------
;; Sample Assembly Code for the DAC9
;; Generate a falling sawtooth wave
;;-------------------------------------------------------------------

include "m8c.inc"
include "DAC9.inc"

          area bss (RAM)
iVal::    blk  2                 ; RAM for loop iteration variable
iMAXVAL:: equ  511               ; Top of ramp plus 1
          area text (ROM, REL)

_main::                          ; (contains infinite loop; never returns)
   mov  A, DAC9_LOWPOWER         ; specify DAC's amplifier power
   call DAC9_Start               ; and turn it on.

Init:
                                 ; Start ramp from the top
   mov  [iVal],   >iMAXVAL       ; upper byte
   mov  [iVal+1], <iMAXVAL       ; lower byte

RampDown:
   dec  [iVal+1]                 ; decrement lower byte
   jnc  skip_dec_wVal_upper      ; if lower goes from 00h to FFh, dec upper
   dec  [iVal]                   ; decrement upper byte
   jc   Init                     ; if upper goes from 00h to FFh, start over
```

```
skip_dec_wVal_upper:
   mov  X, [iVal]                ; upper byte
   mov  A, [iVal+1]              ; lower byte
   call DAC9_WriteStall
   jmp  RampDown                 ; next step
```

C での同じコードは以下のようになります。

```
//-------------------------------------------------------------------
// Sample C Code for the DAC9
// Generate a rising sawtooth wave.
//-------------------------------------------------------------------

#include "m8c.h"
#include "DAC9.h"

#if DAC9_OFFSETBINARY
    const INT  iMAXVAL=510;         // top of ramp
    const INT  iMINVAL=0;           // bottom of ramp
#else
    #if DAC9_TWOSCOMPLEMENT
        const INT  iMAXVAL=255;     // top of ramp
        const INT  iMINVAL=-255;    // bottom of ramp
    #else
        // DAC9_SIGNANDMAGNITUDE
    #endif
#endif


void main(void)
{
   INT  iVal;                    // RAM for loop iteration variable

  DAC9_Start(DAC9_FULLPOWER);   // power up the DAC

   if(DAC9_OFFSETBINARY)

   {
      while(1)
      {
         for(iVal=iMINVAL; iVal <= iMAXVAL; iVal++)
            DAC9_WriteStall(iVal);                    // update DAC
      }
   }
   else if(DAC9_TWOSCOMPLEMENT)
  {
      while(1)
      {
         for(iVal=iMINVAL; iVal <= iMAXVAL; iVal++)
         {
           DAC9_WriteStall(iVal);                    // update DAC
         }
      }
   }
```

