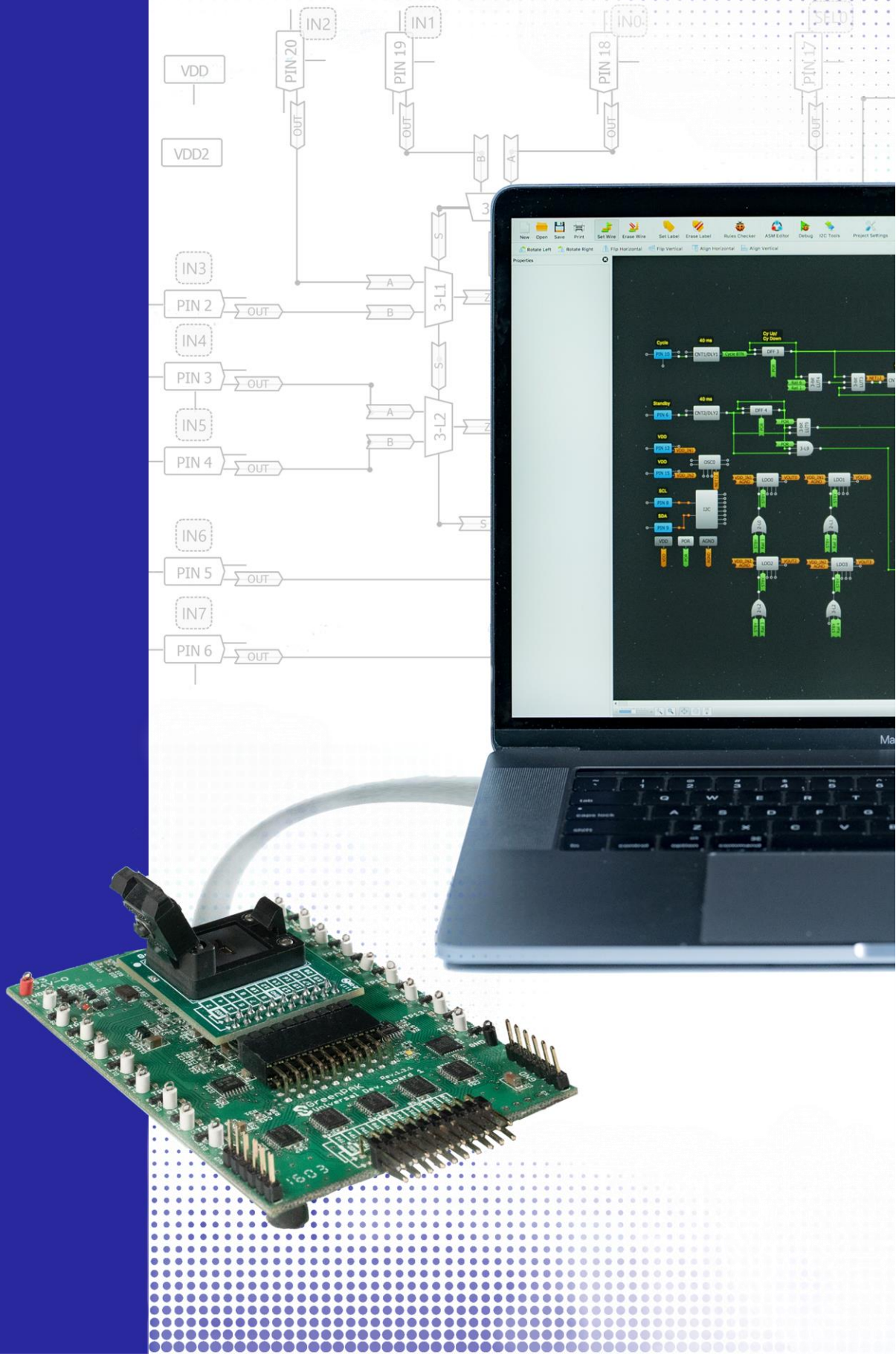
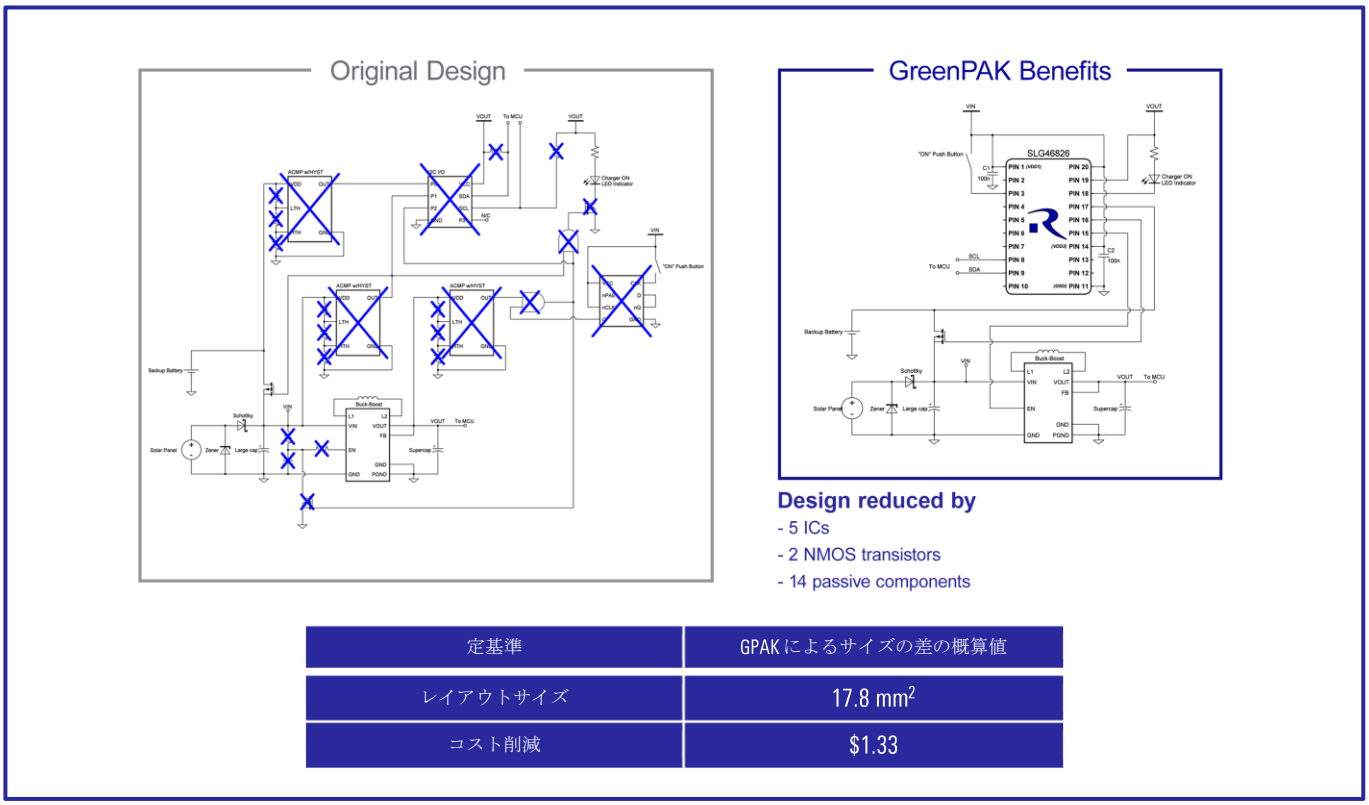
RENESAS

The GreenPAK™ Cookbook



GreenPAK の紹介

Renesas Electronics の GreenPAK IC は、システムレベルの回路開発者が直面する多くの共通の問題に対して、小規模 かつ手ごろなコストで独自のソリューションを提供する Configurable Mixed-signal IC（CMIC）製品です。 GreenPAK を使用することで PCB サイズ、BOM コスト、製品化の時間を大幅に抑えることができます。



定基準	GPAK によるサイズの差の概算値
レイアウトサイズ	17.8 mm^2
コスト削減	$1.33

GreenPAK を使用した規模縮小の例

GreenPAK は柔軟な設計を実現し、多くの機能を備えているため、様々な用途でお使いいただけます。適切な意図とノウハウに基づいて、 設計者はどの産業分野のどのような用途にも GreenPAK を利用することができます。 本書はこの意図とノウハウに対応するよう設計されていて、GreenPAK をプロジェクトのどこで使用できるかを設計者に示すための説明書となっています。本書では、設計者自身が GreenPAK を使用できるように、各種技法の概要を示し、アプ リケーションを詳しく説明しています。本書の設計をそのまま使用することも、本書の技法の一部を設計に組み込むこともできます。

説明書の構成

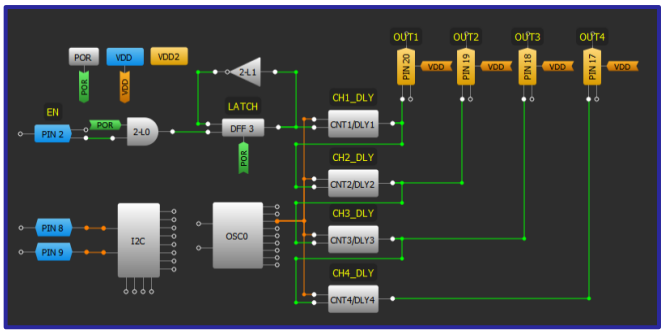
本書は主に、技法とアプリケーションの 2 つのタイプで構成されます。技法では、1 つまたは少数のマクロセルブロックのみを 使用して達成できるタスクに焦点を合わせます。アプリケーションセクションでは、技法を組み合わせて実際の価値あるアプリ ケーションを作成する方法について説明します。一般的に、容易な技法とアプリケーションから順に示します。

各アプリケーションには関連する GreenPAK Designer ファイルがあり、これを参照し、編集することができます。

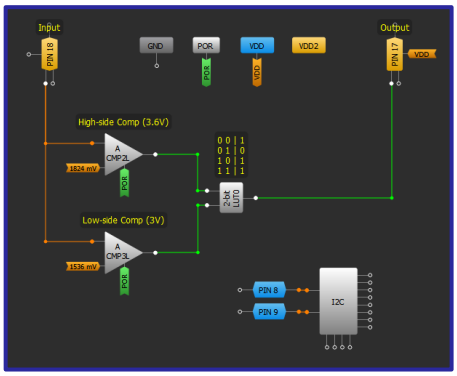
クックブック（実践回路集）を参考に独自の回路を作る

クックブックで概説されているアプリケーションは、実際のアプリケーションを単純に実現したものです。しかし、GreenPAK IC には、このクックブックで示されているデザインよりもはるかに多くの機能を実現するマクロセルと機能があります。ルネサス エレクトロニクスは、設計者が何千もの独自の設計を作成するのを支援してきました。クックブックのアプリケーションと似ていたり異なっていたりするシンプルなアプリケーションが拡張、結合、パーソナライズされています。

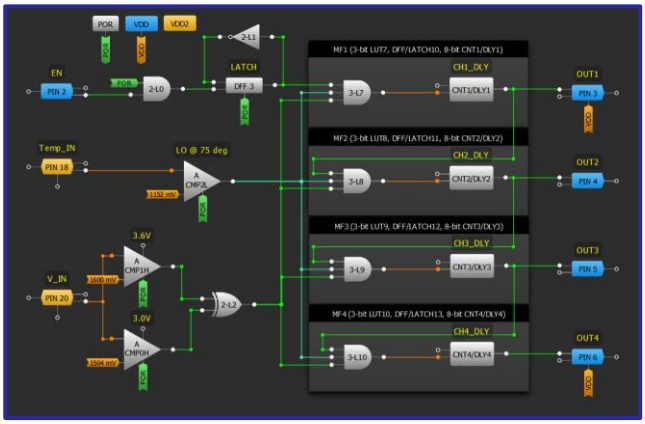
たとえば、「アプリケーション:基本シーケンサー」は「第 4 章:安全機能」内の多くのアプリケーションと組みあわて、自己制御型のカスタマイズされたシーケンシングアプリケーションを作成できます。



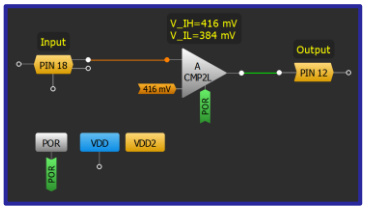
電源シーケンス



ウィンドウコンパレータ



ユニークソリューション



過熱検出

結果として得られる統合ソリューションはより複雑になりますが、使用できるすべてのマクロセルが使われているわけではありません。IC 内のすべてのマクロセルは自由に利用できるため、このクックブックのデザインで利用可能な置換と変更の数は無限にあります。このクックブックに示されているデザインを完全に再利用したい場合でも、このクックブックのテクニックの一部を独自のデザインに組み込む場合でも、遠慮なくご利用ください。

Contents

GreenPAK の紹介 2

説明書の構成 2

クックブック（実践回路集）を参考に独自の回路を作る 3

Contents 4

第 1 章: 基本ブロックと機能 8

技法：マクロセルについて 9

概要：デジタルマクロセル 9

技法：LUT マクロセルを使用した標準ロジックの構成 10

概要：オシレータ 10

概要：アナログコンパレータ 10

概要：I/O 11

概要：相互接続 11

技法:GreenPAK Designer を使用したシミュレーションとエミュレーション 12

技法: GreenPAK プログラミング 13

技法: OE ピン 14

アプリケーション：パリティビット発生器 15

アプリケーション: ワンホットエンコーダー 16

アプリケーション：8 ビットマルチプレクサ 17

アプリケーション: デマルチプレクサ 18

第 2 章: シーケンシャル ロジック 19

技法:CNT/DLY 精度の最適化 20

技法：CNT/DLY ブロックのシーケンス処理 21

アプリケーション：システムリセット 22

アプリケーション：複数ボタンのリセット 23

アプリケーション：基本シーケンサ 24

アプリケーション: カスケードシーケンサー 25

アプリケーション:電圧監視パワーシーケンサ 26

アプリケーション：出荷モードコントローラ 27

技法: ASM からの同期ステートマシンの作成 28

アプリケーション: N ビット長 ビットストリーム 29

技法: ビットストリームの多重化 30

アプリケーション：10 年カウンタ 31

アプリケーション: 方形波発生器 32

アプリケーション: 2 つのイベントボタンを押す 33

第 3 章：信号の調整 34

技法：CNT/DLY ブロックをデグリッチフィルターとして使用 35

技法: エッジ検出器 36

アプリケーション：割り込みコントローラ 37

技法: 双方向カウンタの作成.....38
アプリケーション：エンコーダ.....39
アプリケーション: 距離の計測.....40
アプリケーション: 周波数範囲検出器.....41
アプリケーション：周波数分周器.....42
技法：ゼロ電圧クロス検出.....43
アプリケーション: アナログストレージ素子.....44
技法：ACMP 電力消費量の削減.....46
技法: ウェイク/スリープのコントローラー.....47
アプリケーション：ウィンドウコンパレータ.....48
アプリケーション：過熱保護.....49
アプリケーション：バッテリー充電インジケータ.....50
アプリケーション: インフォテイメント用低電圧インジケータ.....51
アプリケーション: ウォッチドッグタイマー.....52
アプリケーション: 電圧レベル検出.....53
アプリケーション: 電源バックアップ管理.....54
アプリケーション: N パルス監視ウォッチドッグ.....55
技法: 温度センサーブロックの使用.....56
アプリケーション: 外部センス抵抗による電流検出.....57
アプリケーション: 1 つのアナログ信号の 4 つのレベルを監視する.....58
アプリケーション: 4 つのアナログ信号をモニターする.....59
技法：I2C を使用した設計の変更.....61
技法: I2C コマンドの生成.....62
技法: Serial to Parallel Interface (SPI) ブロックの使用.....63
技法：レベルシフト回路.....64
技法:プリセットされた数のパルスを送る.....65
技法：シフトレジスタの構築.....66
アプリケーション: I2C GPIO の拡張.....67
アプリケーション：シリアル/パラレル （外部クロック）.....68
アプリケーション：シリアル/パラレル （内部クロック）.....69
アプリケーション：パラレル/シリアル.....70
アプリケーション:双方向通信(送信優先).....71
アプリケーション:双方向通信(受信優先).....72
アプリケーション: ASM と I2C を用いた 7 セグメント表示.....73
アプリケーション: I2C を用いた通信 MUX.....74
アプリケーション: I2C レベルシフター.....75
アプリケーション:接続検出.....76
アプリケーション: カスタムパターン ジェネレータ.....77
技法: デューティーサイクル検出を使用したシリアルプロトコルの送信.....78
技法: シフトレジスタによるシリアルプロトコルの読み出し.....79
技法:パイプ遅延によるシリアルプロトコルの呼び出し.....80

技法: D/A コンバータ (DAC)の使用 81
技法: EPG 82
アプリケーション: ACK チェックとデータ比較を備えた I2C マスタ側 Read コマンド 83
アプリケーション: ACK チェックを備えた I2C Master Write Command 機能 84
アプリケーション: シフトレジスタを使用した I2C パターン生成 85
アプリケーション: EPG を使った長いパターン 86
アプリケーション: SPI マスターの基本 87
第 6 章: パルスベースの制御 **88**
技法：一定のデューティサイクル設定 89
技法: ワンショット実装 90
アプリケーション：定電流 LED ドライバ 91
アプリケーション: I2C による RGB LED 制御 92
技法: ブリージング LED パターンの作成 93
アプリケーション: ブリージング RGB LED 94
アプリケーション: I2C によるブリージング RGB LED 制御 95
技法: PWM モードでの DCMP/PWM マクロセルの使用 96
アプリケーション: PWM の選択 97
アプリケーション: ACMP と DAC を使用した PWM ジェネレータ 98
アプリケーション: ADC を使用した PWM ジェネレータ 99
技法: デューティ サイクル検出 100
アプリケーション: 周波数アナログ電圧コンバータ 101
アプリケーション: 周波数デューティサイクルコンバータ 102
アプリケーション: リニア周波数変調 103
アプリケーション: 電圧制御発振器 104
第 7 章: Power Management **105**
技法: 出力放電 106
アプリケーション：チャージポンプ 107
アプリケーション：2 ステージの チャージポンプ 108
アプリケーション: 出力調整付きチャージポンプ 109
技法: LDO レギュレータ 110
アプリケーション: フレキシブル電源アイランド 111
アプリケーション: 昇圧コンバータ 112
アプリケーション: 降圧 Converter 113
アプリケーション: Back-to-Back 接続 逆流防止回路 114
第 8 章: Motor Control **116**
アプリケーション：H ブリッジコントロール 117
技法: HV OUT CTRL ブロックの使用 118
技法: 通常モードでの SLG47105 PWM ブロックの使用 119
技法: プリセット レジスタ モードでの SLG47105 PWM ブロックの使用 120
アプリケーション: 定電圧ブラシ付きモータドライバー 121

アプリケーション: 定電流ブラシ付きモータドライバー 122
アプリケーション: PWM チョッパーを使用した定電流 123
アプリケーション: ソフト ON/OFF による単方向 DC モータ制御 124
アプリケーション:ソフト ON/OFF による双方向 DC モータ制御 125
アプリケーション: プッシュスタート/ホールドストップ 126
アプリケーション: バイポーラステッピングモータドライバ 127
第 9 章: Advanced Analog Features 128
アプリケーション: オペアンプを使った可変アクティブフィルタ 129
アプリケーション：可変反転オペアンプ 130
アプリケーション：可変非反転オペアンプ 131
アプリケーション: 計装アンプ 132
アプリケーション: オペアンプを使ったボルテージフォロア 133
アプリケーション：オペアンプとＰチャネル FET を使用した電流源 134
アプリケーション：オートトリム 135
アプリケーション：オペアンプと N チャネル FET を使用した電流シンク 136
アプリケーション: OpAmp を使った電圧レギュレータ 137
技法: チョッパー ACMP をデジタル可変抵抗と一緒に使う 138
アプリケーション：サンプル・ホールド回路 139

第 1 章: 基本ブロックと機能

この章では、クックブック全体で使用される GreenPAK の基本的な機能ブロックの多くについて紹介します。また、ルックアップ テーブル (LUT) を利用する単純な組み合わせロジックの設計例もいくつか紹介します。

技法：マクロセルについて

この技法は GreenPAK Designer の任意のバージョンで有効です。

このセクションではよく使用される GreenPAK のブロックの概要といくつかの技法を示します。 ただし、特定のマクロセルについて詳しく知りたい場合は、GreenPAK Designer でマクロセル を選択し、[Properties]ウィンドウの左下にある情報ボタン（図 2）をクリックしてください。

情報ボタン

概要：デジタルマクロセル

デジタルマクロセルは GreenPAK の基本的な機能コン ポーネントです。これには、以下が含まれます。

共通デジタルマクロセル：

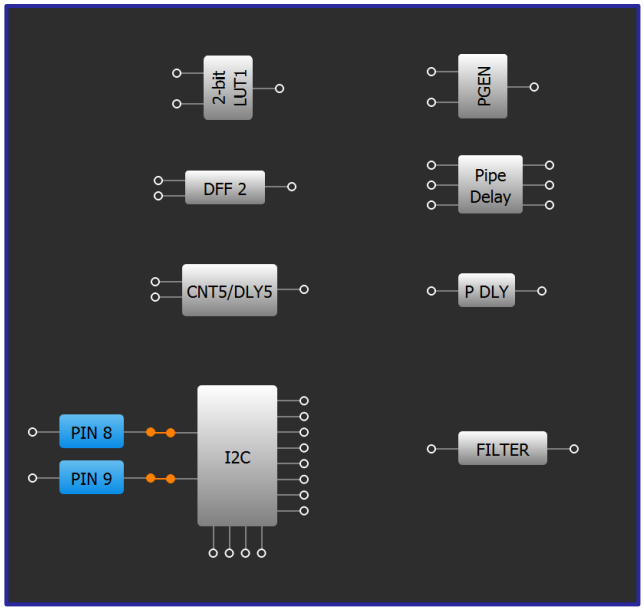
- ルックアップテーブル（LUT）
- D フリップフロップ（DFF）/ラッチ
- カウンタ/遅延（CNT/DLY）

通信：

- I2C（多デバイス）
- SPI（いくつかのデバイス）

共通性の低いもの：

- パターンジェネレータ（PGEN）
- パイプディレイ
- プログラマブルディレイ（PDLY）
- フィルター/エッジディテクタ



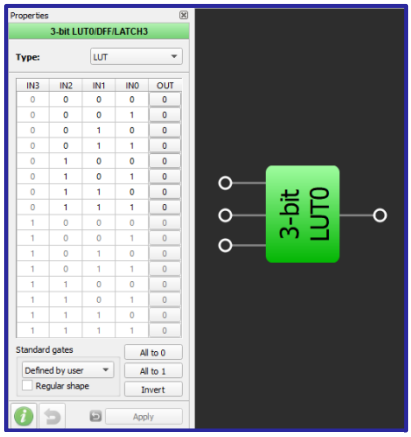
デジタルマクロセル

GreenPAK Designer のコンポーネントの多くは複数のタイプのマクロセルのいずれかとして選択することができます。これはデジタルマクロセルの名前によって示されます。たとえば、「2-bit LUT0/DFF/LATCH0」は名前が示すように、LUT、DFF、ま たはラッチであることがわかります。マクロセルのタイプの選択は[Properties]ウィンドウの[Type]オプションを使用して設定さ れます。

