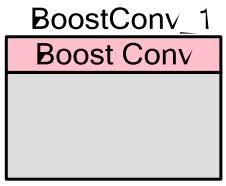
ブースト コンバータ（BoostConv）

1.50



特長

- 入力電圧よりも高く、選択可能な出力電圧を生成
- 入力電圧範囲は 0.5 V ～ 5.5 V
- ブースト済み出力電圧範囲は 1.8 V ～5.25 V
- 選択された入力と出力電圧のパラメータ値に基づいて、ソースは最大 50 mA
- アクティブとスタンバイの 2 種類の操作モード
- 自動サンプ付きブーストスタンバイモード操作は、PSoC3 ES2 と PSoC5 ES1 では非サポート。これらのデバイスでは、ブースト アクティブモード操作のみがサポートされます。

概要説明

ブースト コンバータ（BoostConv）コンポーネントは、PSoC ブースト コンバータ ハードウェアブロックを設定・制御する機能を提供します。ブースト コンバータにより、望ましいシステム電圧より低い入力電圧を、望ましいシステム電圧レベルにまで持ち上げることができます。コンバータは、入力電圧を望ましい出力電圧に変換するために、外付けインダクタを使用します。

BoostConv コンポーネントは、デフォルトでは選択出力電圧 3.3 V が有効になっており、BoostConv コンポーネントのパラメータは実行中に API を介して調整できます。コンポーネントカスタマイザで指定されている設定がデフォルト設定です。

ブースト コンバータには次の 2 つの主要な操作モードがあります。

- **アクティブ** – アクティブ モードは通常の動作モードで、ブースト レギュレータから安定した出力電圧が常時得られます。
- **スタンバイ** –スタンバイモードは、低消費電力操作モードです。

ブーストコンポーネントを使用するタイミング

BoostConv コンポーネントは、システムに利用できる電源が操作に必要な電圧レベルより低い場合に使用します。電池またはその他の入力電圧が使用でき、高い出力電圧を生成します。

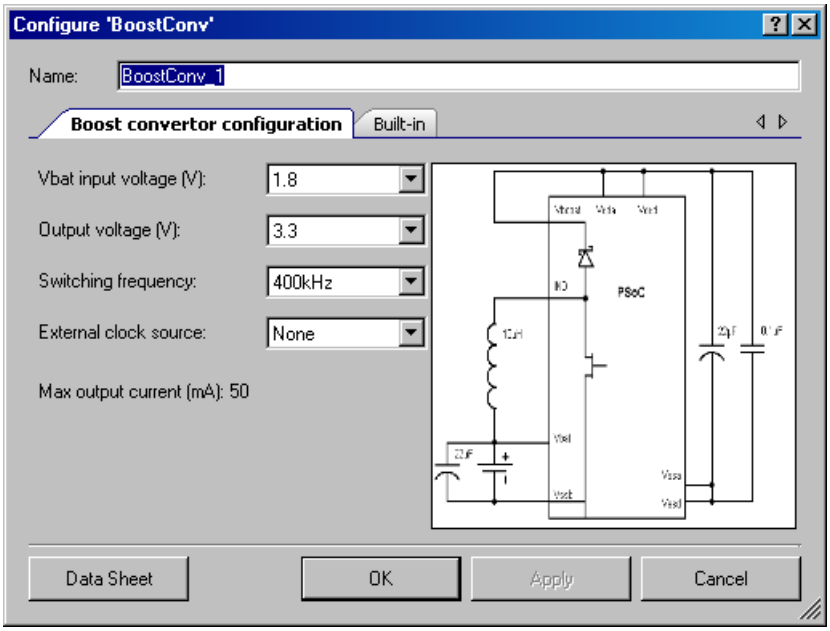
例えば、出力電源として 0.5 V の太陽電池を使用するシステムで、ブーストブロックを使用して 1.8 V PSoC のコアに電力を供給することができます。別の用例としては、3.3 V のシステムで、5.0 VLCD パネルに電力を供給するために BoostConv コンポーネントを使用するということも挙げられます。