```
    else
    {
        DAC9_Stop();                                    // power off DAC
    }
}
```

設定レジスタ

API は、DAC6 ユーザーモジュールの完全なインターフェースを提供します。設定レジスタに直接記述する場合、出力デバイスの代わりに更新することができます。どちらも出の障害を防止するために理解する必要があるタイミングの考慮事項が存在します。以下のレジスタが DAC9 スイッチド キャパシタ DAC ブロックのために使用されます。

Table 6. ブロック LSB: レジスタ CR0

ビット	7	6	5	4	3	2	1	0
値	1	0	符号	マグニチュード			0	0

符号では、正の値に 「1」(AGND ~ RefHi)、負の値に 「0」(RefLow ~ AGND) を使用します。デフォルトは 「1」です。これは MSB ブロックで使用されているセンシング機能と逆です。符号を変えるには、API の書き込み関数の 1 つを使用します。マグニチュードのデフォルトは 「0」です。マグニチュードを変えるには、API の書き込み関数の 1 つを使用します。

Table 7. ブロック LSB: レジスタ CR1

スイッチド キャパシタ タイプ A								
ビット	7	6	5	4	3	2	1	0
値	0	1	0	0	0	0	0	0
スイッチド キャパシタ タイプ B								
ビット	7	6	5	4	3	2	1	0
値	1	0	0	0	0	0	0	0

Table 8. ブロック LSB: レジスタ CR2

ビット	7	6	5	4	3	2	1	0
値	アナログバス	0	1	0	0	0	0	0

AnalogBus はデバイス エディタの構成時で有効または無効にされています。

Table 9. ブロック LSB: レジスタ CR3

スイッチド キャパシタ タイプ A								
ビット	7	6	5	4	3	2	1	0
値	0	0	1	1	0	0	電源	
スイッチド キャパシタ タイプ B								
ビット	7	6	5	4	3	2	1	0
値	0	0	1	1	1	0	電源	

電源：デフォルトは Off です。この値を設定するには、API の 「Start call」を使用します。

Table 10. ブロック MSB: レジスタ CR0

ビット	7	6	5	4	3	2	1	0
値	1	0	符号	マグニチュード				

符号は API 書き込みルーチンによって設定されます。マグニチュードを変えるには、API の書き込み関数の 1 つを使用します。

Table 11. ブロック MSB: レジスタ CR1

ビット	7	6	5	4	3	2	1	0
値	0	1	0	0	0	0	0	1

Table 12. ブロック MSB: レジスタ CR2

ビット	7	6	5	4	3	2	1	0
値	アナログバス	0	1	0	0	0	0	0

AnalogBus はデバイス エディタの構成時で有効または無効にされています。

Table 13. ブロック MSB: レジスタ CR3

ビット	7	6	5	4	3	2	1	0
値	0	0	1	1	BMux		電源	

BMux を構成して、LSB PSoC ブロックから接続を選択します。電源：0 = オフ、1 = 低、2 = 中、3 = フル。デフォルトは Off です。この値を設定するには、API の 「Start call」を使用します。

Copyright © 2001-2014 © Cypress Semiconductor Corporation. 本文書に記載される情報は、予告なく変更される場合があります。Cypress Semiconductor Corporation (サイプレス セミコンダクタ社) は、サイプレス 製品に組み込まれた回路以外のいかなる回路を使用することに対しても一切の責任を負いません。特許またはその他の権限下で、ライセンスを譲渡または暗示することもありません。サイプレス 製品は、サイプレスとの書面による合意に基づくものでない限り、医療、生命維持、救命、重要な管理、または安全の用途のために使用することを保証するものではなく、また使用することを意図したものでもありません。さらにサイプレスは、誤動作や故障によって使用者に重大な傷害をもたらすことが合理的に予想される、生命維持システムの重要なコンポーネントとしてサイプレス製品を使用することを許可していません。生命維持システムの用途にサイプレス 製品を供することは、製造者がそのような使用におけるあらゆるリスクを負うことを意味し、その結果サイプレスはあらゆる責任を免除されることを意味します。

PSoC® は Cypress Semiconductor Corp の登録商標であり、PSoC Designer ™ および Programmable System-on-Chip ™ は Cypress Semiconductor Corp. の商標です。本文書で言及するその他すべての商標または登録商標は、各社の所有物です。

全てのソース コード (ソフトウェアおよび / またはファームウェア) はサイプレス セミコンダクタ社 (以下「サイプレス」) が所有し、全世界の特許権保護 (米国およびその他の国)、米国の著作権法ならびに国際協定の条項により保護され、かつそれらに従います。サイプレスが本書面によりライセンシーに付与するライセンスは、個人的、非独占的かつ譲渡不能のライセンスであって、適用される契約で指定されたサイプレスの集積回路と併用されるライセンシーの製品のみをサポートするカスタム ソフトウェアおよび / またはカスタムファームウェアを作成する目的に限って、サイプレスのソース コードの派生著作物をコピー、使用、変更そして作成するためのライセンス、ならびにサイプレスのソース コードおよび派生著作物をコンパイルするためのライセンスです。上記で指定された場合を除き、サイプレスの書面による明示的な許可なくして本ソース コードを複製、変更、変換、コンパイル、または表示することは全て禁止されます。

免責条項： サイプレス は、明示的または黙示的を問わず、本資料に関するいかなる種類の保証も行いません。これには、商品性または特定目的への適合性の黙示的な保証か゛含まれますが、これに限定されません。サイプレスは、本文書に記載される資料に対して今後予告なく変更を加える権利を留保します。サイプレスは、本文書に記載されるいかなる製品または回路を適用または使用したことによって生ずるいかなる責任も負いません。サイプレスは、誤動作や故障によって使用者に重大な傷害をもたらすことが合理的に予想される生命維持システムの重要なコンポーネントとしてサイプレス製品を使用することを許可していません。生命維持システムの用途にサイプレス製品を供することは、製造者がそのような使用におけるあらゆるリスクを負うことを意味し、その結果サイプレスはあらゆる責任を免除されることを意味します。

ソフトウェアの使用は、適用されるサイプレス ソフトウェア ライセンス契約によって制限され、かつ制約される場合があります。