技法：LUT マクロセルを使用した標準ロジックの構成

この技法は任意の GreenPAK で有効です。

GreenPAK Designer でルックアップテーブルを使用して 2 つ、3 つ、または 4 つの 入力のデジタルロジック、単一出力のロジックマクロセルを構成することができま す。ロジック構成は[Properties]ウィンドウで編集できます。 GreenPAK の設計に実装された大部分のロジックは MUX、AND、OR などの標準 ロジックです。これらの共通の構成を効率的に処理できるように、[Properties]ウィ ンドウには、ロジックテーブルを標準のゲート構成に自動的に変換するための [Standard gates]オプションが用意されています。[Regular shape]オプションが 選択されていない状態では、LUT 形状は標準のゲート記号に変わります。



3 ビット LUT0 の構成

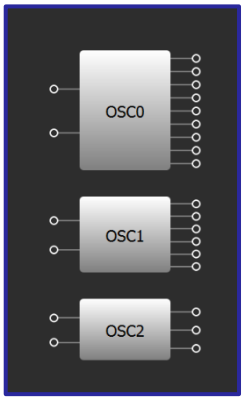
概要：オシレータ

GreenPAK IC には少なくとも 2 つのオシレータが装備され、SLG46826 のような最近のモデルに は 3 つのオシレータが装備されています。GreenPAK のオシレータの一般的な分周前周波数は 次のとおりです。

- 2KHz の低速、低出力のオシレータ
- 2MHz の中速
- 25MHz の高速

オシレータにはいくつかの出力が備わっており、それぞれ柔軟なクロック設定が可能ないくつかのプリ ディバイダを備えています。電力を節約するため、Auto-power on により、クロックが不要な場合 にオシレータをオフにすることができます。

オシレータについての詳細情報は、コンポーネントの選択時に情報ボタンをクリックすると表示されます。

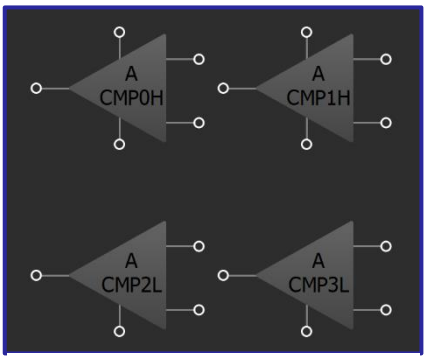


オシレータ

概要：アナログコンパレータ

ほぼすべての GreenPAK に 2 つ以上のアナログコンパレータ（ACMP）が装備 されていて、それぞれに 2 つの入力源 IN+および IN-が装備されています。それぞれへの入力源は[Properties]ウィンドウで設定できます。

アナログコンパレータについての詳細情報は、コンポーネントの選択時に情報ボ タンをクリックすると表示されます。

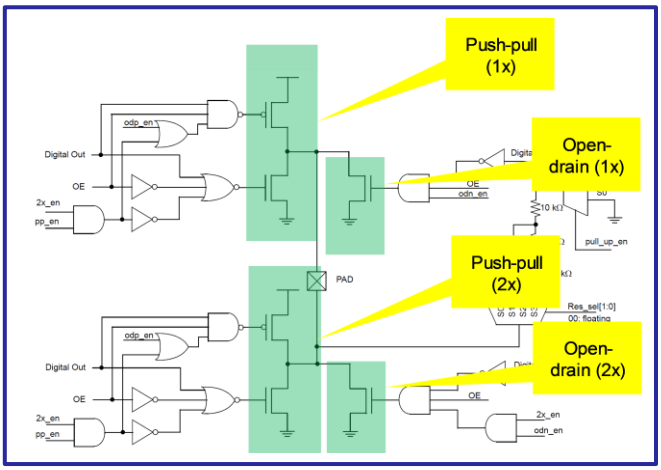


ACMPs

概要：I/O

GreenPAK のほとんどの I/O は非常に高い柔軟性を備えています。I/O 機能はピンごと、パーツごとに大きく異なる ため、特定の GreenPAK IC に決定する前に、必要なピ ン構成に合わせて設計コンセプトを規定する必要があり ます。

出力は NMOS または PMOS 構成でプッシュ/プル、またはオープン/ドレインとして構成できます。たとえば、ス ケール係数の 2x は出力強度が 2 倍であることを示しま す。また、10kΩ、100kΩ、1MΩ のプルアップおよびプル ダウン抵抗オプションを入出力ピンの設定により利用でき ます。



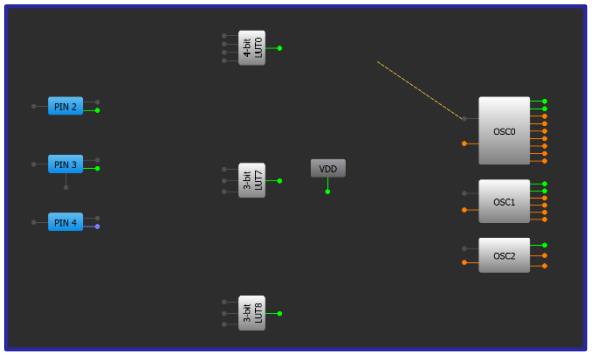
通常の I/O 構成

複数の入力オプション、たとえばデジタル入力、シュミット トリガーを使用したデジタル入力、定電圧デジタル入力、 アナログ入力が利用できます。アナログ入力は ACMP への入力として使用されます。

概要：相互接続

GreenPAK Designer との相互接続は容易です。システムによ り、可能な接続にガイドされます。いずれかの接続ポイントをク リックすると、以下が行われます。

- 可能なすべての接続が緑色で強調表示されます。
- これらの任意の緑色の接続まで伸ばすことができる「ゴム バンド」接続が割り当てられます。
- 相互接続されると接続が緑色のワイヤーで示されます。



相互接続

技法:GreenPAK Designer を使用したシミュレーションとエミュレーション

エミュレーションはすべての GreenPAK IC で利用でき、シミュレーションは多くの GreenPAK IC で利用できます。

GreenPAK Designer のツールバー

デザインを開発するときは、機能をすばやくテストできることは重要です。 GreenPAK Designer は、デバッグを効果的かつ簡単にします。

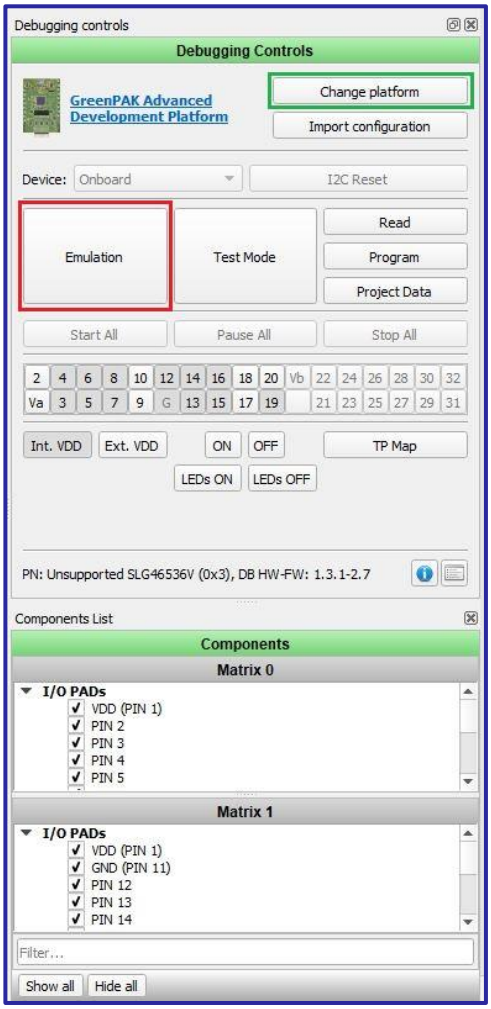
デザインをすばやく確認する方法は 2 つあります:

1. シミュレーション
2. エミュレーション

シミュレーションは、物理的な IC を必要とせずに、現実の条件で回路の動作をシミュレートします。シミュレーションは現実のシステムのすべての状況を提供できるわけではないことに注意してください。

エミュレーションを使用すると、デモ ボードと GreenPAK チップがあれば、パーツを永続的にプログラミングすることなく、ハードウェアで直接、設計の動作を確認できます。 これにより、プロジェクトにすばやく変更を加え、エミュレーションを使用して予想通りの動作かを確認できます。

1. デザインをデバッグする準備ができたら、[デバッグ] ボタン (上の図の赤いボックス) を選択して、エミュレーション/シミュレーション選択メニューに移動します。
2. 次に、デザインを確認するプラットフォームを選択します。
3. プラットフォームを選択したら、デバッグ メニューに移動します。そこでは、選択したプラットフォームに応じて、さらにアクションが行えるようになっています。
4. プラットフォームを変更する必要がある場合は、[Change Platform] を選択していつでも変更できます。

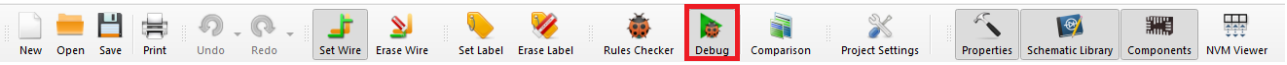


デバッキングメニュー

プラットフォーム選択メニュー

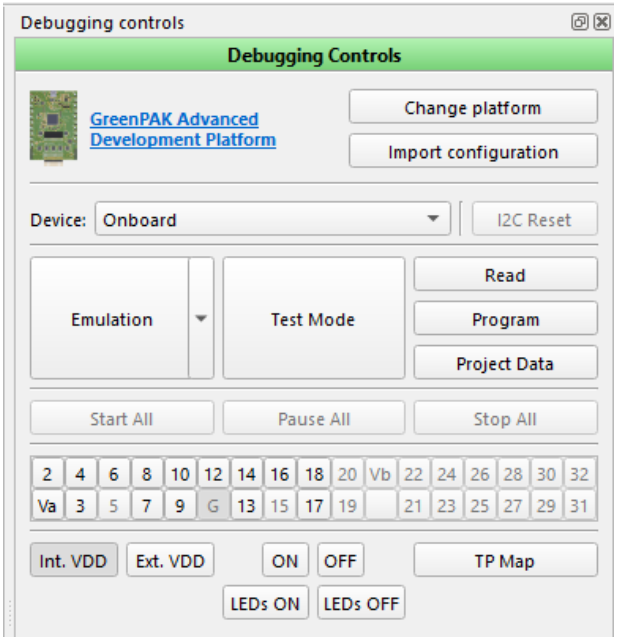
技法: GreenPAK プログラミング

全ての GreenPAKs に Debug 機能が用意されています。



GreenPAK Designer のツールバー

設計を行うとき、機能を素早くテストできることは重要です。GreenPAK デザイナーは、デバッグを効果的で簡単にします。



Change platform
テストに使用するハードウェアプラットフォームを選択します。

Import configuration
他のプラットフォームから、テストポイント等ユーザー設定の取り込みを行います。

Device
外部に実装された個別のデバイスアドレスのチップを操作することにも対応します。.

I2C Reset
デバイスと I2C のシリアル通信が確立されているとします。その場合、全てのマクロセルの構成と接続マトリクスによって提供されるすべての接続を含むデバイスを初期電源投入状態にリセットすることができます。これを実行するには、I2C リセットレジスタビットを“1”にします。これにより 、NVM から全てのレジスタデータのリロードを含む、デバイスのパワーオンリセット(POR) シーケンスが再度有効になります。

Emulation

- 現在のプロジェクトのエミュレーションがチップにロードされ（ただし、プログラムは行われません）、ハードウェアボードでのテストの準備が整います。
- エミュレーション (同期)。エミュレーションに加えて、プロジェクトで行われた各変更はすぐにチップにロードされます。

Test mode
テストモードは、チップの I/O パッドをユーザーが設定した TP コントロールに接続または切断するために使用されます。また、ユーザーはエミュレーション無しでテストモードを使用してプログラムされたチップをチェックできます。これを行うには、テストモードと内部 VDD ボタンをオンにします。テストモードは、チップに電力が供給されていなくても機能します。ユーザーは手動で電源を制御します。

Read
ハードウェアボードを使ってチップのデータを読み出します。

Program
現在のプロジェクトをチップに書き込み（プログラミング）ます。一部のチップモデルでは、ユーザーは Program ボタンの書き込みオプションから以下の選択が可能です。:

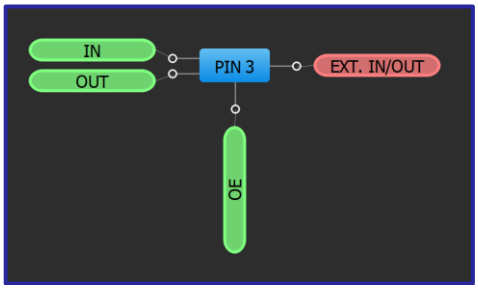
- Program NVM (チップの NVM への書き込み)
- Program EEPROM (チップの EEPROM への書き込み)

Project Data
NVM と EEPROM のビットの一覧表（特定のチップレビジョンで利用可能)

技法: OE ピン

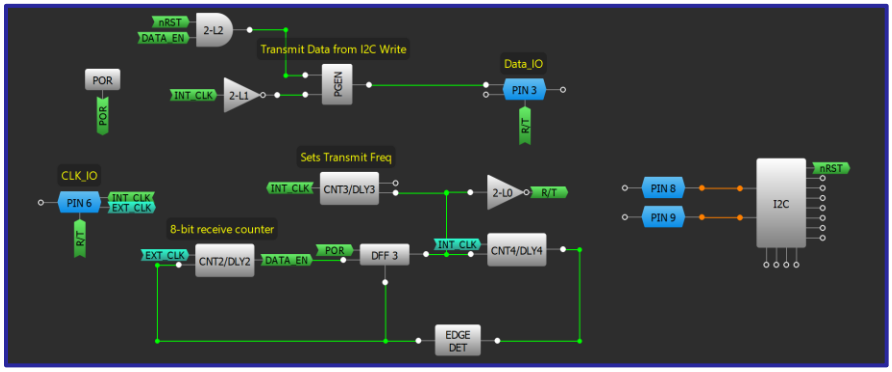
この技法は、OE ピンを備えた GreenPAK で使用できます。

通常、GreenPAK I/O は入力または出力として設定されます。出力**イネーブル(OE)**ピンは、ほとんどの GreenPAK 内にある選択ピンであり、デジタル入力とデジタル出力の間でピンを動的に変化させることができます。OE GPIO が固定入力として設定されている場合、OE ピンはグランドに設定され、GPIO が固定出力として設定されている場合、OE ピンは VDD に設定されます。GPIO を「Digital input/Output」として設定すると、マトリックス内でこの選択を行うことができます。

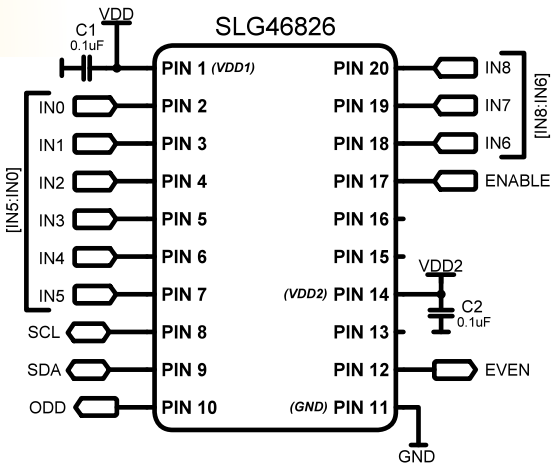


GPIO を Digital input/Output として設定すると、双方向通信が可能になります。また、GPIO を 論理的な High と Low に加えて Hi-Z に設定することもできます。

GPIO を双方向通信に使用する場合は、OE 選択用のタイミング回路を実装することが重要です。以下の回路例では、CNT2 が 8 つのクロックを確認した後、CLK_IO と Data_IO の OE ピンが Low から High に切り替わり 、その結果、内部信号を送信するための出力として OE ピンが設定されます。さらに 8 クロック後、OE ピンは Low にリセットされ、外部信号を受信するために再び入力として設定されます。



アプリケーション：パリティビット発生器

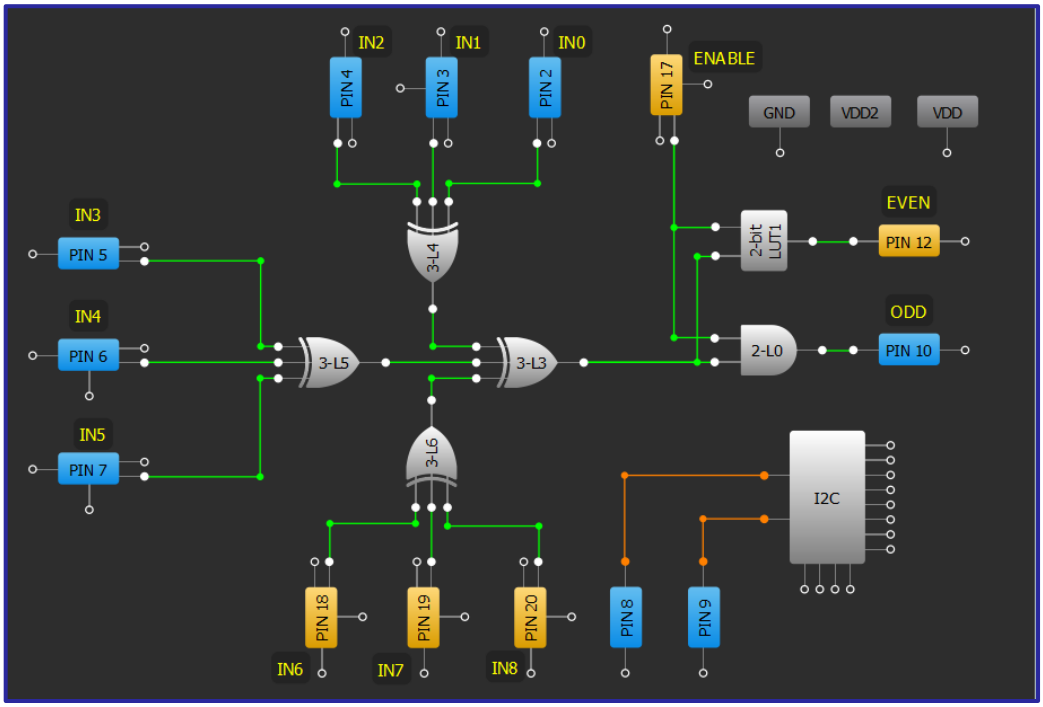


パリティビット発生器は信号の完全性をチェックするために使用されま す。巡回冗長チェック（CRC）を簡単に実装できます。パリティビットは、 入力データが壊れていないことを確認するため、MCU または他の制御 装置にデータを送信する前に使用されます。

必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

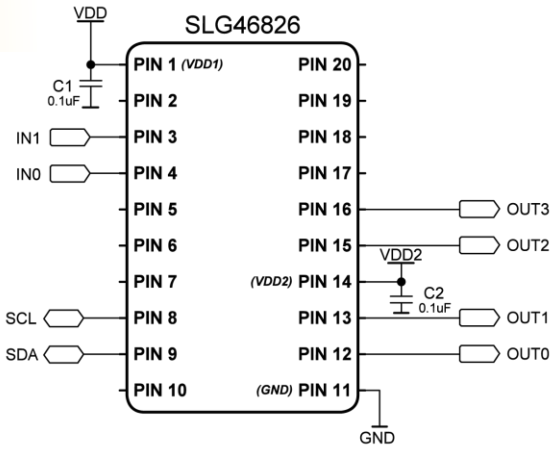
GreenPAK の図



設計手順

1. 技法：LUT マクロセルを使用した標準ロジックの構成に従い、XOR ゲートを使用して入力ピンを接続します。XOR ゲートは論理“1”の実行合計を計算するために使用できます。
2. ENABLE 信号のロジックを追加します。

アプリケーション: ワンホットエンコーダー

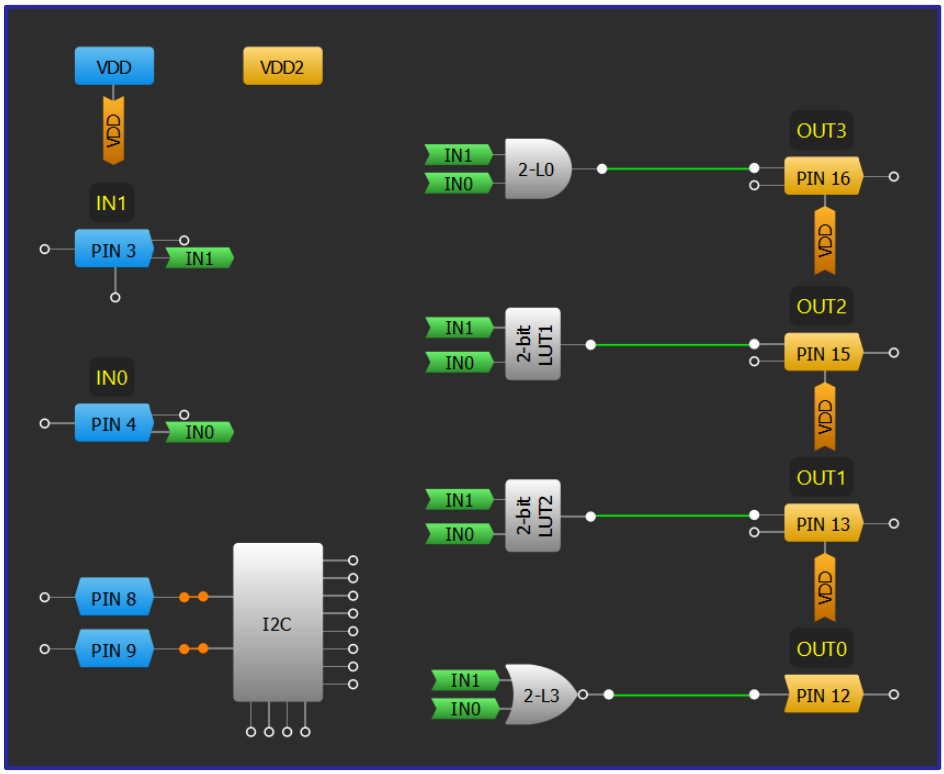


ワンホットは、単一の HIGH (1) を含み、他のすべてのビットが LOW (0) に設定されたビットのグループ、例えば 0001 です。 1 の集まりと単一の 0 を持つ逆の実装は、ワンコールドと呼ばれます。 この設計は、入力の 2 ビットコードに基づいて、指定された 1 ホット出力を出力するエンコーダーです。