入力/出力接続

BoostConv コンポーネントでは、プロジェクト回路図での接続は不要です。固定機能ピンがブースト コンバータ ブロック回路をサポートします。システム回路には、入力電圧（Vbat）、出力電圧（Vout）、インダクタピン（Ind）、電池アース（Vssb）に接続があります。「機能説明」のセクションに記載されている回路図を参照してください。

コンポーネント パラメータ

BoostConv コンポーネントをデザイン上にドラッグし、ダブルクリックして設定ダイアログを開きます。



Vbat 入力電圧

これは、ブースト コンバータ ブロックへの入力電圧として使用される電源 Vbat です。このシステム回路は、この電圧を Vbat PSoC に接続しています。入力電圧は 0.5 V～ 5.5 V です。この値は、最大出力電流の見積もりの計算に使用されます。デフォルト値は 1.8 V です。

出力電圧

これはブースト コンバータ ブロックが維持する目標出力電圧です。望ましい出力電圧をプルダウン メニューから選択します。出力電圧レベルは、1.8 V ～ 3.6 V の間では 0.1 V 単位で調整でき、4.00 V ～ 5.25 V の間では 0.25 V 単位で調整できます。デフォルト値は 3.3 V です。

出力電圧が 3.6V を超える場合は外付けショットキー ダイオードが必要です。

出力電圧値は、実行中に BoostConv_SelVoltage()関数を介して調整できます。

スイッチング周波数

これはブースト コンバータ ブロックが作動するスイッチング周波数です。この値は計数タイプで、次の周波数のうちいずれかを設定できます。

- 100 kHz
- 400 kHz
- 2 MHz
- 外付け 32 kHz

100 kHz、400 kHz、および 2 MHz のスイッチング周波数は、ブースト コンバータ ブロック内部の発振器を使用して生成されます。外付けの 32 kHz スイッチング周波数は、スタンバイモードの自動サンプ制御用です。

外付けクロック源

ブースト周波数の外付け 32 kHz は、ブースト コンバータ ブロックが外付けクロックを使用する設定になっている場合のスイッチング信号源です。この値は、次の周波数のうちいずれかを設定できます。

- なし
- ECO 32kHz
- ILO 32kHz

なし外付け 32 kHz クロック (ECO または ILO) のオプションは、ES3 シリコンでのみサポートされています。PSoC 3 ES2 または PSoC 5 ES1 シリコンの場合は「なし」を選択します。

最大出力電流

これは、指定の Vbat 値および出力電圧値に基づいたブースト コンバータ ブロックから使用できる最大出力電流の見積もりで、読み取り専用値です。

配置

BoostConv コンポーネントは、シリコンで専用のブースト コンバータ ハードウェアブロックを使用します。配置オプションはありません。

リソース

リソース	デジタル ブロック					API メモリ（バイト数）		ピン （外部入出力ごと）
	Datapaths	マクロセル	状態レジスタ	制御レジスタ	カウンタ7	フラッシュ	RAM	
BoostConv 固定 HW *	N/A	N/A	N/A	N/A	N/A	450	2	N/A

* BoostConv コンポーネントは、シリコンで専用のブースト コンバータ ハードウェアブロックを使用します。

アプリケーション プログラミング インタフェース

アプリケーション プログラミング インタフェース（API）のルーチンを使うと、ソフトウェアを使用してコンポーネントを設定できます。下表は、各関数を列挙し、そのインターフェースについて説明しています。さらに、各関数に関する詳細な説明が続きます。

デフォルトでは、PSoC Creator はインスタンス名「BoostConv_1」を設計上のコンポーネントの最初のインスタンスに割り当てます。この名前は、識別子の構文ルールに従った一意の値に変更できます。インスタンス名は、すべてのグローバル機能名、変数、定数記号の接頭辞になります。読みやすいように、下表では「BoostConv」というインスタンス名を使用しています。

関数	説明
BoostConv_Start	BoostConvコンポーネントを起動し、ブーストブロックをアクティブモードにします。
BoostConv_Stop	BoostConv コンポーネントを無効にします。ブースト コンバータ回路への電源をオフにします。
BoostConv_EnableInt	ブーストブロックの低電圧割り込みの生成を有効にします。
BoostConv_DisableInt	ブーストブロックの低電圧割り込みの生成を無効にします。
BoostConv_SetMode	ブースト コンバータ モードをアクティブまたはスタンバイに設定します。
BoostConv_SelVoltage	ブースト コンバータが維持する目標出力電圧を選択します。