必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

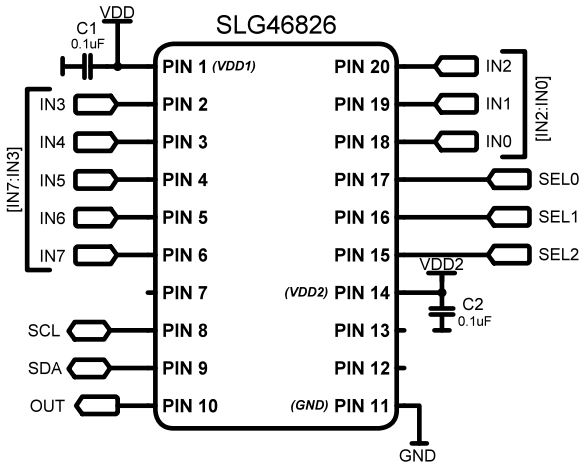


設計手順

1. 2 つの入力ピン (INx) と 4 つの出力ピン (OUTx) を構成します。
2. 各 LUT がアクティブな HIGH 出力用の一意の 2 ビット入力コードを持つように 設定します。たとえば、IN0 と IN1 の両方が LOW の場合、2-L3 は HIGH になります。

アプリケーション：8 ビットマルチプレクサ

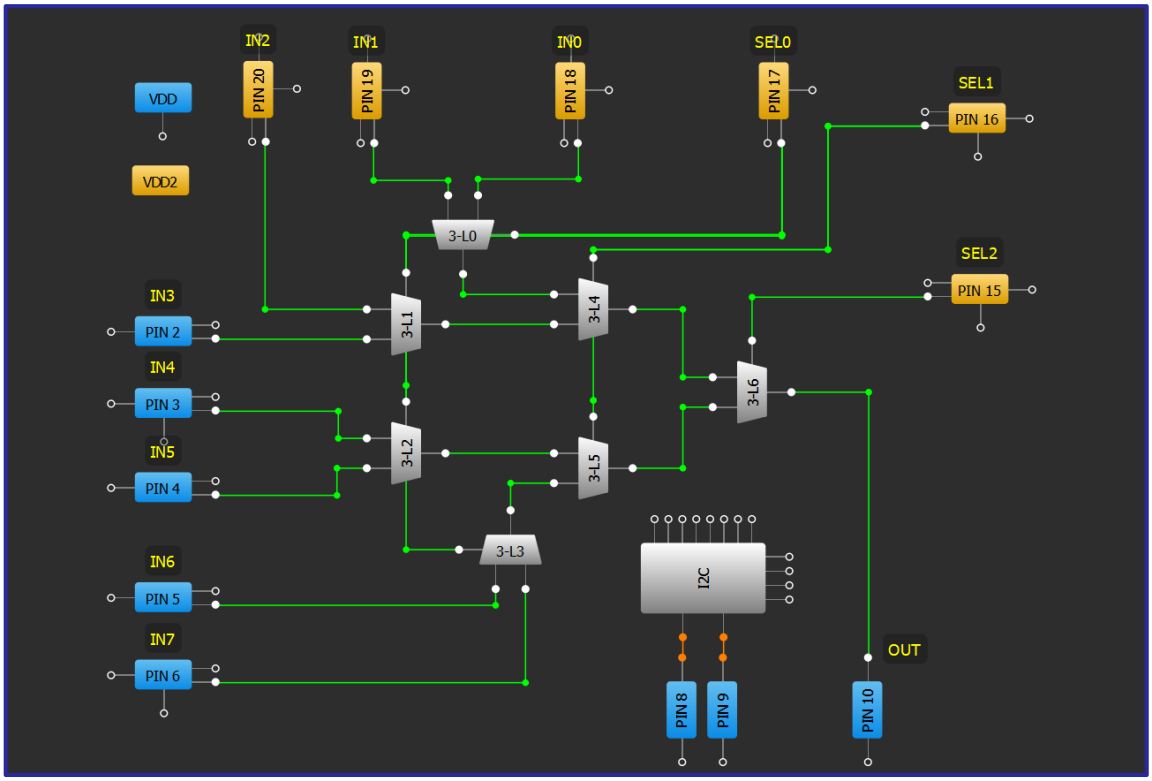
マルチプレクサ（MUX）は複数の入力信号から 1 つの出力を選択するために使用されます。これは、1 つの回線で複数の通信回線を送信する必要があるアプ リケーションで使用されます。GreenPAK を MUX として使用することにより、送 信の待ち時間を、個別ロジック IC に相当するナノ秒単位とすることができます。



必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

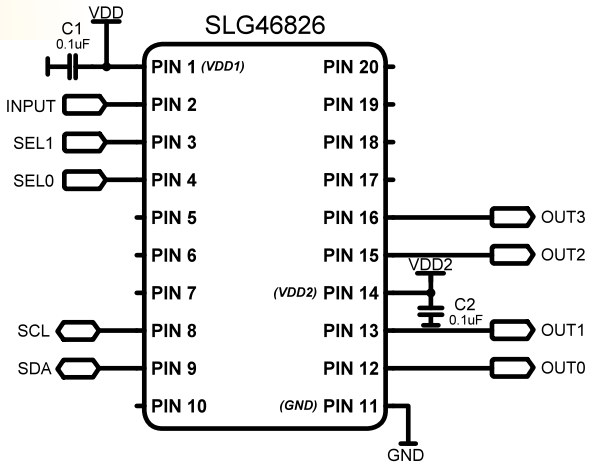
GreenPAK の図



設計手順

1. 技法：LUT マクロセルを使用した標準ロジックの構成に従って、マルチプレクサとして構成された 4 つの LUT に入力 ピンを接続します。INx を A または B に接続し、SEL0 を 4 つのすべての MUX の S に接続する必要があります。
2. 第 2 および第 3 ステージのカスケードマルチプレクサブロックを追加し、上位の SEL ビットを増やします。
3. 最終ステージのマルチプレクサに接続された出力ピンを追加します。

アプリケーション: デマルチプレクサ

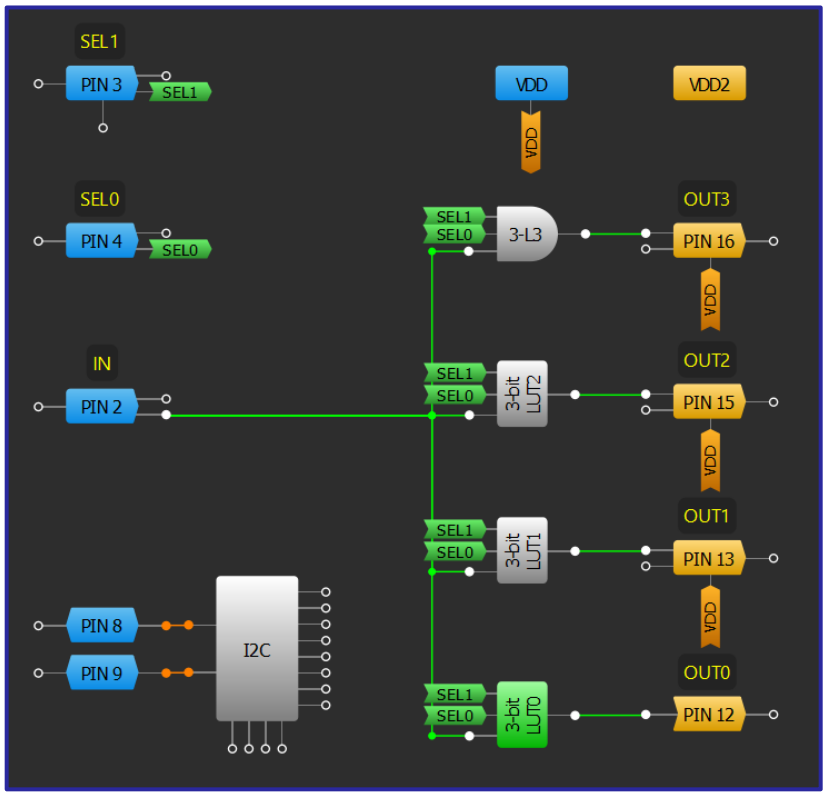


デマルチプレクサは、複数のチャンネルのうち、どのチャンネルに入力信号を送信するかを選択するために使用されます。デマルチプレクサは、1 つの回線で複数の異なるタイプのデータを送信する必要があるアプリケーションで使用され、通信システムで一般的に見られます。

必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図



設計手順

1. 信号入力(IN)、2 つの選択ライン(SELx)、および 4 つの出力ピン(OUTx)の入力ピンを設定します。
2. 選択ラインの特定のロジック入力時に IN からの信号をそれぞれ渡すように LUT を設定します。たとえば、SEL0、SEL1、および IN が HIGH の場合、3-L3 は HIGH になります。

第 2 章: シーケンシャル ロジック

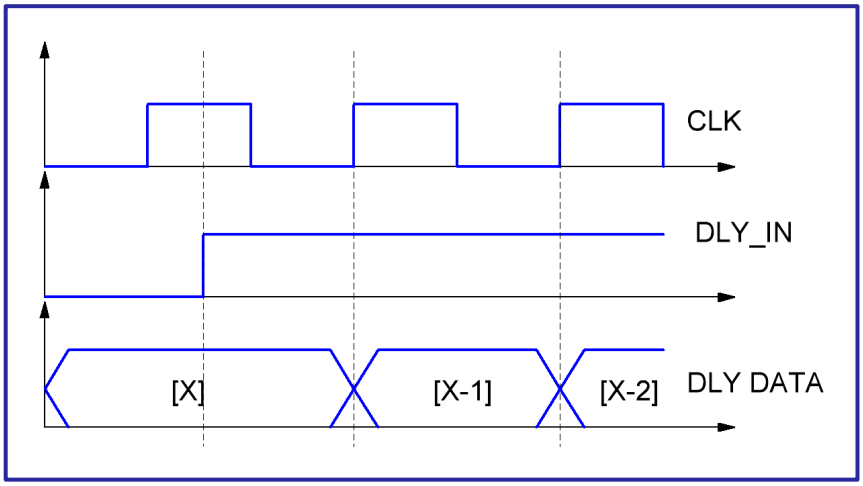
この章では、シーケンシャルロジックを含むアプリケーションについて説明します。
シーケンシャルロジックアプリケーションとしては、カウンタ、システムリセット回路、パワーシーケンサ、ステートマシンなどです。

技法:CNT/DLY 精度の最適化

この手法は、どの GreenPAK でも機能します。発振器と CNT/DLY ブロックの精度は部品によって異なります。

GreenPAK IC は、発振器を内蔵したすべてのチップと同様に、タイミングに固有のばらつきがあります。これは、製造、温度、GreenPAK の場合はユーザー設計のやり方などの要因に起因します。シンプルな設計原則を使用することで、GreenPAK 設計内のカウンタと遅延の精度を向上させることができます。

発振器と CNT/DLY ブロックの関係を考慮する必要があります。オシレーターはグローバルオシレーターです。これらは任意の数の CNT/DLY ブロックに使用され、最初は遅延またはカウンタの開始/停止に同期されません。その結果、カウンタまたは遅延がイネーブルになると、次のクロック エッジでのみインクリメントが開始されます。これは下の図に示されており、遅延のイネーブル信号はクロックサイクルの途中でアクティブになり、次の立ち上がりエッジまでデクリメントを開始しません。



Rising エッジ選択時の DLY ブロックの動作

これは、CNT/DLY ブロックの一般的な遅延時間の計算に考慮されます。

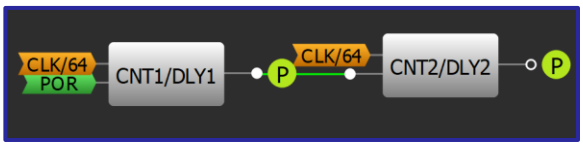
$$Delay_{time}(typical) = \frac{(Counter_Data + 1) + t}{clock}, where\ t\ is\ between\ 0\ and\ 1$$

したがって、「Counter_Data」の値が大きくなるにつれて、遅延時間に対する「t」の影響は比例して小さくなります。さらに、より速い「クロック」値が使用されている場合、より大きな「Counter_Data」値を使用しても、遅延時間の絶対値は同じに保たれます。選択した CNT/DLY ブロックの **[プロパティ]** ウィンドウで、**カウンタ データ**値と**クロック ソース**の両方を変更できます。

さらに、それぞれの GreenPAK のデータシート内のタイミング特性を参照して、電源オン時間、周波数セトリング時間、温度全体の偏差のパーセントなどの要因を考慮する必要があります。

技法：CNT/DLY ブロックのシーケンス処理

この技法は任意の GreenPAK で有効です。

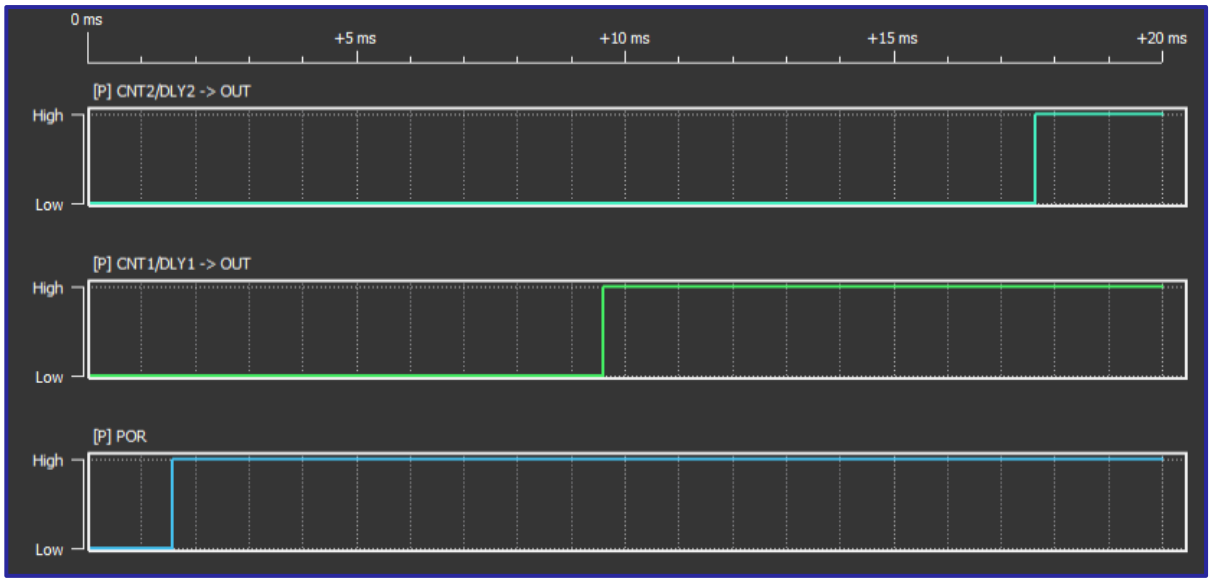


立ち上がりエッジの順次遅延

シーケンス信号を生成するために DLY ブロックを連結することが できます。1 つの DLY ブロックの出力を別の DLY ブロックの入力 に連結することにより、順次遅延のセットを作成することができます。

順次遅延セットの DLY ブロックは[Properties]ウィンドウ内の[Mode]設定で Delay に設定する必要があります。通常、[Edge select]設定はすべてのシーケンス処理コンポーネントで同じにする必要があります。図 11 に、Power-On-Reset（POR）信号 の 8ms の遅延の立ち上がりエッジに設定された 2 つの順次 CNT/DLY ブロックの影響を示します。

CNT ブロックを長いカウント時間に対して結合することもできます。CNT ブロックの結合時は、CNT の CLK は前のカウンタの



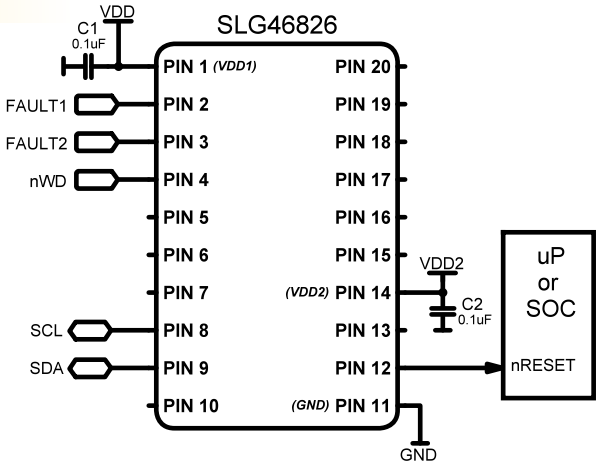
順次遅延のシミュレーション

出力によって駆動する必要があります。そのためには、プロパティで CNT ブロックを選択し、[Properties]ウィンドウで前の CNTx/DLYx から供給される Clock 接続を選択します。

順次カウンタ

アプリケーション：システムリセット

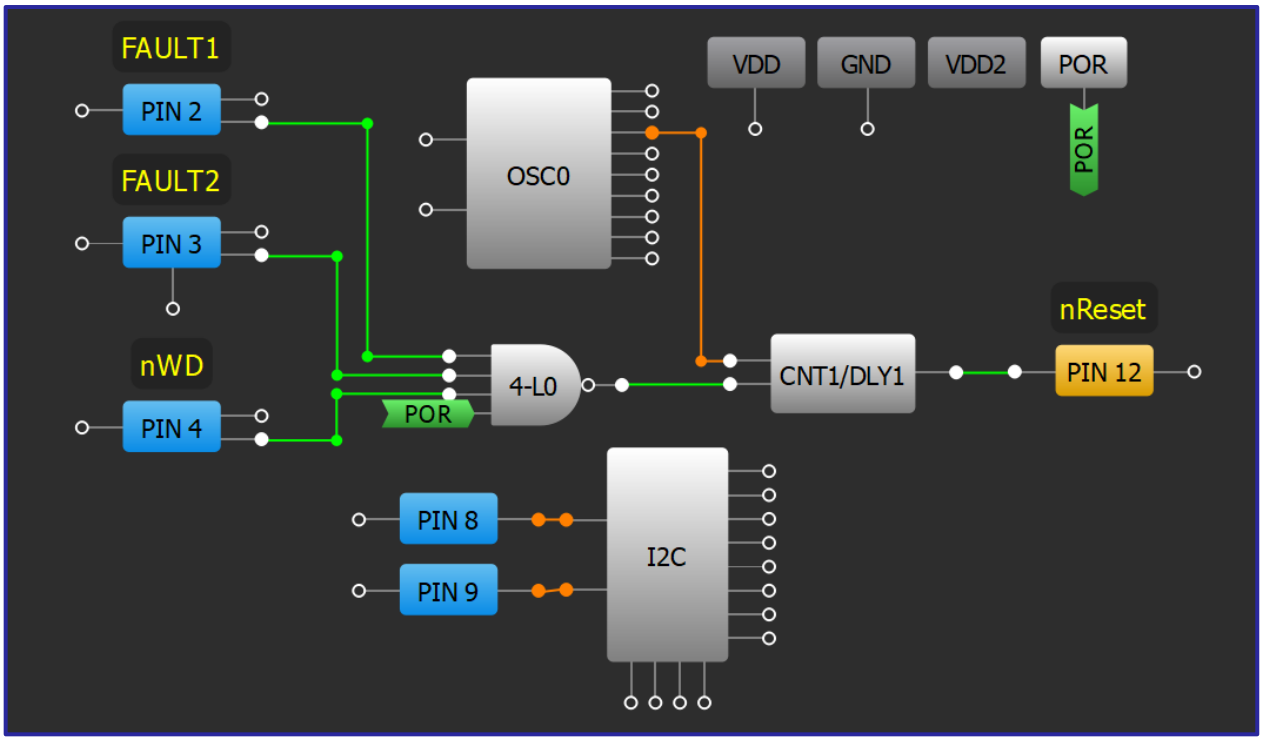
システムリセット IC は障害発生、手動リセット、一時的な供給電圧の低下などの際に、マイクロプロセッサをリセットするために使用されます。



必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

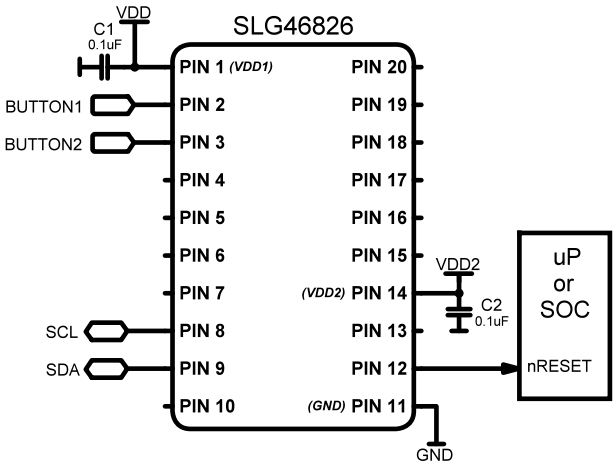


設計手順

1. I/O を各入力信号用の入力として構成します。
2. いずれかの回線が有効な場合に HI 信号を生成するための LUT ロジックを追加します。ロジックは各信号が active-hi か active-low かによって決まります。
3. CNT/DLY ブロックを One shot モードに設定し、Edge select を Rising に設定します。適切なパルス長が生成されるよ うに、Counter data を設定します。active-low パルスの場合、Output polarity を Inverted (nOUT)に変更します。
4. CNT/DLY ブロックの出力を出力ピンに接続します。

アプリケーション：複数ボタンのリセット

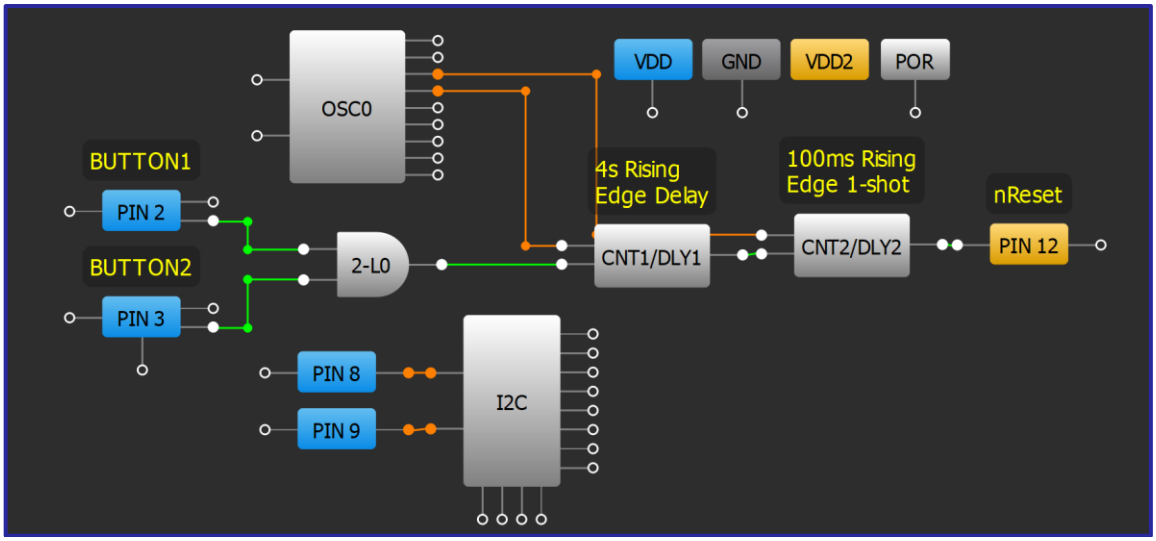
ハードリセットを実行するために複数のボタンを押したままにする操作は、多くのデバイスで共通のインターフェースです。個々の IC にこのアプリケーションを実装すると、システムの残りの部分で 1 つ以上のソフトウェア、ファームウェア、ハードウェアの問題が発生している場合でもリセットが認識され、作用します。



必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

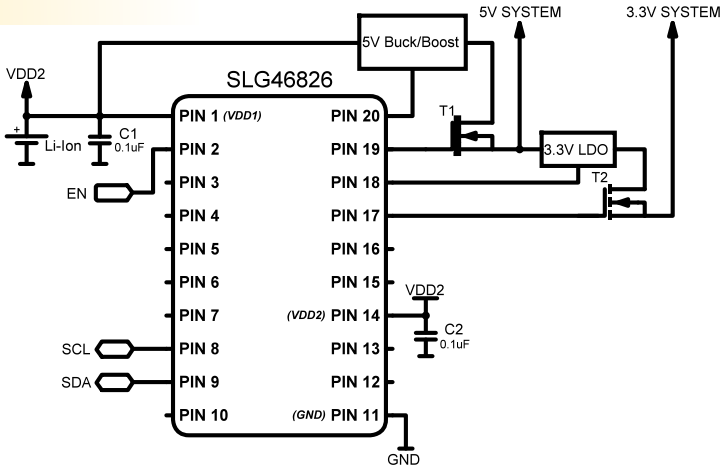


設計手順

1. I/O を各ボタン用の入力として構成します。
2. 両方のボタンが有効な場合に HI 信号を生成するための LUT ロジックを追加します。ロジックは各信号が active-hi か active-low かによって決まります。
3. CNT/DLY ブロックを Delay モードに設定し、Edge select を Rising に設定します。適切なボタン保持時間が生成されるように、Counter data を設定します。active-low パルスの場合、Output polarity を Non-inverted (OUT)に変更します。
4. 2 番目の CNT/DLY ブロックを One shot モードに設定し、Edge select を Rising に設定します。適切なパルス長が生成されるように、Counter data を設定します。active-low パルスの場合、Output polarity を Inverted (nOUT)に変更します。
5. CNT/DLY ブロックの出力を出力ピンに接続します。

アプリケーション：基本シーケンサ

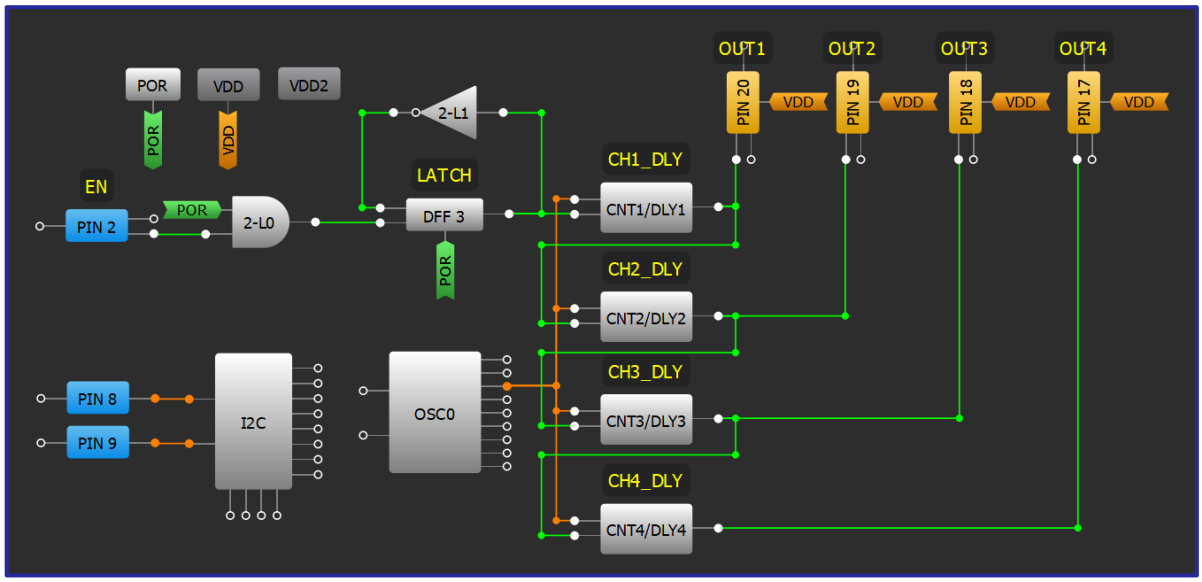
シーケンサは、設計者がシステムのさまざまな部分を順次有 効化する必要がある場合に使用されます。この機能は特に、 複数のパワーレールを必要とするアプリケーションで重要となることがあります。



必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

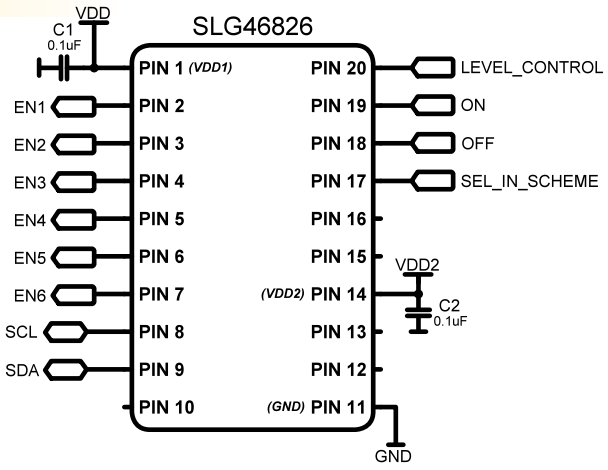


設計手順

1. LUT を使用して、適切な起動条件を設定します。
2. ラッチまたは DFF を使用して起動信号を維持し、DLY ブロックに継続して入力します。
3. 技法：CNT/DLY ブロックのシーケンス処理に従って遅延を連結します。
4. 各遅延チャネルを適切な出力ピンに接続します。

アプリケーション: カスケードシーケンサー

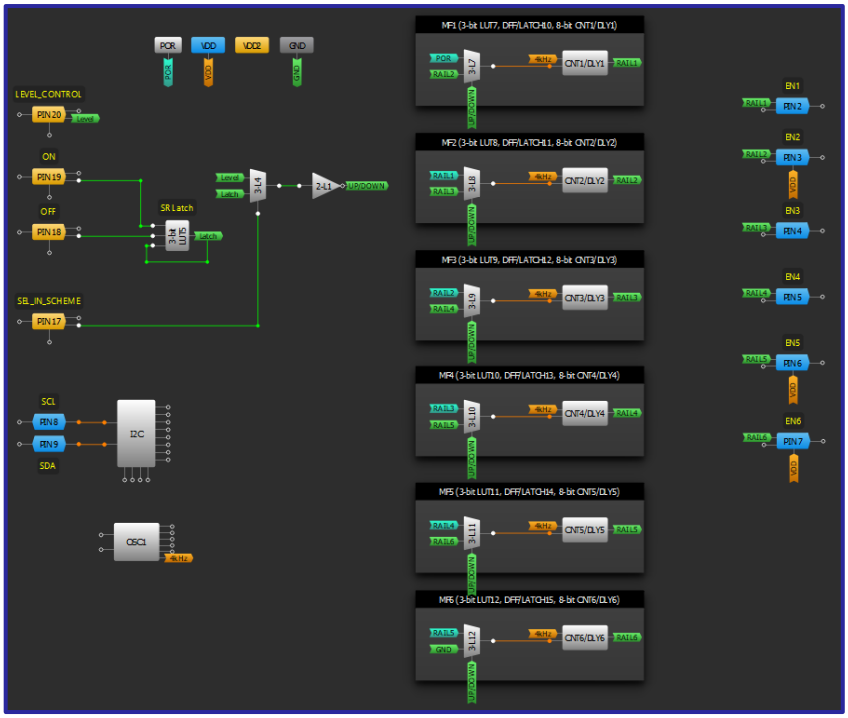
シーケンサーは、設計者がシステムのさまざまな部分を順番にアクティブ化する必要がある場合に使用されます。 通常、カスケード シーケンスを使用して、ある電圧レールの下にあるすべてのレールがオフになるまで、そのレールがオフにならないようにします。



必要コンポーネント

- 任意の GreenPAK
- 他のコンポーネントは必要ありません

GreenPAK の図

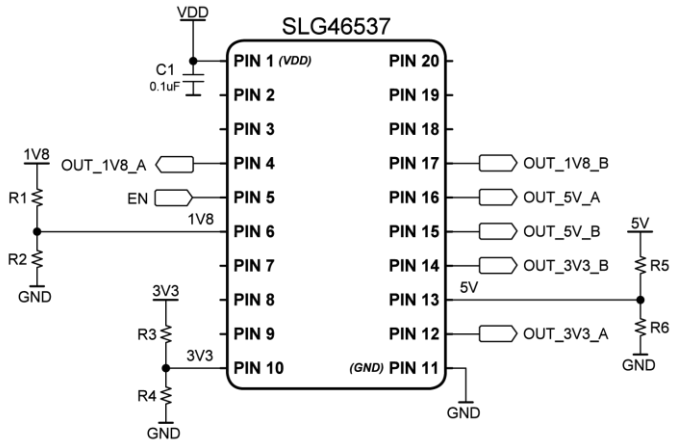


設計手順

1. 入力構造の構成です。 ここでは、PIN17 でレベル制御とラッチ制御を選択できます。
2. カスケード効果を実現するために、多機能ブロックの DLY へマルチプレクサ経由で入力します。
3. DLY の出力を push pull の出力ピンに接続します。

アプリケーション:電圧監視パワーシーケンサ

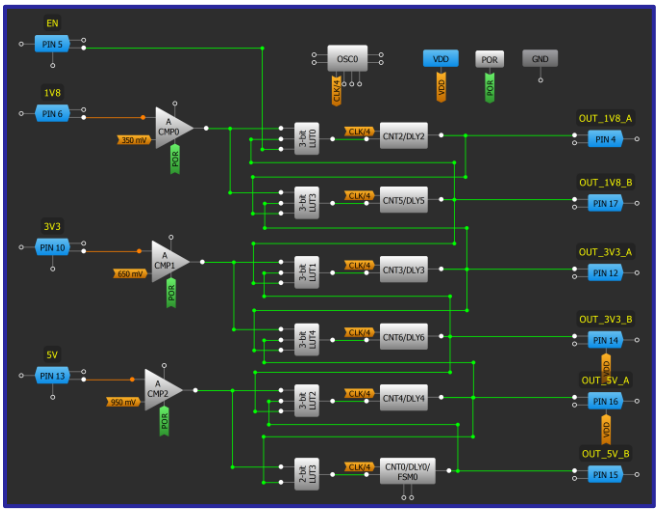
シーケンサーは、設計者がシステムのさまざまな部分を順次アクティブ化する必要がある場合に使用されます。電源レールとフォルト状態を監視することにより、回路設計者はシステムの必要なシーケンスを自己調整することができます。

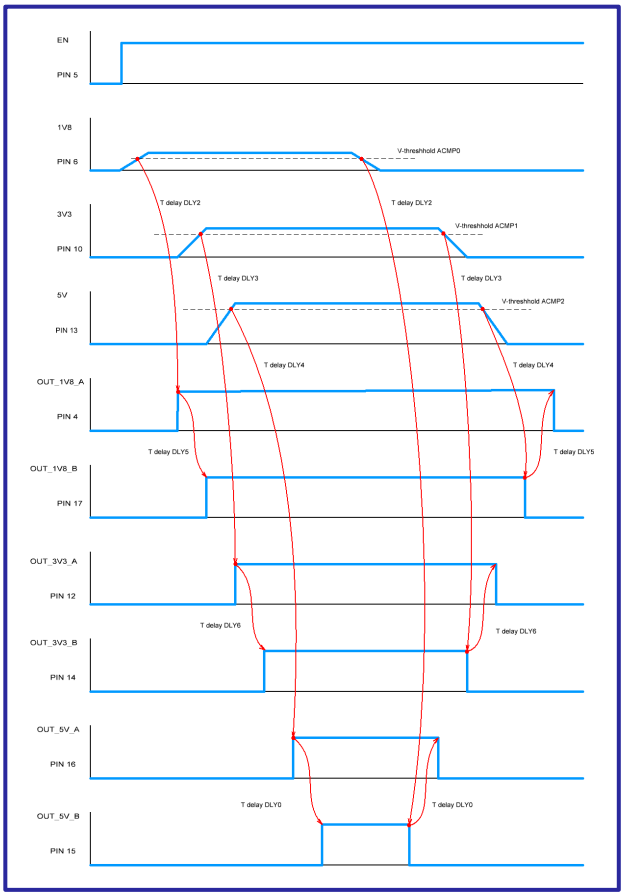


必要なコンポーネント

- 3 つの ACMP を持つ任意の GreenPAK
- 6 つの抵抗

GreenPAK の図



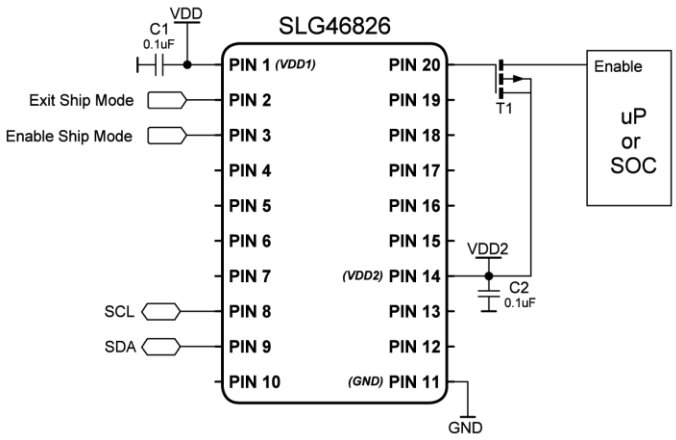


設計手順

1. EN および電圧監視用の入力ピンを設定します。
2. 出力ピンをシーケンスシステムに設定します。
3. ACMP の電源を入れ、POR を PWR UP に接続し、それぞれの IN ソースを所望の電圧スレッショルドレベルに設定します。
4. 必要な遅延時間で DLY を構成します。
5. 適切なロジック機能を使用して LUT を構成します。

アプリケーション：出荷モードコントローラ

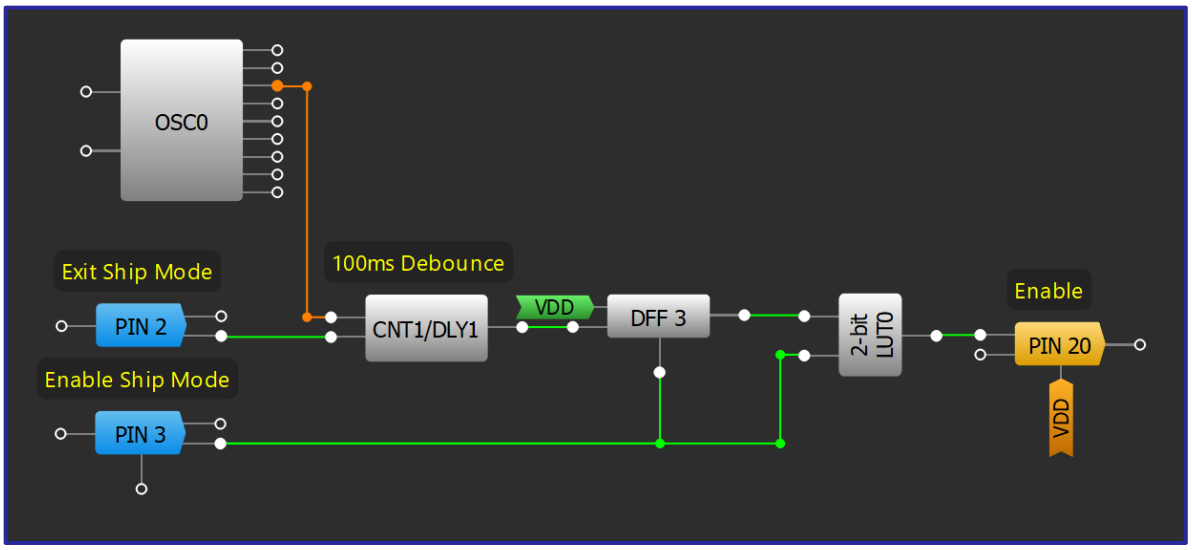
超低電力ボタンモニターを使用して、製品がエンドユーザー の元に届いていない状態で、バッテリー寿命を節約すること ができます。これにより、ユーザーは適切な状態で使用を開始することができます。



必要なコンポーネント

- 任意の GreenPAK
- 外部 PMOS ロードスイッチ

GreenPAK の図

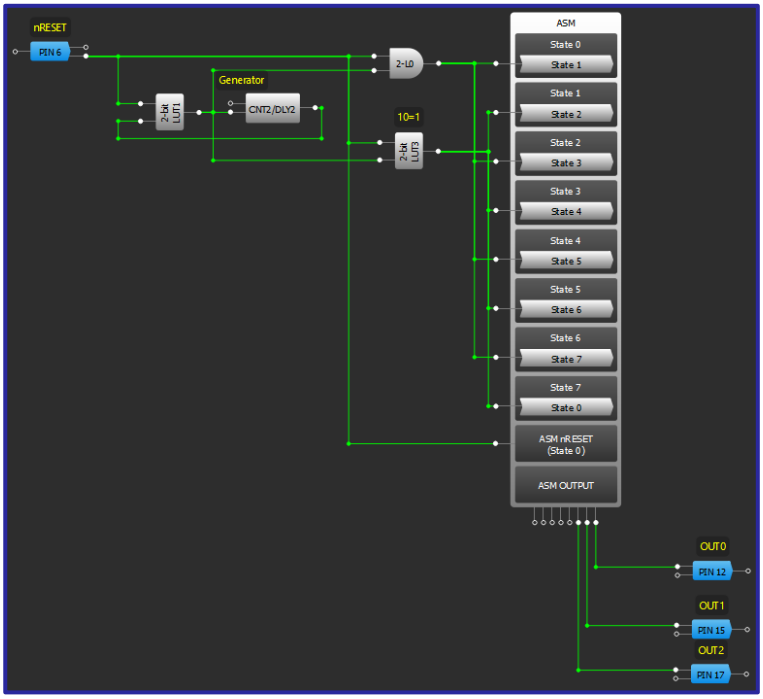


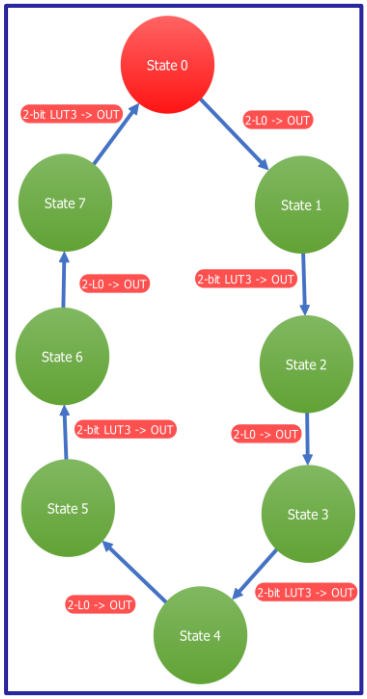
設計手順

1. ピン 2 を 1MΩ のプルアップ付きの入力として構成します。
2. CNT1/DLY1 で適切なボタン遅延時間を設定します。
3. 出荷モードの内外の正しい極性になるように LUT の内容を修正します。

技法: ASM からの同期ステートマシンの作成

同期ステートマシン(SSM)は、遷移条件が満たされると、着信クロックのエッジで遷移します。GreenPAK 非同期ステート マシン (ASM) マクロセルを SSM に変換する一般的なアプローチでは、ASM の遷移時間よりも大きいパルス幅のクロック信号を使用します。





上記の 3 ビット カウンタの例の SSM を考えてみましょう。CNT2 と 2 ビットの LUT1 を使用してクロックを生成します。ASM は、直列に接続された 8 つの状態を使用します。2 ビットの LUT0 と 2 ビットの LUT3 を使用して、近傍の 2 つの状態で論理 High 信号による遷移を防止します。各状態の ASM 出力の値を以下に示します。

RAM

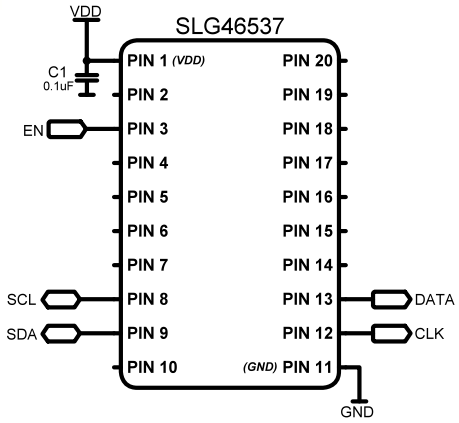
State name	Connection Matrix Output RAM							
	OUT7	OUT6	OUT5	OUT4	OUT3	OUT2	OUT1	OUT0
State 0	0	0	0	0	0	0	0	0
State 1	0	0	0	0	0	0	0	1
State 2	0	0	0	0	0	0	1	0
State 3	0	0	0	0	0	0	1	1
State 4	0	0	0	0	0	1	0	0
State 5	0	0	0	0	0	1	0	1
State 6	0	0	0	0	0	1	1	0
State 7	0	0	0	0	0	1	1	1

PIN6 が High になると、ASM はリセット状態 (状態 0) から次の状態 (状態 1) に変化します。次の状態遷移は、CNT2 が最初に High、次に Low、というようにトグルするときに発生します。