関数	説明
BoostConv_SelFreq	スイッチ周波数に、100 kHz、400 kHz、2 MHz（ブースト コンバータ ブロック内部への生成）、32 kHz （ECO-32 kHzの発振器チップからブースト コンバータ ブロックへ外部ソース）の4つの値のうち、1つを設定します。
BoostConv_EnableAutoThump	自動サンプモードを有効にします（ブーストブロックがスタンバイモードで、スイッチ周波数が32 kHzのときにのみ有効）。
BoostConv_DisableAutoThump	自動サンプモードを無効にします。
BoostConv_ManualThump	ブースト コンバータ スイッチトランジスタの単一パルスを強制します。
BoostConv_ReadStatus	ブーストブロック状態レジスタを返します。
BoostConv_SelExtClk	32 kHz周波数源（32 kHz ECO または 32 kHz ILO）を設定します。
BoostConv_ReadIntStatus	ブーストブロック割り込み状態レジスタの内容を返します。
BoostConv_Init	カスタマイザが提供した初期値を用いてBoostConvレジスタを初期化します。
BoostConv_Enable	この関数はブーストブロックを有効にします（アクティブモードのときのみ有効）。デフォルトではコンポーネントは有効です。
BoostConv_Disable	ブーストブロックを無効にします。

グローバル変数

関数	説明
BoostConv_initVar	ブースト コンバータが初期化されたかどうかを示します。変数は、0に初期化され、BoostConv_Start()が初めて呼び出されたときに1にセットされます。これにより、コンポーネントはBoostConv_Start()ルーチンへの最初の呼び出し後、再初期化なしに再起動できます。 コンポーネントの再初期化が必要な場合、BoostConv_Start() やBoostConv_Enable()関数の前に、BoostConv_Init()関数を呼び出すことができます。

void BoostConv_Start(void)

説明： BoostConvコンポーネントを起動し、ブーストブロックをアクティブモードにします。コンポーネントは、チップが起動したときにこの状態になっています。これが推奨されているコンポーネント操作開始の方法です。BoostConv_Start() は変数initVarを設定し、関数BoostConv_Init()を呼び出し、関数BoostConv_Enable()を呼び出します。

パラメータ: なし

戻り値： なし

副作用: 変数initVarがすでに設定されている場合、この関数は単に関数BoostConv_Enable()を呼び出します。

void BoostConv_Stop(void)

説明： BoostConv コンポーネントを無効にします。ブースト コンバータ回路への電源をオフにします。

パラメータ: なし

戻り値： なし

副作用: なし

void BoostConv_SetMode (uint8 モード)

説明： この関数はブースト コンバータ モードを設定します （アクティブまたはスタンバイ）。

パラメータ: (uint8) モード – ブーストブロックに操作モードを設定します。

モード	注
BoostConv_BOOSTMODE_ ACTIVE	アクティブモードでは、ブーストブロックは選択された出力電圧を保ちます。
BoostConv_BOOSTMODE_STANDBY	低消費電力の状態、バンドギャップとコンパレータ回路がアクティブです。自動サンプモードは、外付け32 kHzクロックとともに使用して、出力電圧を制御します。

戻り値： なし

副作用: スタンバイモードでは、この関数は自動サンプモードを有効にし、スイッチング周波数クロック源を32 kHz外付けクロックに設定します。

void BoostConv_SelVoltage (uint8 電圧)