ASM を使用して SSM を作成するプロセスの詳細については、「AN-1126 ASM から同期変換」を参照してください。

アプリケーション: N ビット長 ビットストリーム

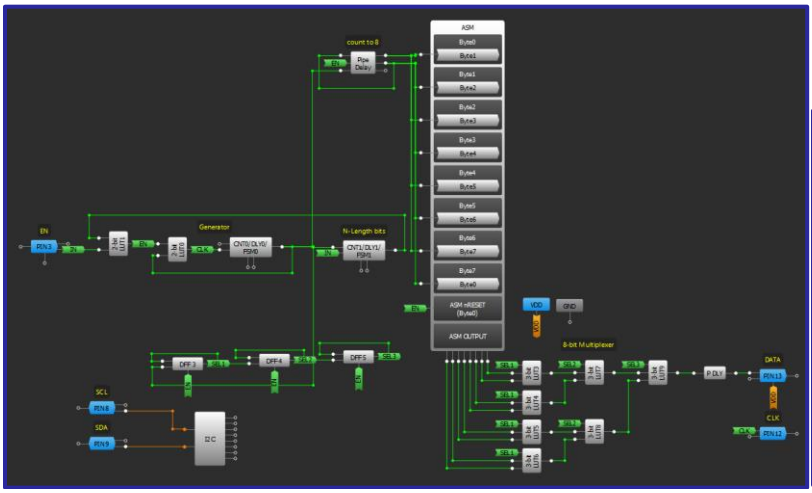
ビットストリームは、通信パスを介して連続的に送信されるビットのシーケンスです。GreenPAK は、最大 64 ビットの繰り返し文字列を作成できます。

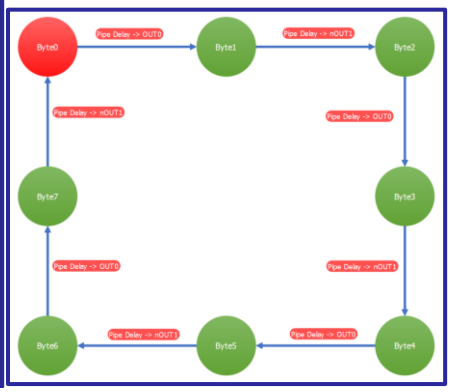


必要なコンポーネント

- ASM をもつ任意の GreenPAK

GreenPAK の図





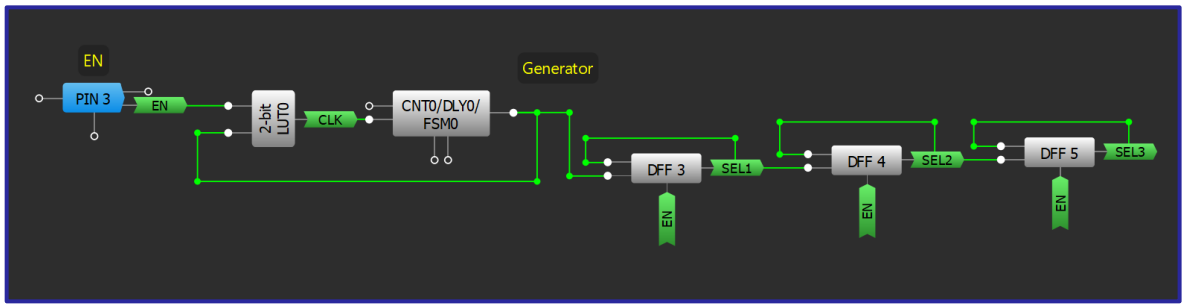
設計手順

1. 「アプリケーション:8 ビットマルチプレクサ」を使用して、ジェネレータと 8 ビットマルチプレクサを構成します。
2. **CNT1** を設定することにより、ビットストリームの長さを決定します。
3. 「技法: ASM からの同期ステートマシンの作成を使用して ASM を構成します。
4. 8 ビットマルチプレクサとジェネレータの出力を目的の出力 PIN に接続します。
5. ビットストリームの長さ(**CNT1** のカウンタデータ)は、I2C を使用して変更できます。
6. ASM 出力 RAM に格納されたデータは、I2C を使用して変更できます。

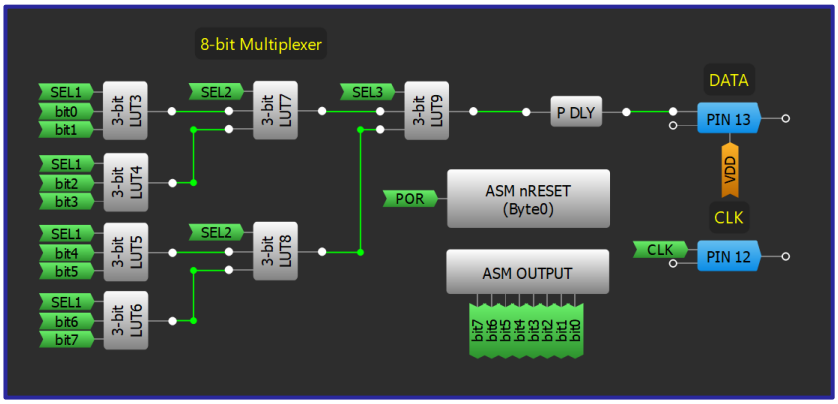
技法: ビットストリームの多重化

この技法は、任意の GreenPAK で使用できます。

GreenPAK は、データパターンを転送するためによく使用されます。 データが GreenPAK から転送される場合、またはデータが SoC から複数のラインに沿って送信される場合、それらは 1 つのラインで伝送するためにマージされる必要があります。以下は、ASM 出力 RAM から発信されるビットストリームを多重化する GreenPAK の例です。



ジェネレータ回路は上に示されています。 ジェネレータは、同期データ送信用の CLK ラインの動作を実装し、EN 信号によって有効になります。ジェネレータは、1 つのライン上で送信されたデータを正しく組合わせるためのマルチプレクサ操作アルゴリズムも実装します。

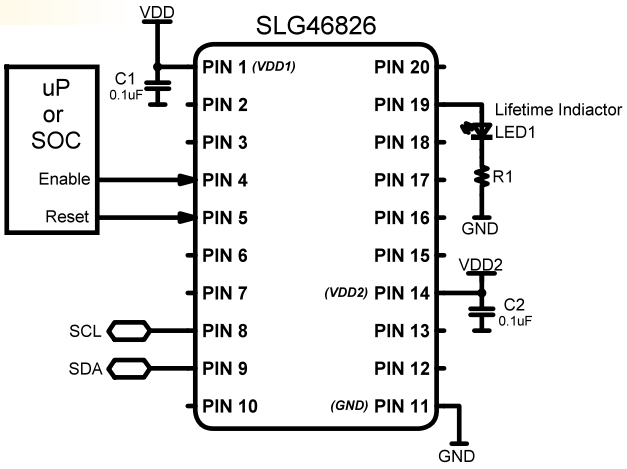


IN3	IN2	IN1	IN0	OUT
0	0	0	0	0
0	0	0	1	1
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

8 ビットマルチプレクサ回路を上に示します。すべての LUT は 8 ビットマルチプレクサとしてまとめて設定され(AN-1003 を参照)、MUX の真理値表も上に示されています。マルチプレクサは、ジェネレータのアルゴリズムに従って ASM ブロック出力からのビットの組み合わせを出力し、データは 1 つのラインで DATA に送信されます。EN が Low の場合、DATA 出力は常に ASM 出力 RAM の MSB と同じ になります。ASM 出力 RAM は I2C 経由で変更できます。 データビットを変更するために ASM の状態を変更するロジックを実装することもできます。ASM が使用できない場合は、データ値に対して入力を High または Low に設定できま。

アプリケーション：10 年カウンタ

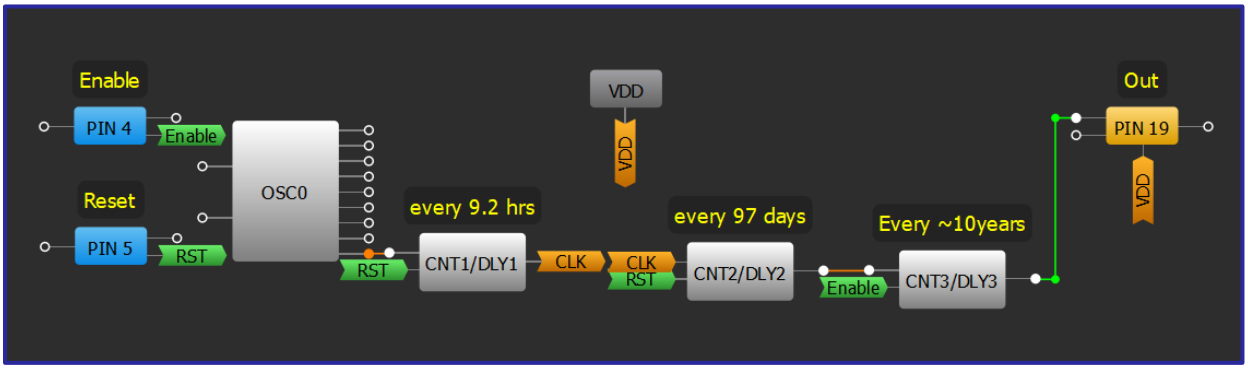
パワーバジェットに対する大きな負担なしで製品の寿 命を測定するために、非常に長いカウンタを使用する ことができます。



必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

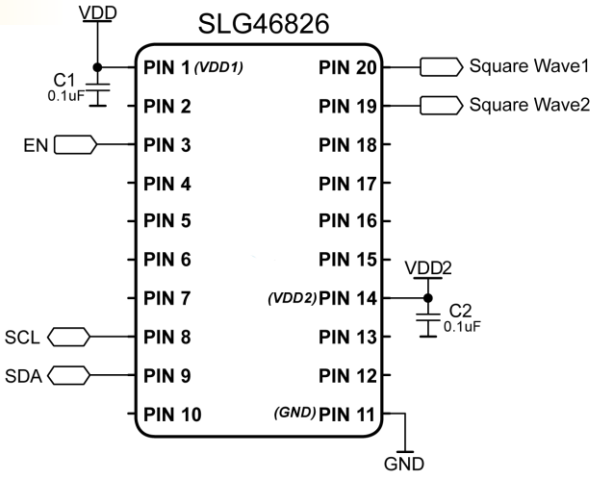
GreenPAK の図



設計手順

1. 技法：CNT/DLY ブロックのシーケンス処理に従ってカウンタを連結します。
2. 入力ピンと出力ピンを接続します。
3. CNT プロパティでタイミングを設定します。

アプリケーション: 方形波発生器

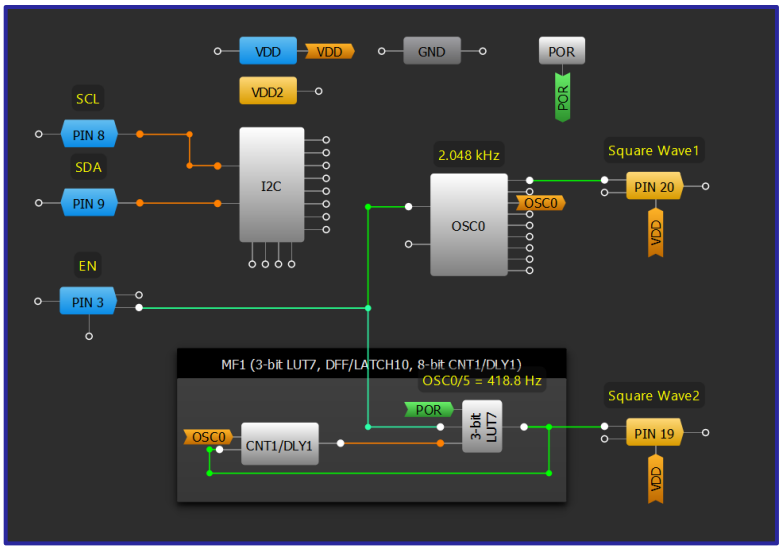


矩形波は、デジタルシステムのクロッキングに不可欠です。これらは、発振器ブロックまたはカスタマイズされた周波数の遅延ロジックを使用して GreenPAK に簡単に実装できます。

必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図



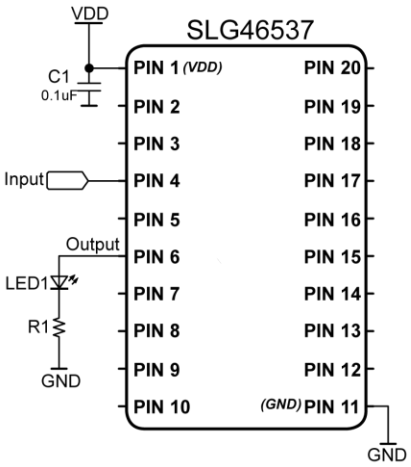
設計手順

1. **EN** 入力と方形波出力を構成します。
2. 内部発振器を使用して **PIN20** に方形波を生成します。「**CLK**」**前置分周器**と「**OUT0**」**第 2 分周器** を変更して、周波数をカスタマイズすることができます。
3. Both エッジ遅延と LUT の両方を使用して、**PIN19** で方形波を生成します。この構成により、ユーザーはより細かい調整によって方形波の周波数を分周することができます。

$$分周係数 = カウンタデータ + 2$$

アプリケーション: 2 つのイベントボタンを押す

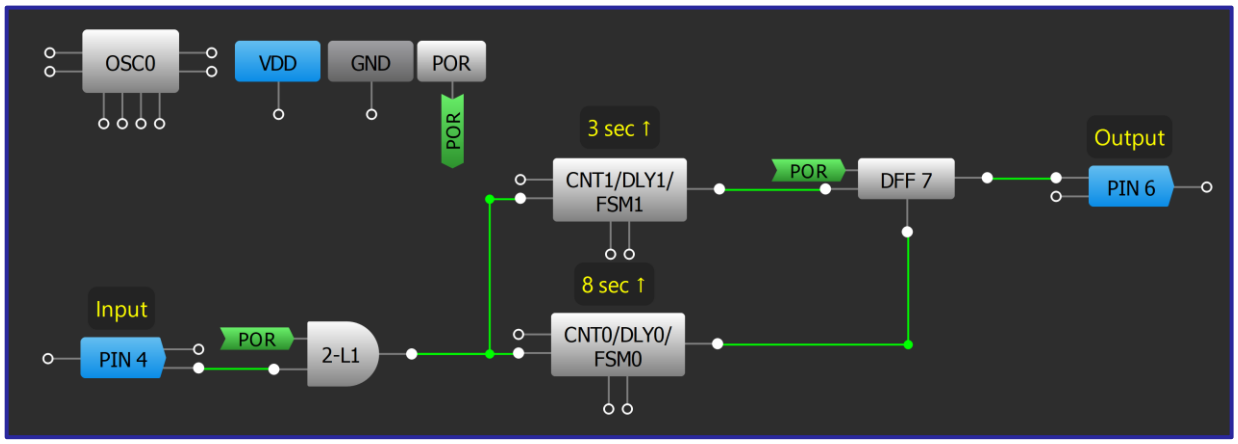
1 つのボタンを使用して複数のイベントを生成することは、外部の制御用コンポーネントを節約するための一般的なソリューションです。 1 つのボタンと所定の時間間隔を使用して、LED の点灯の制御することができます。



必要なコンポーネント

- 任意の GreenPAK
- 1 つの LED
- 1 つの抵抗

GreenPAK の図



設計手順

1. GPIO ピンをボタンの入力および LED 制御の出力として構成します。
2. **CNT0/DLY0**、**CNT1/DLY1**、および最後の状態を記憶するために DFF を追加します。
3. CNT/DLY ブロックを「DLY」モードに設定し、**エッジ選択**を「Rising」に設定します。
4. **CNT0/DLY0** 出力を 「nOUT」として構成します。

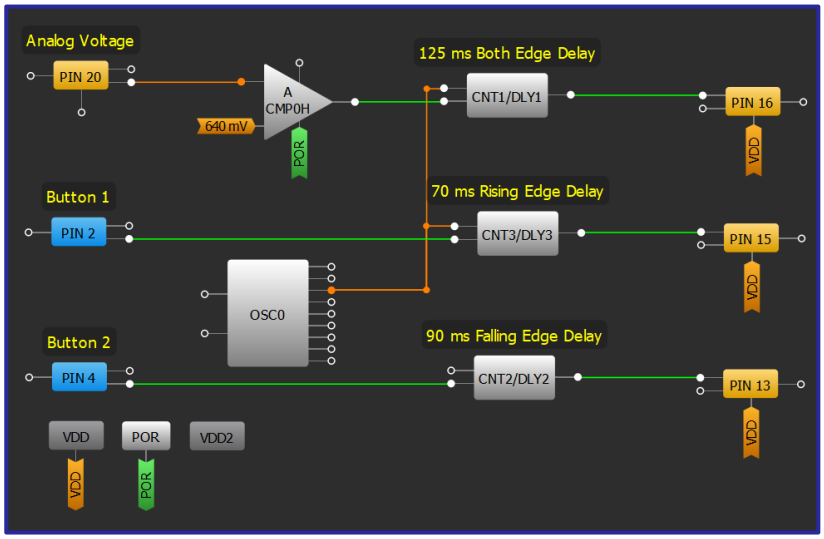
第 3 章：信号の調整

この章では、外部信号を調整してシステム内での動作のために有効にするアプリケーションが紹介されています。これに関わるアプリケーションとしては、分周、倍周、フィルターそしてセンサーコントローラなどが挙げられます。

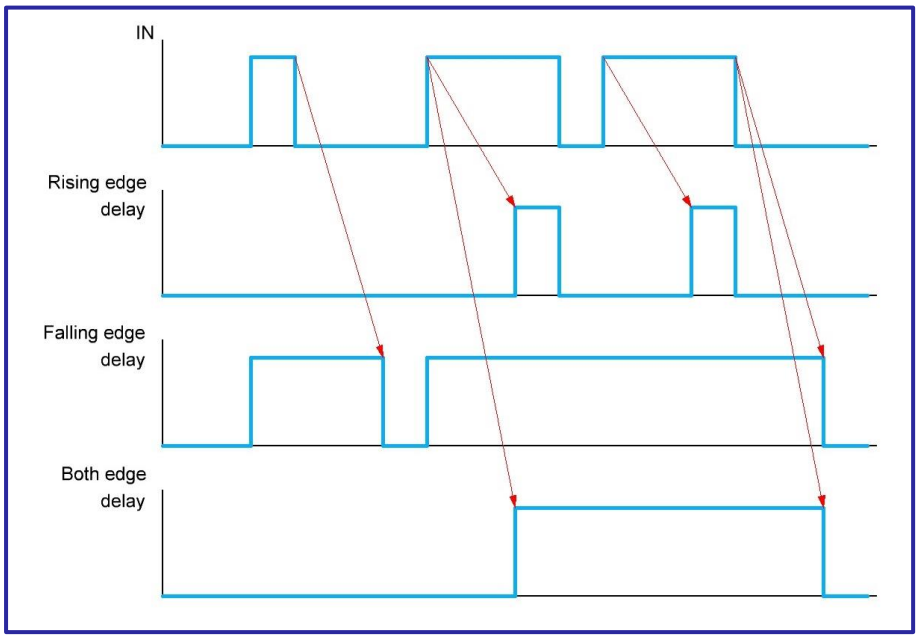
技法：CNT/DLY ブロックをデグリッチフィルターとして使用

GCNT/DLY ブロックは各 GreenPAK に内蔵されているため、この技法はどの GreenPAK でも使用できます。.

グリッチ/スプリアス信号を除去する必要がある場合、デグリッチ/デバウンスフィルターの使用が一般的です。グリッ チはさまざまな状況で発生する可能 性があります。たとえば、ボタンを押 している/放しているときや、電圧レベ ルが（入力 PIN またはアナログコン パレータの）しきい値に非常に近くて ヒステリシスまたはシュミットトリガー がない場合などです。 デグリッチ遅延を設定するための 3 つのエッジトリガーオプションとして、 立ち上がり、立ち下がり、両方があり ます。この場合、遅延ブロックは対応 する極性を持つ遅延値よりも短いパ ルスをフィルタリングします。立ち上がりエッジの場合はアクティブハイ、立ち下がりエッジの場合はアクティブロー、両方のエッ ジ遅延の場合はハイおよびローです。下図を参照してください。



デグリッチ遅延オプションの例

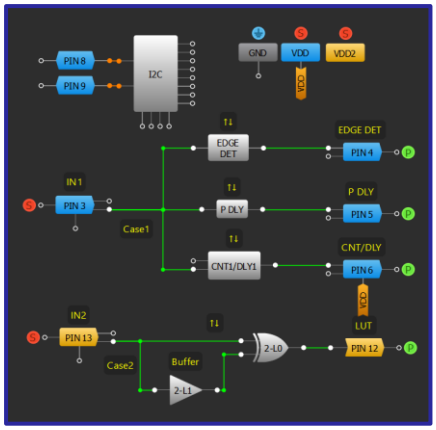


エッジ遅延の動作

技法: エッジ検出器

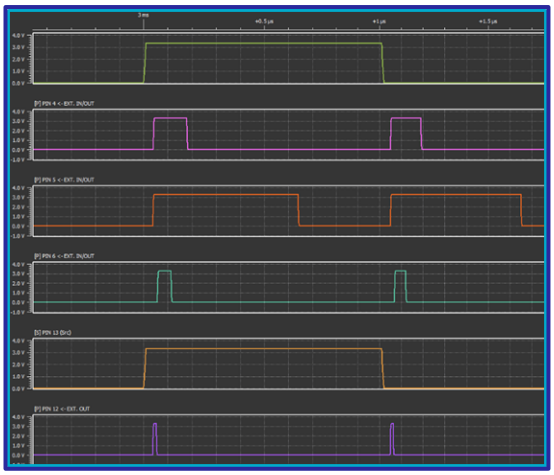
この技法は、任意の GreenPAK で使用できます。

エッジ検出器は、デジタル電子機器の重要なコンポーネントです。それは 1 つの入力と 1 つの出力を持つシンプルな回路です。 エッジ検出器は、定義されたエッジ(立ち上がり、立ち下がり、またはその両方)が検出されると、短いパルスを生成します。 リセット機能、ウォッチドッグタイマー、またはその他のエッジ依存アプリケーションを実装する場合に便利です。エッジ検出器を実装するには、いくつかの方法があります(下図を参照)。 エッジ検出器の構築方法の詳細な説明とその他の例については、「AN-1046 さまざまなエッジ検出器回路」を参照してください。



エッジ検出器の実装

オプション 1 は、EDGE DET、P DLY、および CNT/DLY ブロックに組み込まれているエッジ検出器機能を使用します。 これらのブロックには、オプション 2 と比較してより長い持続パルスを生成するという利点があります(下図)。

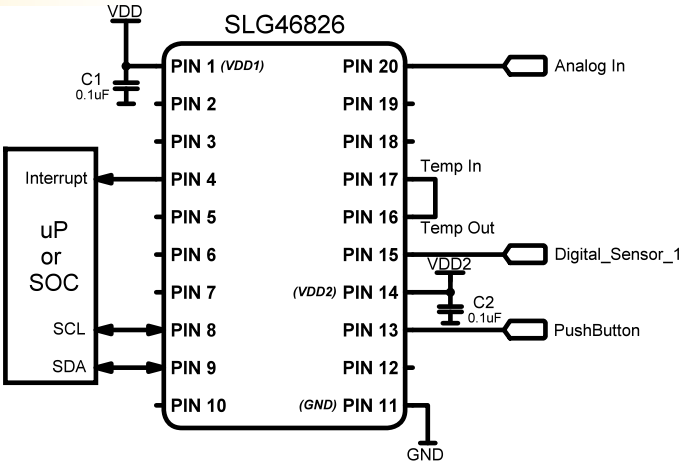


異なるエッジ検出器のタイミング

オプション 2 では、バッファを介した信号の伝搬によって生じる小さな遅延を使用します。この遅延信号は、2 ビット XOR によって、伝搬時間が非常に短い元の入力と比較されます。バッファを通る短い遅延は、XOR の入力間に差を生じさせ、XOR の出力に短いパルスを生成します。 LUT の内部構造のため、それらの入力の伝搬遅延時間は異なります。

アプリケーション：割り込みコントロー ラ

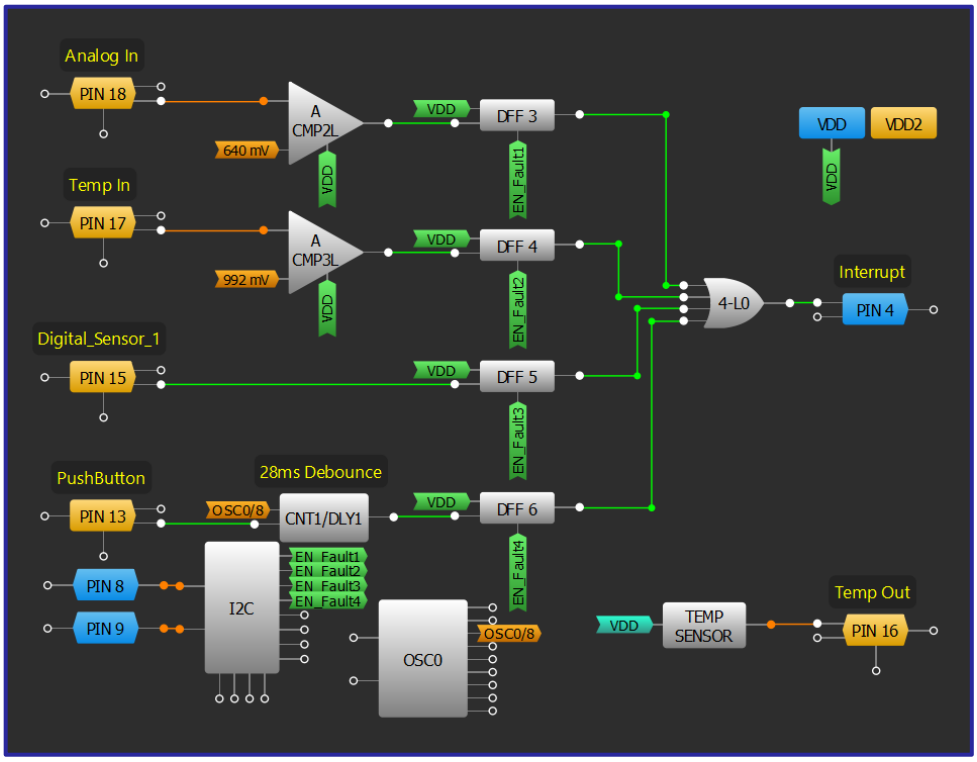
GreenPAK は複数の割り込み信号を監視し、ホストプロ セッサが動作するようにその信号を集約することができま す。μP または SOC は、障害の原因を特定するために、 I2C を通して各 DFF の出力を読み取ることができます。



必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図



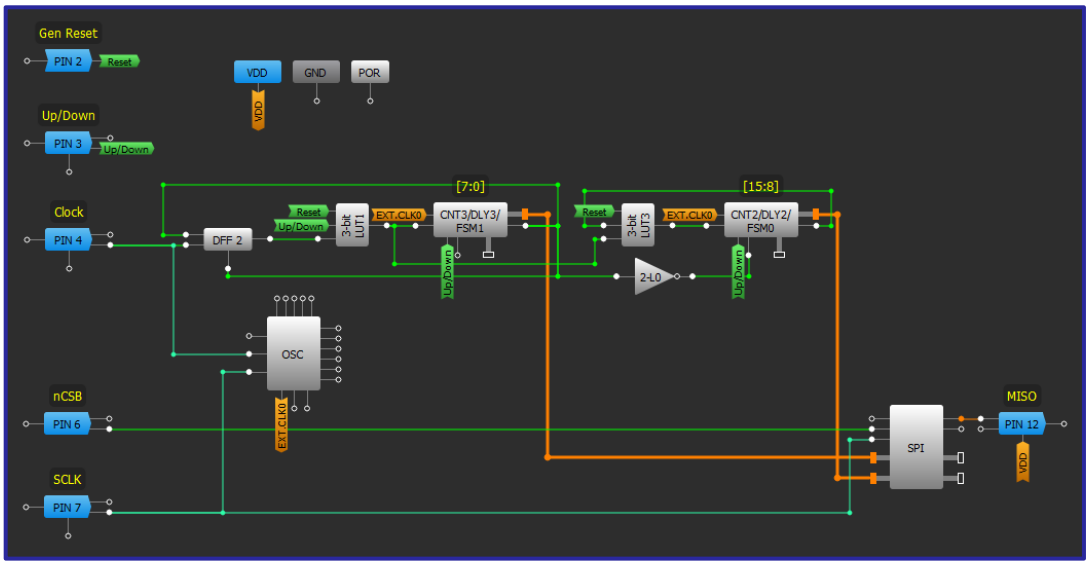
設計手順

1. ピン 16、17、18 をアナログ入力/出力として構成します。
2. ACMP のレベルを適切なしきい値に設定します。
3. DFF を OR ゲートにワイヤー接続し、ピン 4 を出力として設定します。
4. DLY1 を適切なデバウンス時間に設定します。

技法: 双方向カウンタの作成

この技法は、SPI インターフェースを含む任意の GreenPAK で使用できます。この技法の代替手段としては、FSM ブロックを持つ他の GreenPAK を使用し、I2C 読み取りコマンド、パラレル出力、またはその他の方法を使用してカウンタ情報を格納することによって実現できます。

カウンタは、入力イベント(パルス、エッジ)をカウントするために使用される基本的なデジタル回路で、多くの場合、デジタルフリップフロップのカスケードを使用して構築されます。GreenPAK では、一部の CNT/DLY ブロックはより堅牢であり、GreenPAK マトリックスの接続による制御により、インクリメントできるだけでなく現在の値をデクリメントまたは保持できる**有限ステートマシン (FSM)** として使用できます。この技法を使うことにより、例えば GreenPAK の 2 つの FSM ブロックを使用してパルス入力 (**クロック**) を監視し、SPI マクロセルを介して対応する 16 ビットシーケンスを出力することができます。



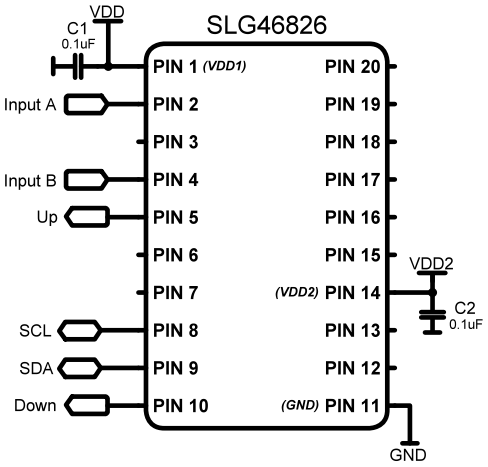
SPI 出力付き 16 ビット FSM

SPI ブロックへの出力を備えた 16 ビット FSM は、**FSM0**、**FSM1** により構築された 16 ビットレジスタで、入力クロックパルスをカウントします。ユーザーはいつでも SPI 経由で値を読み取ったり、16 ビットレジスタをリセットしたり、カウント方向を変更したりすることができます。

16 ビットカウンタは、2 つのカウンタ (**FSM0** ブロックと **FSM1** ブロック)と追加のロジックを使用して実装されます。 ビット [15:8] は FSM0 に格納され、[7:0] は **FSM1** に格納されます。2 つの FSM はともに SPI ブロックに接続されており、SPI 経由でシリアルデータを出力できます。カウント方向は、FSM ブロックの **UP** マトリクス入力に直接接続されたアップ/ダウンピンによって制御されます。 このピンが High の場合、システムはカウントアップし、このピンが Low の場合、システムはカウントダウンします。**Gen Reset** ピンは、両方のカウンタ値(アクティブ High)をリセットするために使用されます。

クロック入力ピンは、**FSM1** と **FSM0** の **CLK** 入力に同時に印加されます。 **FSM1 は**各クロックをカウントしますが、**FSM0 は FSM1** カウンタ値が 255 でアップ/ダウン信号が High の場合、または **FSM1** カウンタ値が 0 でアップ/ダウン信号が Low の場合にのみカウントします。この機能は、**FSM0** の **KEEP** 入力を使用して実現されます。この信号が High の場合、クロック信号にもかかわらず **FSM0** のカウンタ値は変化しません。**KEEP** はインバータを介して **FSM1** 出力に接続されています。 一方、**FSM1** の出力は、カウンタ値が 0 でアップ/ダウン信号が Low の場合、またはカウンタ値が 255 でアップ/ダウン信号が High の場合にのみ High になります。

アプリケーション：エンコーダ

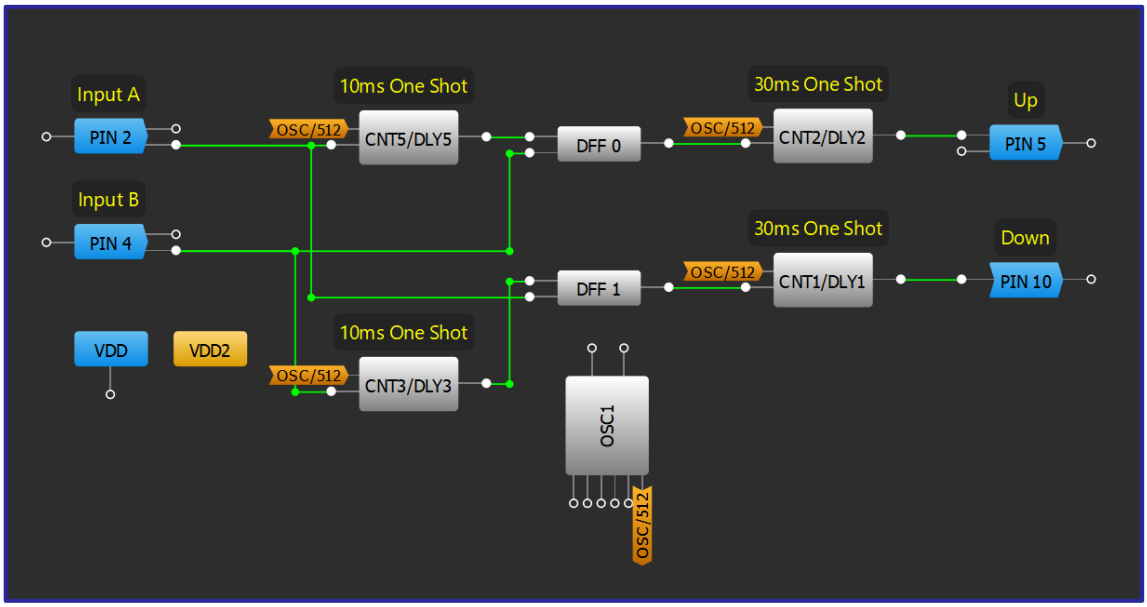


エンコーダは一般的に、回転運動または直線運動をデジタル信号に変換するため に使用されます。この設計はマウスホイールやヘッドセットのボリューム調節用に 最適化されています。

必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

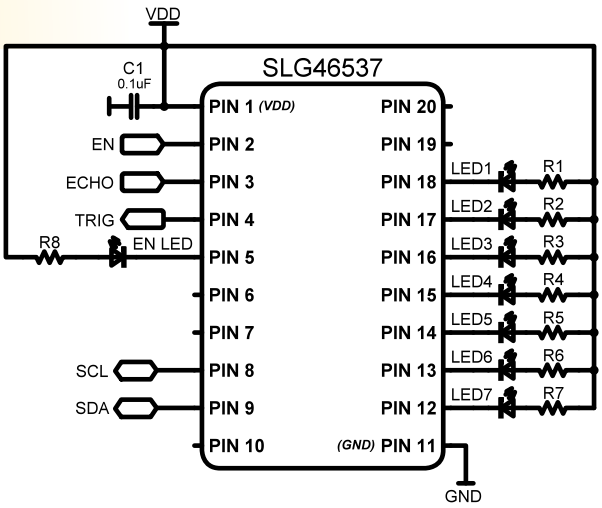
GreenPAK の図



設計手順

1. ピンをデジタル入力として構成します。
2. 適切なフィルター時間を使用して、CNT3/DLY3 および CNT5/DLY5 を One shot モードに設定します。
3. 方向（上または下）を検出するように DFF を設定します。
4. 適切な出力パルス幅を使用して、CNT1/DLY1 および CNT2/DLY2 を One shot モードに設定します。

アプリケーション: 距離の計測

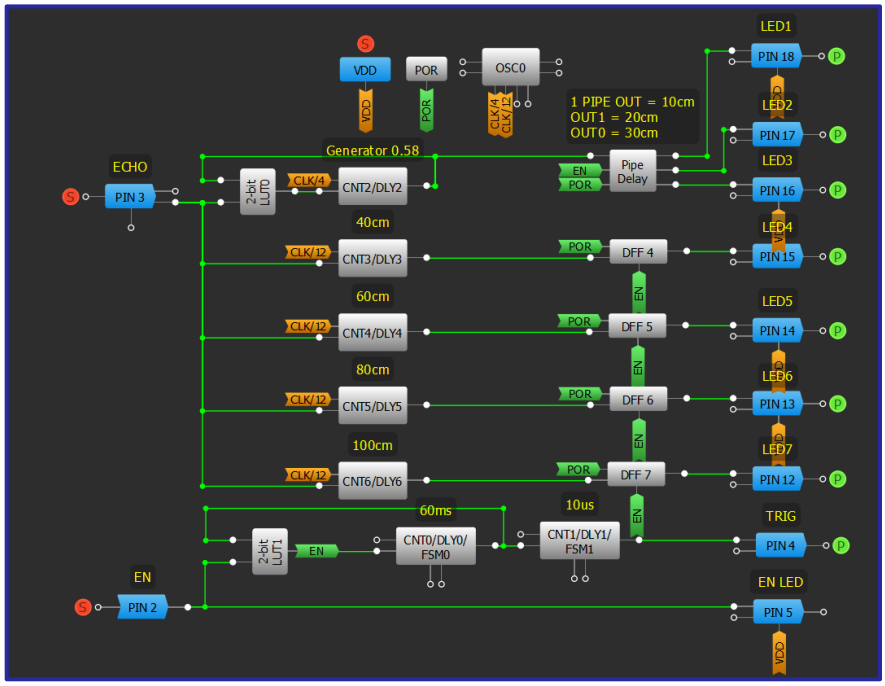


超音波測距モジュールは、非接触測定機能を提供します。この設計は、HC-SR04 に基づく超音波距離計用のコントローラです。

必要なコンポーネント

- 任意の GreenPAK
- 距離測定ごとの LED
- 距離測定ごとの抵抗

GreenPAK の図

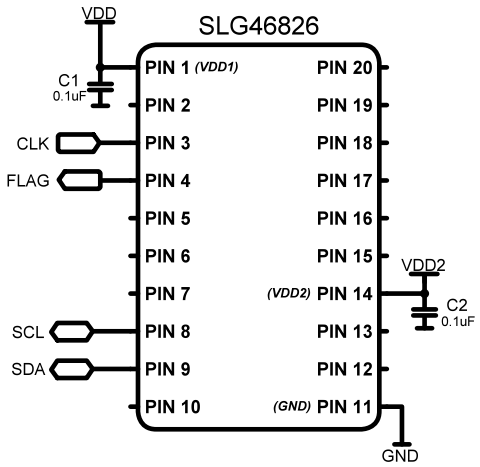


設計手順

1. GPIO 入力を **ECHO** に、出力を **Trig** に設定します。
2. LUT ロジックと **CNT/DLY0** を追加して、ENABLE 信号を備えたジェネレータを作成します。
3. パイプ遅延と **CNT/DLY2** を追加して、距離を検出するジェネレータを作成します。
4. CNT/DLY ブロックを立ち上がりエッジ遅延として設定して、さまざまな距離を測定します。
5. 距離データをラッチするための DFF を追加し設定します。
6. 各 DFF 出力を所望の出力ピンに接続し、オープンドレインとして設定します。

アプリケーション: 周波数範囲検出器

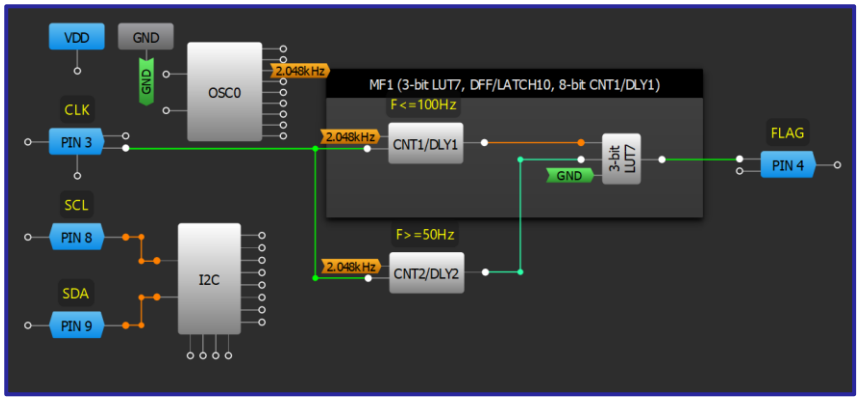
多くのデバイスには、動作する特定の周波数範囲があり、入力クロックがこの範囲内にとどまる必要があります。このアプリケーションは、入力クロック周波数が所望の範囲内にあるかどうかを検出するために使用されます。

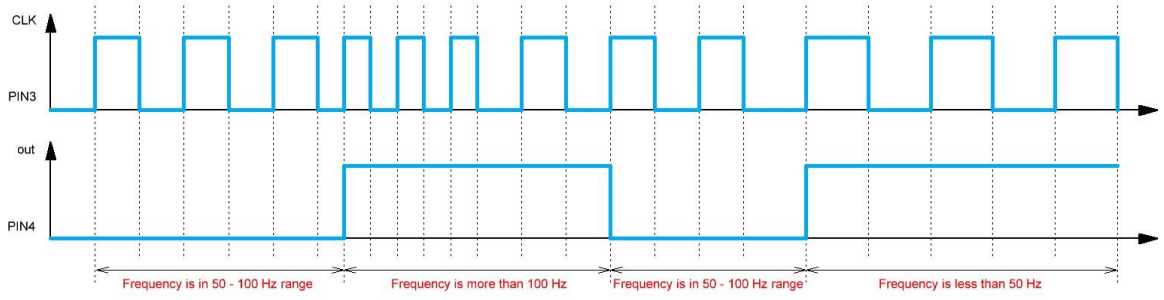


必要なコンポーネント

- 任意の GreenPAK

GreenPAK の図



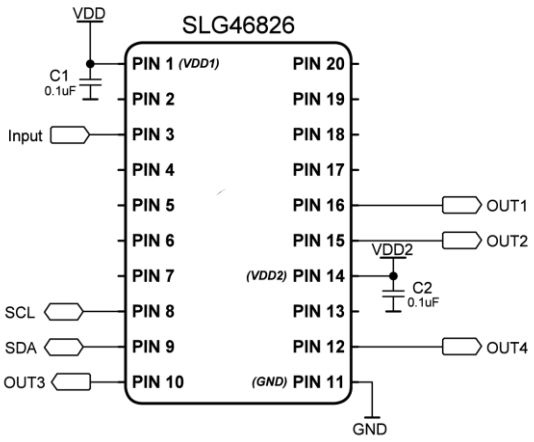


設計手順

1. GPIO ピンをクロックの入力とフラグの出力として設定します。
2. CNT/DLY ブロックを立ち上がりエッジ検出付きの「周波数検出」モードに設定します。
3. 各 CNT/DLY ブロックをそれぞれ最小周波数値と最大周波数値に設定します。
4. 周波数が目的の周波数範囲外にあるときに LUT が High になるように設定します。

アプリケーション：周波数分周器

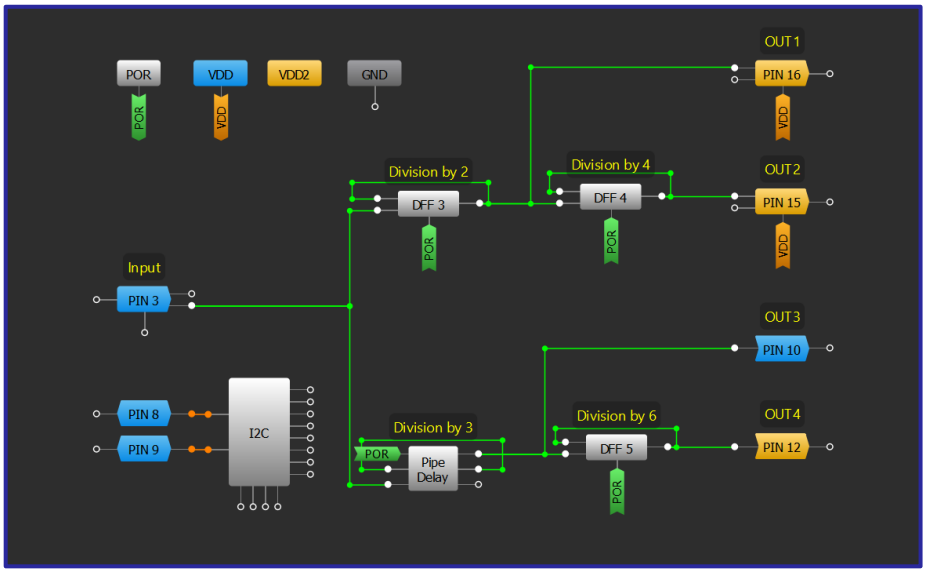
周波数分周器は周波数を複数の分周比に分周するために使用されます。電子防御装置、通信システム、および実験機器のパフォーマンスを向上させるために使用することもできます。