説明： この関数は、ブースト コンバータが維持する目標出力電圧を選択します。

パラメータ: (uint8) 電圧 – ブースト コンバータ ブロック用の目標出力電圧。3.6 Vを超える出力電圧には外付けショットキー ダイオードが必要です。

電源設定	値	注
BoostConv_VOUT_OFF	0x00	Off - High-Z
BoostConv_VOUT_1_8V	0x03	1.8 V
BoostConv_VOUT_1_9V	0x04	1.9 V
BoostConv_VOUT_2_0V	0x05	2.0 V
BoostConv_VOUT_2_1V	0x06	2.1 V
BoostConv_VOUT_2_2V	0x07	2.2 V
BoostConv_VOUT_2_3V	0x08	2.3 V
BoostConv_VOUT_2_4V	0x09	2.4 V
BoostConv_VOUT_2_5V	0x0A	2.5 V
BoostConv_VOUT_2_6V	0x0B	2.6 V
BoostConv_VOUT_2_7V	0x0C	2.7 V
BoostConv_VOUT_2_8V	0x0D	2.8 V
BoostConv_VOUT_2_9V	0x0E	2.9 V
BoostConv_VOUT_3_0V	0x0F	3.0 V
BoostConv_VOUT_3_1V	0x10	3.1 V
BoostConv_VOUT_3_2V	0x11	3.2 V
BoostConv_VOUT_3_3V	0x12	3.3 V
BoostConv_VOUT_3_4V	0x13	3.4 V
BoostConv_VOUT_3_5V	0x14	3.5 V
BoostConv_VOUT_3_6V	0x15	3.6 V
BoostConv_VOUT_4_0V	0x16	4.00 V (外付けショットキー ダイオードが必要)
BoostConv_VOUT_4_25V	0x17	4.25 V (外付けショットキー ダイオードが必要)
BoostConv_VOUT_4_5V	0x18	4.50 V (外付けショットキー ダイオードが必要)
BoostConv_VOUT_4_75V	0x19	4.75 V (外付けショットキー ダイオードが必要)
BoostConv_VOUT_5_0V	0x1A	5.00 V (外付けショットキー ダイオードが必要)
BoostConv_VOUT_5_25V	0x1B	5.25 V (外付けショットキー ダイオードが必要)

戻り値： なし

副作用: なし

void BoostConv_SelFreq(uint8 周波数)

説明： この関数は、スイッチ周波数を4つの値のうちの1つに設定します。

パラメータ: (uint8) switch_freq: 望ましいスイッチ周波数

スイッチ周波数	注
BoostConv__SWITCH_FREQ_100KHZ	専用発振器を用いてブースト コンバータ ブロック内部に生成
BoostConv__SWITCH_FREQ_400KHZ	
BoostConv__SWITCH_FREQ_2MHZ	
BoostConv__SWITCH_FREQ_32KHZ	ECO-32kHz または ILO-32kHz から

戻り値： なし

副作用: なし

void BoostConv_SelExtClk(uint8 ソース)

説明： この関数は、32 kHz 周波数源（ECO-32 kHz または ILO-32 kHz のチップ）を設定します。

パラメータ: (uint8) ソース: 32 kHz 周波数源

名前	説明
BoostConv__EXTCLK_ECO	ECO-32 kHz のチップを32 kHz 周波数源として設定します。
BoostConv__EXTCLK_ILO	ILO-32 kHz のチップを32 kHz 周波数源として設定します。

戻り値： なし

副作用: なし

void BoostConv_EnableAutoThump(void)

説明： この関数は、自動サンプモードを有効にします。自動サンプモードは、ブーストブロックがスタンバイモードのときに使用できます。ブーストブロックのスイッチ周波数クロック源は、32kHz外付けクロックに設定します。このモードで、出力圧が選択値を下回る場合に、スイッチクロックの各エッジでブーストスイッチパルスを生成することにより、スタンバイブースト操作を実行します。

パラメータ: なし

戻り値： なし

副作用: なし

void BoostConv_DisableAutoThump(void)

説明： この関数は、自動サンプモードを無効にします。

パラメータ: なし

戻り値： なし

副作用: なし

void BoostConv_ManualThump(void)

説明： この関数は、ブースト コンバータ スイッチトランジスタの単一パルスを強要します。

パラメータ: なし

戻り値： なし

理論:

副作用: サンプは設定されると、1 ~500ns パルスを生成します。このルーチンでは、ブーストブロックBOOST_CR0レジスタのbit 7 「thump」ビットに‘0’を書き、次に‘1’を書きます。

uint8 BoostConv_ReadStatus(void)