必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図



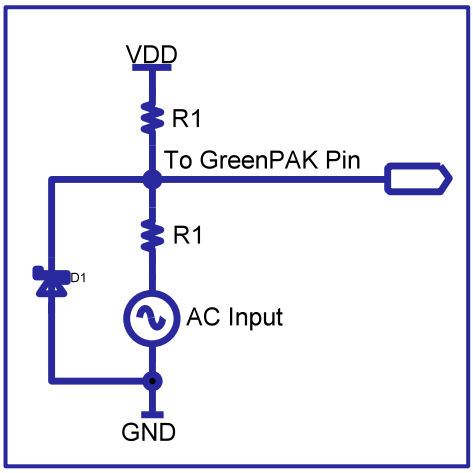
設計手順

1. 各入力信号（入力周波数、係数）用に入力を構成します。
2. DFF とパイプディレイを使用して第 1 ステージ用に周波数を分周します。
3. 別の DFF を使用して、第 2 ステージ用に係数別に信号を分周します。
4. 出力が指定の分周比になるように LUT ロジックを設定します。

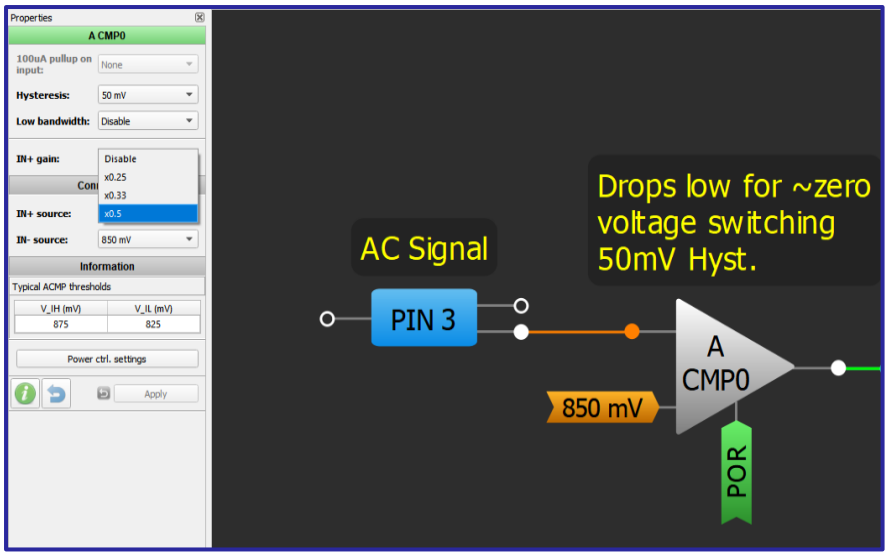
技法：ゼロ電圧クロス検出

この技法は ACMP を備えた GreenPAK で使用できます。

ゼロボルトクロス検出は、周波数やフェーズなどの AC の特性を検出する正確な方 法として一般的に使用されます。 GreenPAK のピン電圧範囲は 0V から 5.5V の VDD 値までです。GreenPAK を 使用して 0V 点で交差する AC 信号を解釈するため、DC オフセットシフトは AC 信 号と GreenPAK ピンの間に実装する必要があります。これを達成するには、図 16 のように GreenPAK の VDD と AC 信号の間に 1:1 の抵抗分割器が必要です。 ゼロ電圧クロス検出では、少なくとも 1 つか 2 つのコンパレータとカウンタが必要で す。コンパレータは、入力される AC 電圧を基準電圧に対してチェックします。基準電圧は 利用可能な GreenPAK の基準電圧または外部の基準点のいずれかから選択できます。適切なゼロボルトクロス検出電圧が利用可能な GreenPAK 基準電圧よりも高い場合、AC 信号を低減するために、コンパレータのプロパティ設定内で IN+ gain オプションを 使用し、低減した AC 値を同様のスケールの参照値と比較することができます。



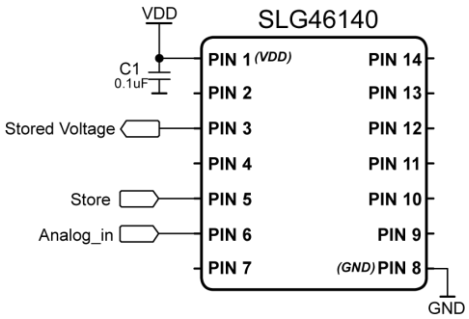
基本的な DC オフセット



IN+ Gain を使用した 1.7V クロッシングの測定

アプリケーション: アナログストレージ素子

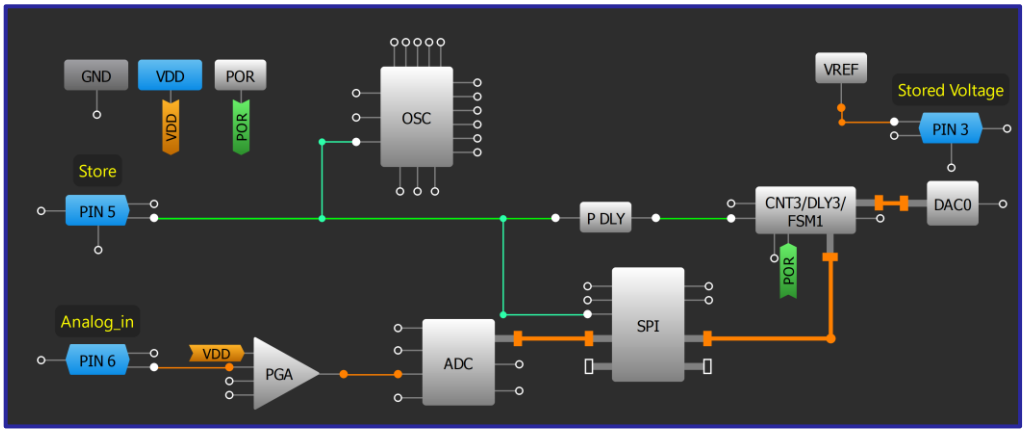
このアプリケーションを使用して、立ち上がりエッジがストア入力に印加されるまで、出力にアナログ電圧を蓄積することができます。入力および出力アナログ電圧は 0～1V の範囲内にあります。

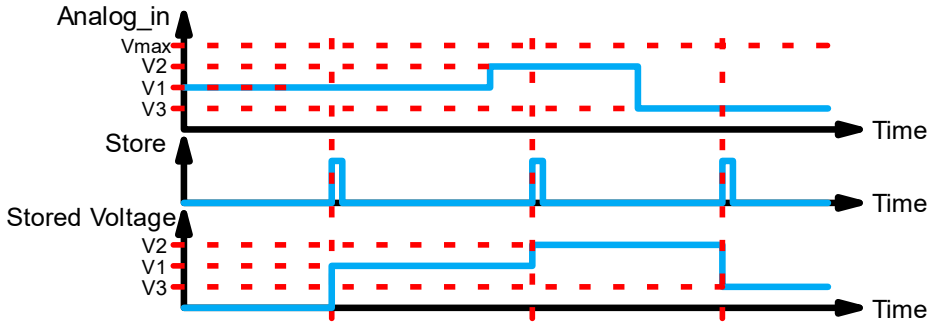


必要なコンポーネント

- **ADC**、SPI、および DAC を備えた任意の GreenPAK

GreenPAK の図





設計手順

1. SPI を「ADC/FSM バッファ」モードに設定し、**PAR 入力データソース**を「ADC」に変更します。
2. **FSM0** を "設定 (カウンタ値 = FSM データ)" に設定し、**FSM データソース**を "SPI [7:0]" に変更します。
3. DAC **入力選択**を「DCMP1 の入力から」に、VREF **ソースセレクタ**を「DAC0 出力」に設定します。
4. ストア入力を SPI **SCLK** に直接接続し、両エッジ遅延として設定した P DLY を介して FSM1 **SET IN** に接続します。

第 4 章：安全機能

この章では、システム内の異常に対して応答し、故障から守るアプリケーションについて紹介します。電子システムに安全性を与えるアプリケーションとしては、バッテリー表示、ウォッチドッグタイマー、温度センサーなどがあります。

技法：ACMP 電力消費量の削減

この技法は ACMP を含む GreenPAK で使用できます。電力消費の削減量は異なります。

GreenPAK CMIC は、システムの電力消費を削減するためのプロジェクトでよく使用されます。ただし、いくつかのコンポーネ ントが GreenPAK 内で有効になっている場合、電力消費を大きく変化させる可能性があります。最も消費されるマクロセルの 中には、アナログコンパレータがあります。表 1 は ACMP の電力消費を強調するために SLG46826 データシートから引用したも のです。

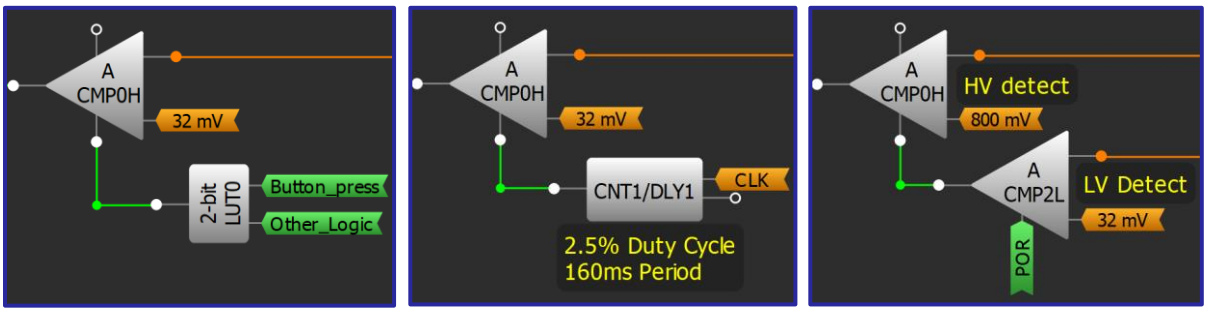
ACMP は使用していないときはシャットダウンすることができます。シャットダウンには次 の 2 つの方法があります。

1. ACMP の PWR UP 入力を使用する 方法。
2. ACMP のウェイク/スリープ（WS） カウンタを有効にする方法。

表 1：SLG46826 の電力消費

Note	V_{DD} = 2.5 V	V_{DD} = 3.3 V	V_{DD} = 5.0 V	Unit
Chip Quiescent	0.39	0.43	0.53	µA
Vref OUT0 (Source none, Source Temp Sensor, Buffer On)	12.79	12.95	13.57	µA
Vref OUT0 (Source none, Source Temp Sensor, Buffer Off)	7.62	7.67	7.87	µA
Vref OUT1 (Source none, Buffer On)	6.53	6.61	7.02	µA
Vref OUT1 (Source none, Buffer Off)	1.40	1.44	1.54	µA
Vref (ACMPxH, 0.32 mV, Buffer On)	12.24	12.59	12.21	µA
Vref (ACMPxL, 0.32 mV, Buffer On)	6.93	7.01	7.43	µA
ACMP0H, 1H, 2L, 3L,hysteresis disabled, gain = 1, +IN - IO11, 12, 13, 14 Pull Up 1M, Vref = 32 mV	65.86	67.12	70.77	µA
ACMP0H, 1H, 2L, 3L,hysteresis disabled, gain = 1, +IN - IO11, 12, 13, 14 Pull Down 1M, Vref = 32 mV	37.34	38.05	40.29	µA
ACMP0H, 1H, 100 µA disabled, hysteresis disabled, gain = 1, +IN - IO13, 14 Pull Up 1M	63.85	65.11	68.71	µA
ACMP0H, 1H, 100 µA disabled, hysteresis disabled, gain = 1, +IN - IO13, 14 Pull Down 1M	35.97	36.68	38.87	µA
ACMP0H, 100 µA disabled, hysteresis disabled, gain = 1, +IN - VDD, Vref = 32 mV	36.30	36.96	38.85	µA
ACMP0H, 100 µA enabled, hysteresis disabled, gain = 1, +IN - IO14 Pull Up 1M, Vref = 32 mV	46.77	47.31	49.23	µA
ACMP0H, 100 µA enabled, hysteresis disabled, gain = 1, +IN - IO14 Pull Down 1M, Vref = 32 mV	49.02	50.29	53.75	µA

ウェイク/スリープのコントロールには、WS モードに合わせて設定された専用のカウン タが必要です。これは多くの（すべてではあ りませんが）GreenPAK で利用できます。 PWR UP コントロールは ACMP を備えたど の GreenPAK でも使用できます。信号が HI の場合、ACMP がオンになります。ロジッ ク、カウンタ、または ACMP をオフにするた めのその他のマクロセルを使用すると電力 消費を大幅に抑えることができます。たとえ ば、2 つの電力のしきい値が必要な場合、 低い方のしきい値が満たされるまで、高い 方のしきい値の ACMP を無効にしておくこと ができます。



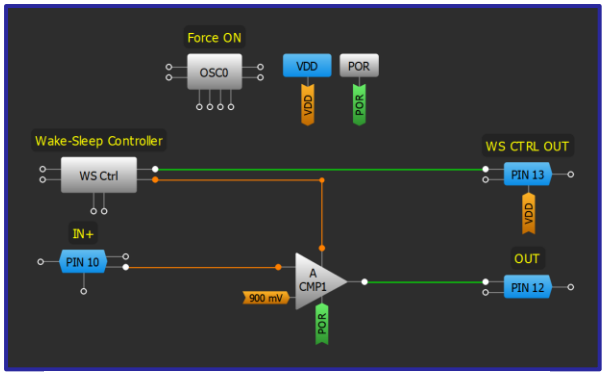
(a) ロジック　(b) 間欠サイクル　(c) カスケード

一般的な PWR UP の構成

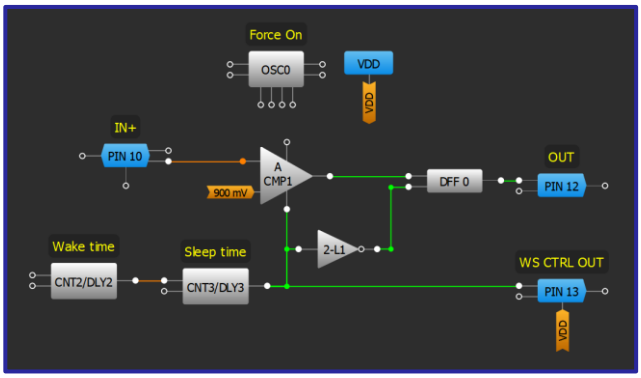
技法: ウェイク/スリープのコントローラー

アナログ マクロセルのウェイク/スリープ制御は、消費電力の削減に役立ちます。これは、ACMP やADC などのアナログ マクロセル用のウェイク スリープ コントローラーで実現できます。

ウェイク/スリープは、アナログ マクロセルの定期的なオンとオフの切り替えに関わります。一部の GreenPAK では、WS Ctrl ブロックを使用してこの機能を実装できます。このブロックがない場合は、2 つのカウンタ (ウェイク時間を変更する必要がない場合は 1 つのカウンタ)、一つの D フリップフロップとインバータを使用して実装できます。上図は、それぞれの方法の使用例です。



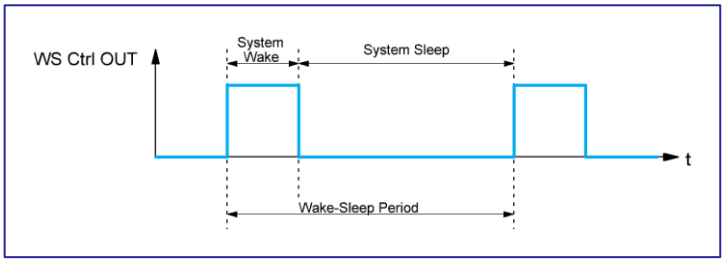
WS Ctrl ブロック による方法



2 つの カウンタによる方法

ウェイク/スリープが実装されていない場合、合計消費電流は次で構成されます:

- 静的電流
- ACMP 電流



ウェイク/スリープの動作

ウェイク/スリープが実装されている場合は、ウェイク/スリープ電流はおおよそ以下です:

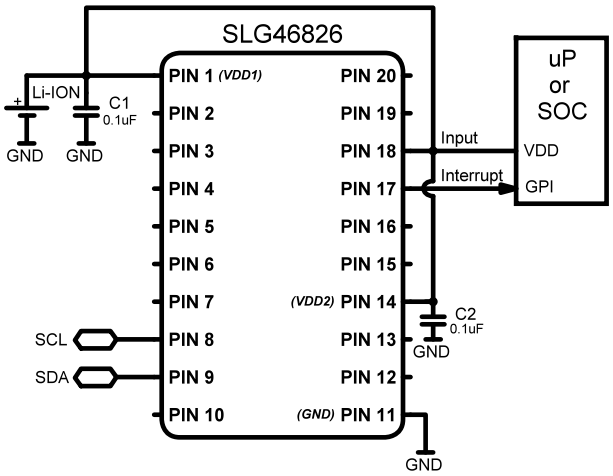
$$I_{WS} = \frac{System\ Wake}{System\ Wake + System\ Sleep} * I_{without\ WS} = \frac{System\ Wake}{WS\ Period} * I_{without\ WS}$$

ウェイク/スリープ実装時の総電流は以下です:

$$Total\ Current = I_{Quiescent} + I_{OSC} + I_{Wake\ Sleep}$$

アプリケーション：ウィンドウコンパレータ

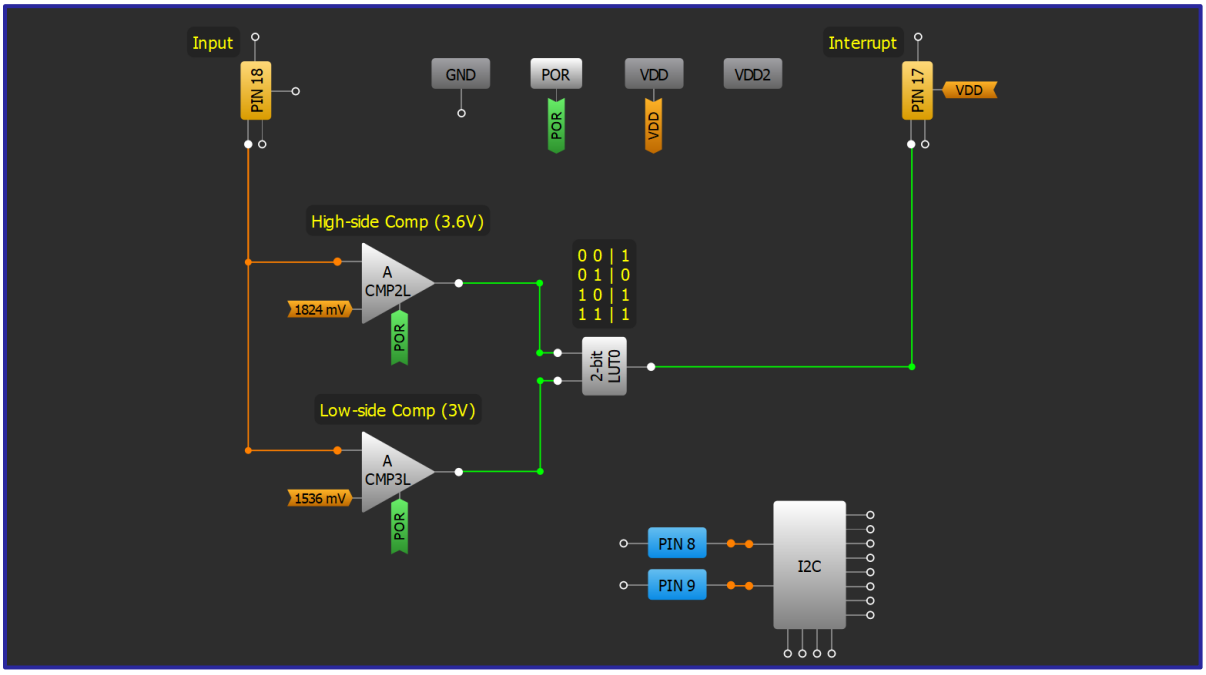
ウィンドウコンパレータは、バッテリーやスーパーキャパシタなど消耗する電源で動作するシステムの設計には、欠かすことのできない要素です。バッテリーの電圧を監視することにより、低バッテリー レベルでは不必要なリソースをデバイスが使用しないように選択 することができます。これによりデバイスの修復不能な破損を防止 することができます。



必要なコンポーネント

- ACMP を備えた任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

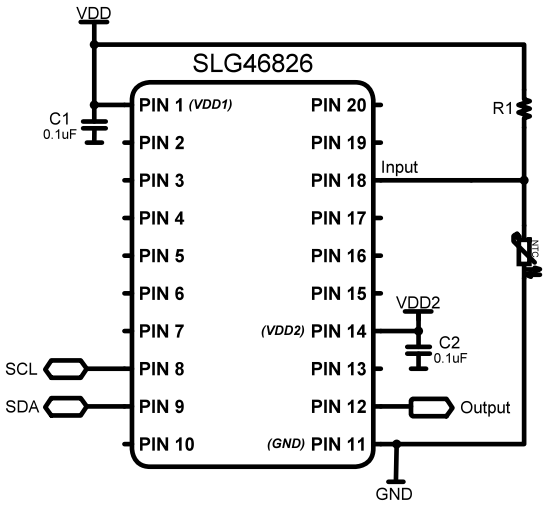


設計手順

1. IN- 入力と IN+ gain オプションを使用して適切な上方しきい値を設定し、上方の ACMP2L を構成します。
2. 下方しきい値を使用して下方の ACMP のために手順 1 を繰り返します。
3. 2 番目のコンパレータの IN+ 入力を ACMP2L IN+ 入力に変更します。
4. Low-side Comp が低い、または High-side Comp が高い場合に割り込みをトリガーするための LUT ロジックを追加します。