説明： この関数は、ブーストブロック状態レジスタの内容を返します。

パラメータ: なし

戻り値： (uint8) ブーストブロック状態レジスタ: BOOST_SR:

ビット	名前	説明
7	BoostConv_RDY	セットすると、内部回路が初期化されたことを示します
6	BoostConv_START	セットすると、コンバータが起動モードであることを示します
5	–	予約
4	BoostConv_OV	1のときは過電圧限度を超える出力、0のときは限度内
3	BoostConv_VHI	1のときはvhigh限度を超える出力、0のときは限度内
2	BoostConv_VNOM	1のときは公称値を超える出力、0のときは公称値未満
1	BoostConv_VLO	1のときはvlow限度を超える出力、0のときは限度内
0	BoostConv_UV	1のときは低電圧限度を超える出力、0のときは限度内

副作用: なし

void BoostConv_EnableInt(void)

説明： この関数は、ブーストブロックの出力低電圧割り込みの生成を有効にします。

パラメータ: なし

戻り値： なし

副作用: なし

void BoostConv_DisableInt(void)

説明： この関数は、ブーストブロックの出力低電圧割り込みの生成を無効にします。

パラメータ: なし

戻り値： なし

副作用: なし

void BoostConv_ReadIntStatus(void)

説明： この関数は、ブーストブロック割り込み状態レジスタの内容を返します。

パラメータ: なし

戻り値： (uint8) ブースト割り込み状態レジスタ BOOST_SR2 bit 0: セットされているときは、ブースト出力低電圧イベントが発生したことを示します。

副作用: なし

void BoostConv_Init(void)

説明： カスタマイザ構成ダイアログ設定に従って、コンポーネントを初期化または再保存します。BoostConv_Init()を呼び出す必要はありません。それはBoostConv_Start() APIがこの関数を呼び出し、またこの関数がコンポーネント操作を開始するための推奨手段であるためです。

パラメータ: なし

戻り値： なし

副作用: カスタマイザ設定ダイアログに従って、すべてのレジスタの値が設定されます。

void BoostConv_Enable(void)

説明： アクティブモードのとき、この関数はブーストブロックを有効にします。コンポーネントはデフォルトで有効になっています。ハードウェアを起動して、コンポーネント操作を開始します。BoostConv_Init()を呼び出す必要はありません。それはBoostConv_Start() APIがこの関数を呼び出し、またこの関数がコンポーネント操作を開始するための推奨手段であるためです。

パラメータ: なし

戻り値： なし

副作用: なし

void BoostConv_Disable(void)

説明： この関数はブーストブロックを無効にします。

パラメータ: なし

戻り値： なし

副作用: なし

ファームウェア ソースコードの例

PSoC Creator の [Find Example Project] (プロジェクト例の検索) ダイアログには、図とコード例を含む多数のプロジェクト例があります。コンポーネント別の例を見るには、[Component Catalog] (コンポーネントカタログ) でダイアログを開くか、図の中のコンポーネントを開きます。一般的な例を見るには、[Start Page] (スタートページ) または [**File**] (ファイル) メニューでダイアログを開きます。必要に応じて、ダイアログの中の [**Filter Options**] (フィルタオプション) を使用して選択できるプロジェクトのリストを絞り込むこともできます。

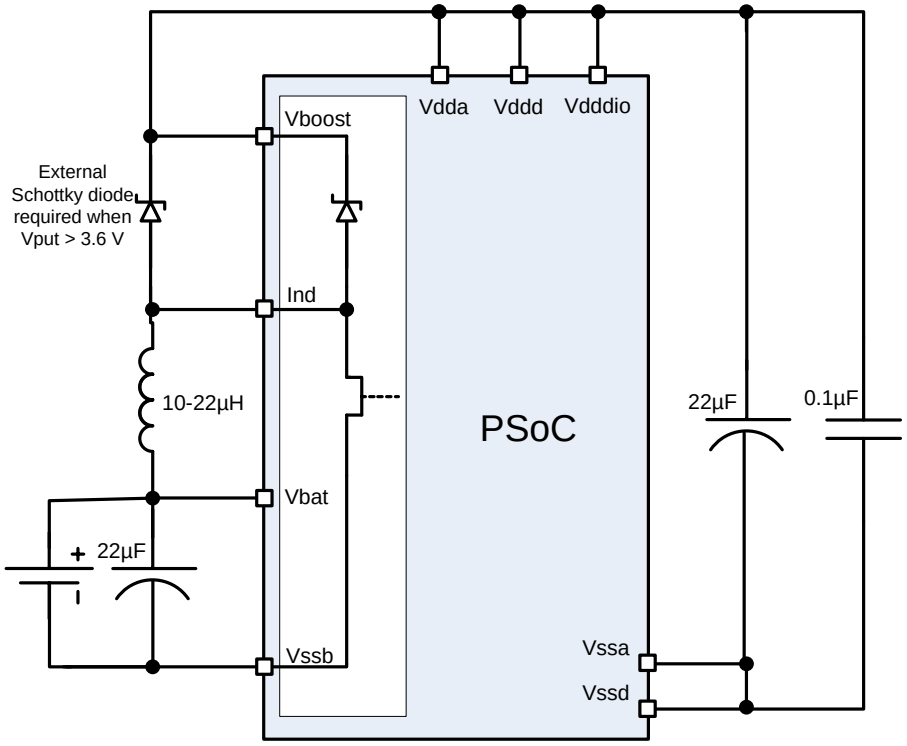
詳細については、PSoC Creator Help (PSoC Creator ヘルプ) の「Find Example Project（プロジェクト例の検索）」を参照してください。

割り込みサービスルーチン

なし

機能説明

図 1. ブーストコンバータの適用例

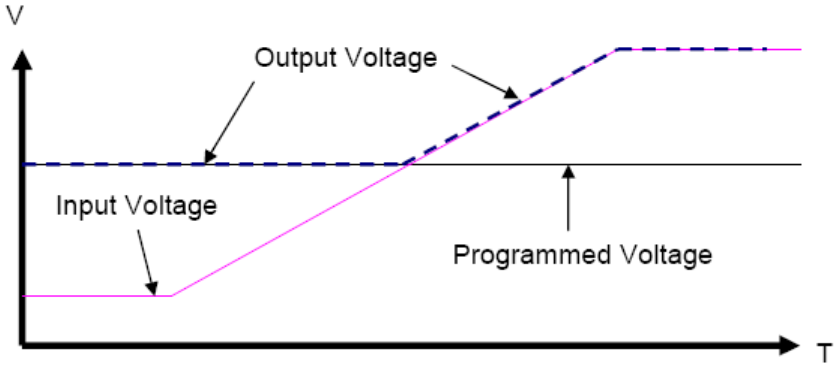


ブーストブロック回路は、デフォルトで、プロセッサの起動が Vboost 電圧で行われるというシナリオのサポートが有効です。ブーストブロックは、デフォルトで、1.8 V の出力電圧を伴ったアクティブモード用に設定されています。BoostConv コンポーネントをプロジェクトで使用すると、ハードウェアのブーストブロック用の構成レジスタにアクセスできます。関数 BoostConv_Start()は、コンポーネント設定ダイアログで行われた設定で BoostConv コンポーネントを構成します。

ブースト コンバータ ハードウェアは、上図に示すチップ上の固定機能ピンを活用します。これらの信号は、BoostConv コンポーネント上に示されていません。

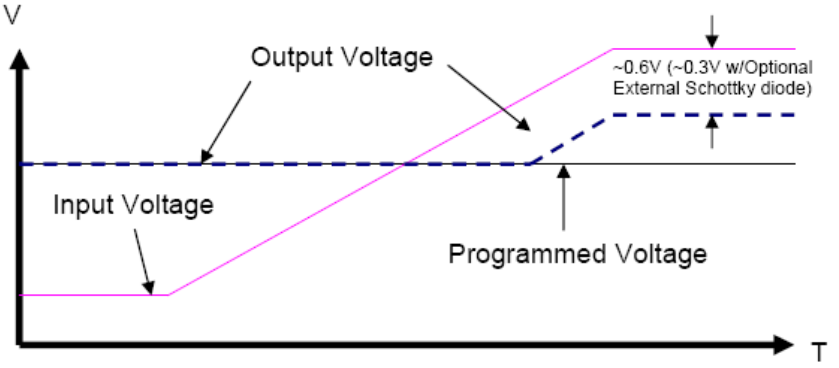
プログラムされた出力電圧を上回る入力電圧による操作

制御レジスタ 2 の場合 (BOOST_CR2): Bit 1 (eqoff) =1。入力がプログラムされた出力電圧を上回る場合に、出力電圧は入力電圧をトラッキングします。下図を参照してください。



- 出力電圧 = 入力 < プログラム電圧のときのプログラム電圧
- 出力電圧 = 入力 > プログラム電圧のときのプログラム電圧

制御レジスタ 2 の場合 (BOOST_CR2): Bit 1 (eqoff) = 1。出力電圧は、オプションの外付けショットキー ダイオード、またはインダクタピンと出力の間の内部シリコンダイオードに順方向バイアスがかかるまで、トラッキングしません。この結果、下記に示すように、出力電圧がダイオードの低下に伴って入力をトラッキングします。



- 出力電圧 = 入力 < プログラム電圧のときのプログラム電圧

- 出力電圧 = 入力電圧 – 入力 > プログラム電圧のときのダイオードの低下分 + ダイオードの低下分

DC 電気的特性と AC 電気的特性

次の値は予想されるパフォーマンスを示唆したもので、初期の特性データに基づいています。下表に別途記述がない限り、すべての T_A = 25°C, V_{dd} = 5.0 V となります。

インダクティブ ブースト レギュレータの DC 仕様

パラメータ	説明	条件	最小値	典型値	最大値	単位
V_{bat}	入力電圧	起動時を含む	0.5	–	5.5	V
I_{boost}	負荷電流 [1, 2]	V_{IN} = 1.6 – 5.5 V、V_{out} = 1.6 – 5.0 V、外付けダイオード	–	–	50	mA
		V_{IN} = 1.6 – 3.6 V、V_{out} = 1.6 – 3.6 V、内蔵ダイオード	–	–	75	mA
		V_{IN} = 0.8 – 1.6 V、V_{out} = 1.6 – 3.6 V、内蔵ダイオード	–	–	30	mA
		V_{IN} = 0.8 – 1.6 V、V_{out} = 3.6 – 5.0 V、外付けダイオード	–	–	20	mA
		V_{IN} = 0.5 – 0.8 V、V_{out} = 1.6 – 3.6 V、内蔵ダイオード	–	–	15	mA
L_{boost}	ブースト インダクタ	10 µH 指定値	4.7	10	47	µH
C_{boost}	フィルタ コンデンサ [3]	22 µF || 0.1 µF 指定値	10	22	47	µF
I_f	外付けショットキー ダイオードの平均順方向電流	外付けショットキー ダイオードは、V_{boost} > 3.6 Vの場合に必要	1	–	–	A
V_r	外付けショットキー ダイオードのピーク逆方向電圧	外付けショットキー ダイオードは、V_{boost} > 3.6 Vの場合に必要	20	–	–	V
I_{lpk}	インダクタのピーク電流		–	–	700	mA
	ハイバネート状態の電流	ブースト アクティブ モード	–	200	–	µA
		ブースト スタンバイ モード、32 kHzの外部水晶振動子、Iboost < 1 µA	–	12	–	µA
V_{boost}	ブースト出力電圧範囲[3, 4]					

[1] 出力電圧が 3.6V を超える場合は外付けダイオードが必要です。

[2] 最大出力電流は、出力電圧<入力電圧の 4 倍の場合です。

[3] デバイスの特性評価に基づく値（生産試験されたものではない）

[4] 2 MHz のブースト周波数では、Vboost は 2 x Vbat に限定されています。400 kHz では、Vboost は 4 x Vbat に限定されています。

パラメータ	説明	条件	最小値	典型値	最大値	単位
	1.8 V		1.71	1.80	1.89	V
	1.9 V		1.81	1.90	2.00	V
	2.0 V		1.90	2.00	2.10	V
	2.4 V		2.28	2.40	2.52	V
	2.7 V		2.57	2.70	2.84	V
	3.0 V		2.85	3.00	3.15	V
	3.3 V		3.14	3.30	3.47	V
	3.6 V		3.42	3.60	3.78	V
	5.0 V	外付けダイオードが必要	4.75	5.00	5.25	V
	負荷変動		–	–	2.5	%
	電源変動		–	–	3	%
	効率	VBAT = 2.4 V、VOUT = 2.7 V、IOUT = 10 mA、Fsw = 400 kHz、Lboost = 22 uH、Cboost = 22 uF || 0.01 uF	82	90	–	%

インダクティブ ブースト レギュレータの AC 仕様

パラメータ	説明	条件	最小値	典型値	最大値	単位
$V_{リップル}$	リップル電圧（ピークツーピーク）	V_{OUT} = 1.8 V、F_{SW} = 400 kHz、I_{OUT} = 10 mA	–	–	100	mV
F_{SW}	スイッチング周波数		–	0.1、0.4、2	–	MHz
	デューティ比		20	–	80	%

コンポーネントの変更

このセクションでは、旧バージョンからのコンポーネントの主要な変更について記載しています。

バージョン	変更点	変更の理由およびその影響
1.50.a	データシートの機能セクションに自動サンプサポートノートを追加	PSoC3 ES2 と PSoC5 ES1のシリコンバグ
	シリコン改定との整合性をアドバタイズするコンポーネントへの情報を追加	このツールは、コンポーネントが不整合のシリコン上で使用された場合にエラー/警告を発します。その場合、ターゲットデバイスをサポートする改訂版を更新してください。
	データシートに特性試験データを追加	
	スリープ モードへのリファレンスをデータシートから削除	コンポーネントはスリープ モードをサポートしません。
	データシートの細部修正および更新	

1.50	PSoC3 ES3 シリコンへのサポートを追加。3つのAPI関数を追加：
	void BoostConv_EnableInt(void);
	void BoostConv_DisableInt(void);
	uint8 BoostConv_ReadIntStatus(void);
		ブースト コンバータは低電圧信号の生成をサポートします。

	API関数を追加： void BoostConv_SelExtClk(uint8);	ブーストコンバータの外部スイッチングクロック源の選択肢 （ILO または ECO）をサポートするため。
	関数BoostConv_Init()を追加。	企業の標準に準拠し、コンポーネントを起動せずに初期化・再保存できるAPIを提供するため。

© Cypress Semiconductor Corporation, 2011-2014. 本文書に記載されている情報は、事前の予告なしに変更される場合があります。サイプレス セミコンダクタ社では、サイプレス製品に統合されている以外の回路の使用については、一切の責任を負いません。また、特許またはその他の権利に基づくライセンスを譲渡することも、含意することもありません。サイプレス製品は、サイプレスとの明示的な書面による合意がない限り、医療、生命維持、救命、重要な管理、または安全に関わる用途での使用を保証するものではなく、そのような使用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が重大な傷害を負うことが妥当に予測される、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は許可していません。生命維持システム用途にサイプレス製品を使用することは、製造者がそのように使用する上での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ます。

PSoC Designer™、Programmable System-on-Chip™、PSoC Express™ はサイプレス セミコンダクタ社の商標であり、PSoC®はサイプレス セミコンダクタ社の登録商標です。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。

すべてのソースコード（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って保護され、それらの規定に従います。サイプレスが本書面により被免許者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り、該当す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被免許者の製品のみをサポートするカスタム ソフトウェアやカスタム 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、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、本ソースコードのいかなる複製、変更、変換、コンパイル、または表示もサイプレスの明示的な書面による許可がない限り禁止されています。

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を含みますが、それらに限定されません。サイプレスは、本文書に記載した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、本文書に記載した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り生じる一切の責任を負いません。さらに、サイプレスは誤動作や故障によって使用者が重大な傷害を負うことが妥当に予測される、生命維持システムの重要なコンポーネントとしてのサイプレス製品の使用を許可していません。生命維持システム用途にサイプレス製品を使用することは、製造者がそのように使用する上での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、その規定に従います。