アプリケーション：過熱保護

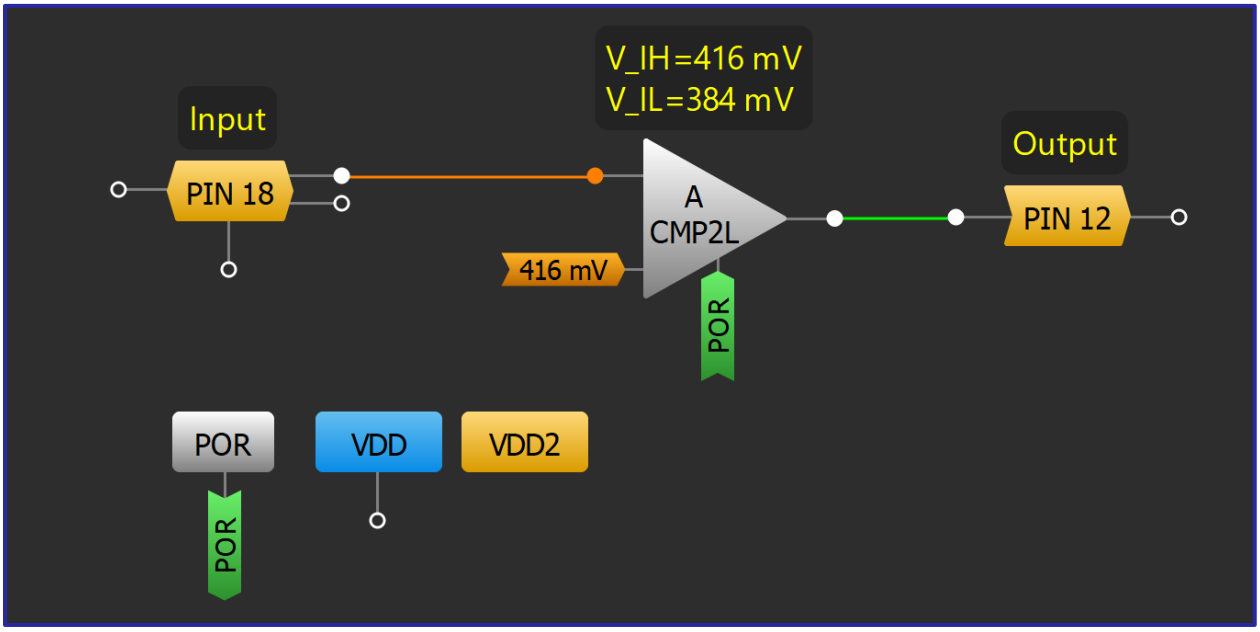
過熱保護回路はシステムに高温を警告するために広く使用されます。この回路により、 システムは内部温度が安全しきい値を超えた場合に過熱から保護されます。



必要なコンポーネント

- ACMP を備えた任意の GreenPAK
- 1 つの抵抗
- 1 つの NTC サーミスタ

GreenPAK の図

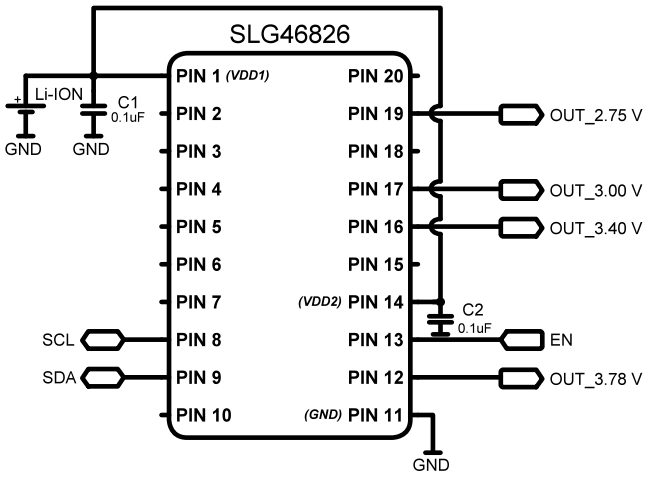


設計手順

1. ACMP2L の IN+ 入力をピン 18 に設定し、IN- 入力を適切なしきい値に設定します。
2. 抵抗の 1 つのノードを VDD に接続し、2 番目のノードをピン 18 に接続します。
3. NTC サーミスタの 1 つのノードをピン 18 に接続し、2 番目のノードを GND に接続します。

アプリケーション：バッテリー充電インジケータ

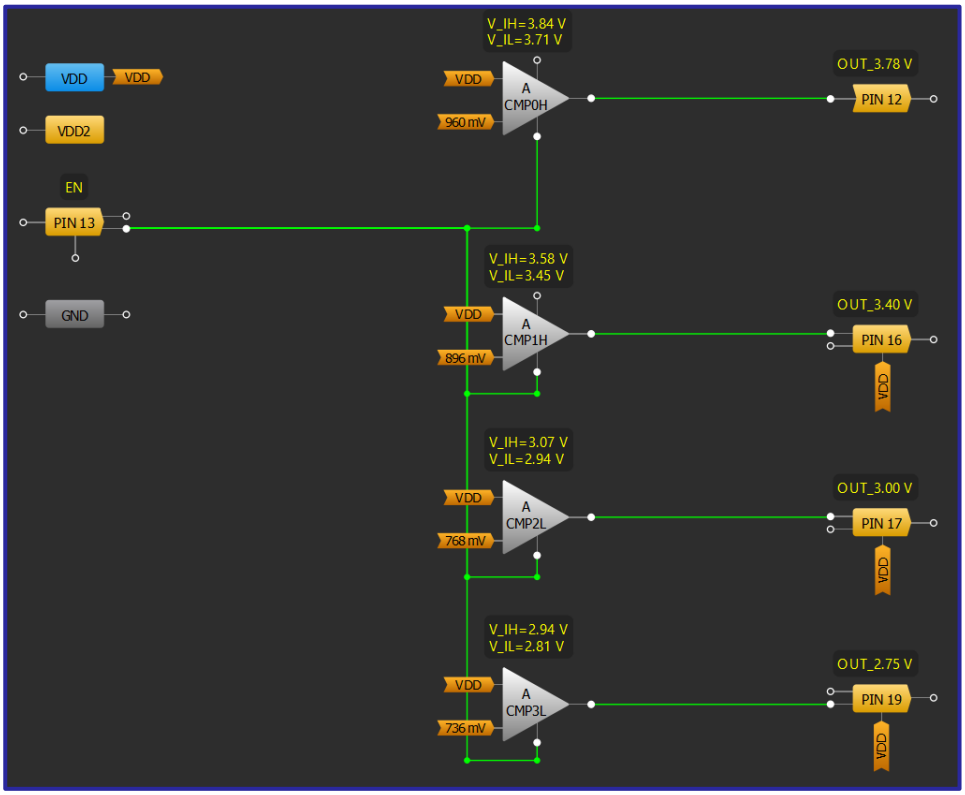
バッテリー充電インジケータは充電状態を示すためにバッテリー式のデバイ スで使用されます。この設計はリチウムイオン電池用に最適化されていま す。



必要なコンポーネント

- ACMP を備えた任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

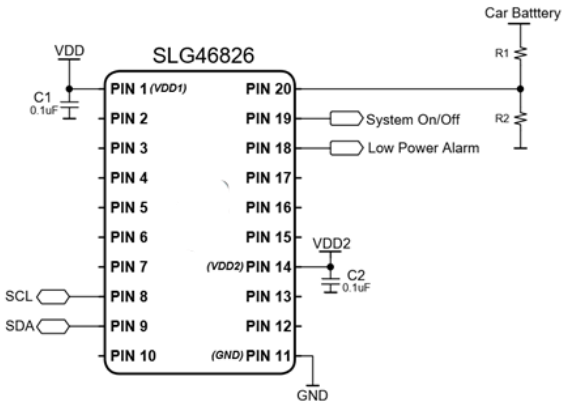


設計手順

1. イネーブルを ACMP0H、ACMP1H、ACMP2L、ACMP3L の PWR UP ピンに接続します。
2. すべての ACMP の IN+ 入力を VDD/ピン 20 に設定し、各 IN- 入力を適切なしきい値レベルに設定します。

アプリケーション: インフォテイメント用低電圧インジケータ

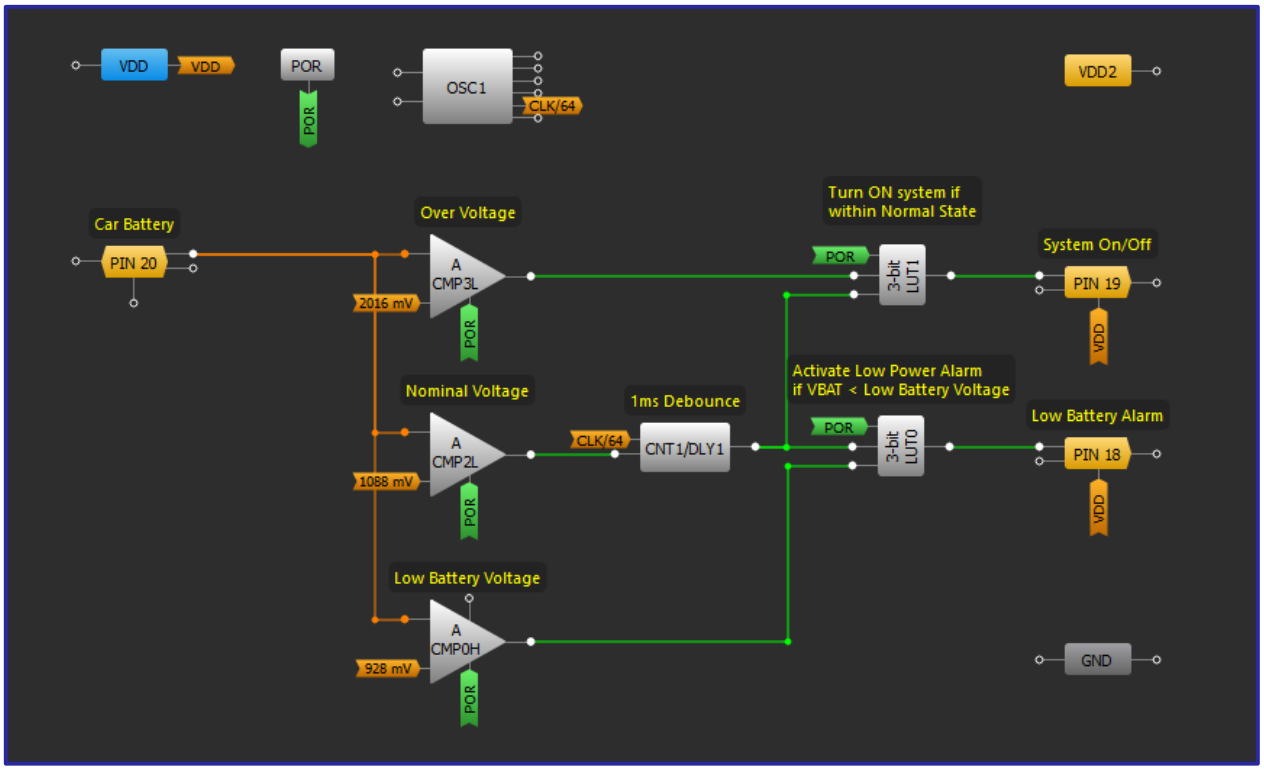
電圧インジケータは、バッテリ駆動のデバイスで充電状態を示すために使用されます。このデバイスは、車のバッテリーの電圧レベルを監視し、必要に応じてインフォテインメント アクティビティを調整して電力を節約します。



必要なコンポーネント

- 任意の GreenPAK
- 分圧器のための 2 つの抵抗

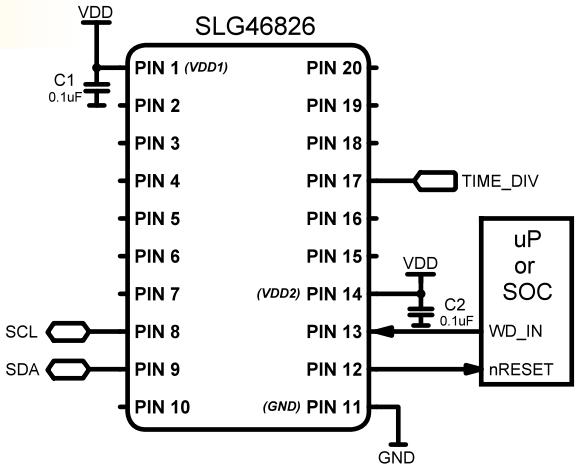
GreenPAK の図



設計手順

1. ACMP0H IN+ ソースを「PIN 20」として、他の ACMP を「ACMP0H IN+ ソース」として構成します。
2. PIN 20 に分圧器を追加して、車のバッテリーからの高電圧に対応します。
3. IN-ソースを所望のしきい値電圧に設定します。
4. 出力する電圧レベル ウィンドウを決定するロジックを構成します。
5. ACMP2L と 3 ビット LUT0 の間にデバウンス遅延を追加します。

アプリケーション: ウォッチドッグタイマー

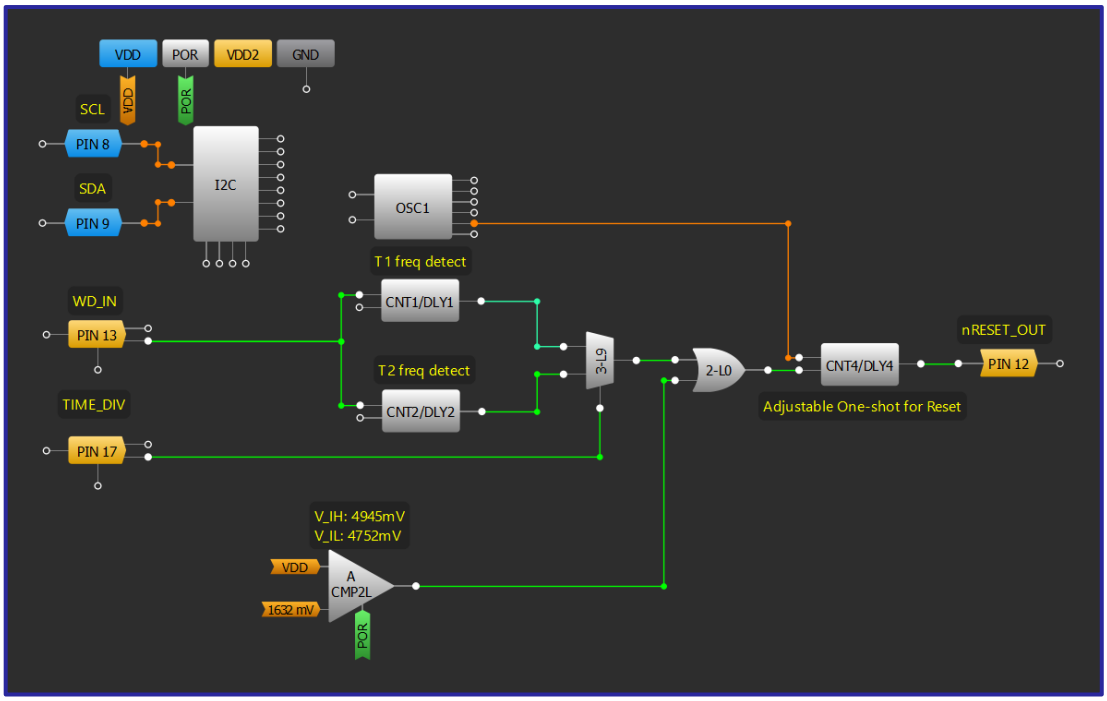


ウォッチドッグ タイマーは、マイクロコントローラまたはマイクロプロセッサが定期的にパルスを送信しない場合にシステム リセット信号を自動的に生成するために使用されます。電源電圧低下の監視は、ウォッチドッグ IC に一般的に付加される機能です。

必要なコンポーネント

- ACMP を備えた任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

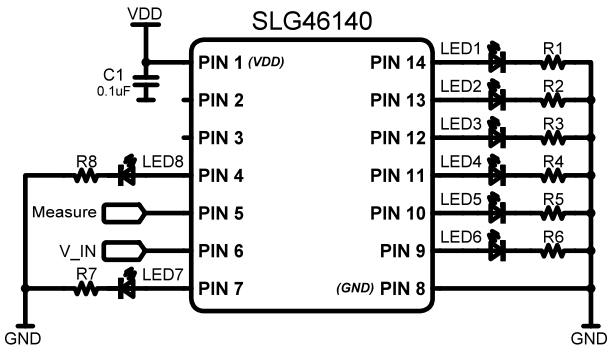


設計手順

1. ACMP で低電圧検出しきい値を設定します。
2. 2 つの CNT/DLY ブロックを「周波数検出」モードに設定します。
3. 低電圧とウォッチドッグ タイムアウトからのアクティブ信号を組み合わせるデジタル論理を設計します。
4. ワンショット ブロックを追加して、リセットパルスをトリガーします。反転してアクティブ LOW にすることができます。

アプリケーション: 電圧レベル検出

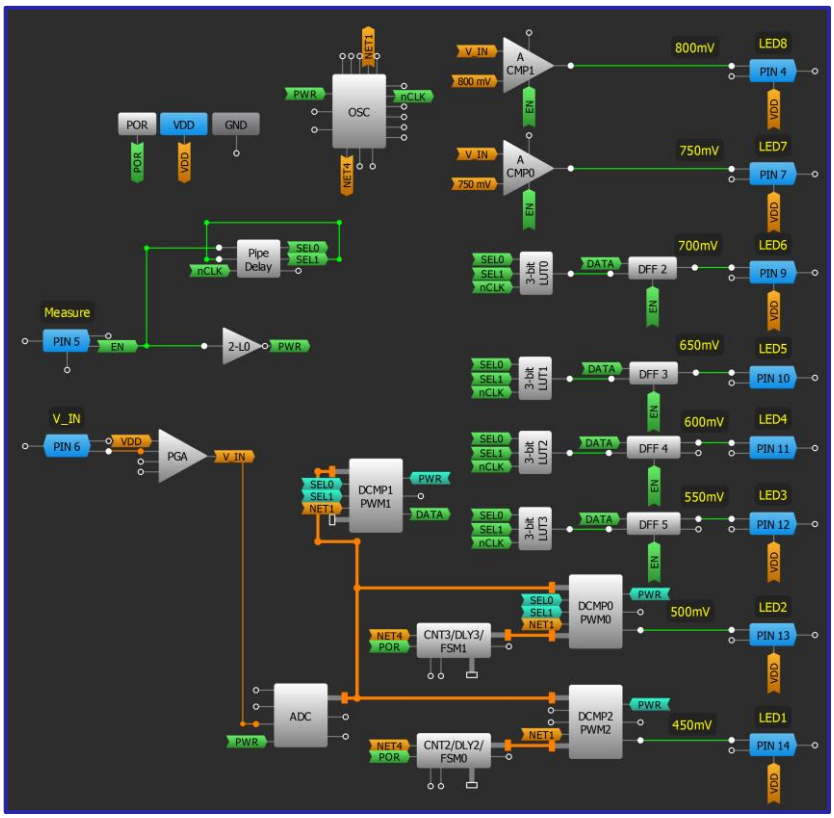
一部のアプリケーションでは、数少ない別個のレベルではなく、複数の電圧レベルを評価する必要があります。このアプリケーションでは、ACMP、DCMP、および ADC を用いて電圧振幅を監視する方法を示します。



必要なコンポーネント

- ACMP/DCMP/ADC を有する GreenPAK
- 8 個までの LED と抵抗

GreenPAK の図

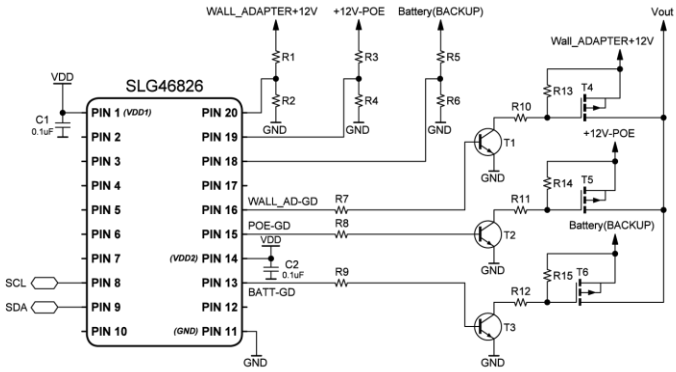


設計手順

1. ADC、DCMP、および ACMP の電源を入れます
2. 「技法: PWM モードでの DCMP/PWM マクロセルの使用」 を用いて DCMP を設定します。
3. 各 ACMP および DCMP の IN- を所望の電圧しきい値レベルに設定します。
4. DCMP1 からのアナログ電圧の振幅のデータを選択して書き込めるように LUT と DFF ロジックを追加します。

アプリケーション: 電源バックアップ管理

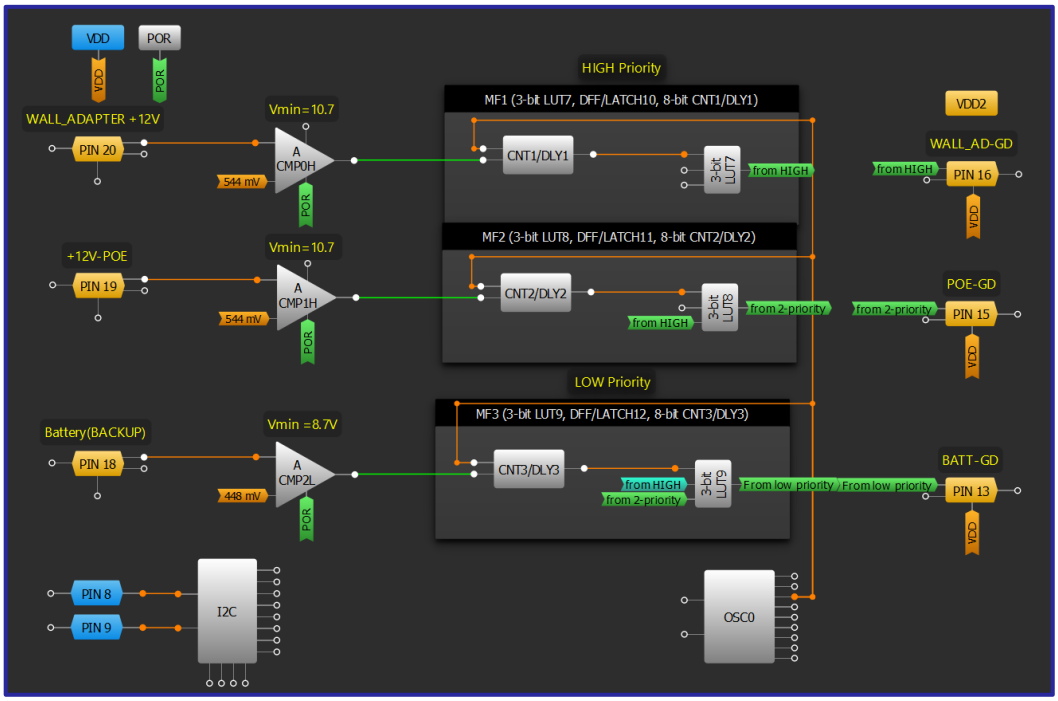
電源バックアップ管理は、設計者が異なる電源からのシステムの無停電電源を保証する必要がある場合に使用されます。



必要なコンポーネント

- 3 つの ACMP を有する GreenPAK
- 入力信号を ACMP の動作値範囲に減衰させる外部抵抗分圧器。

GreenPAK の図

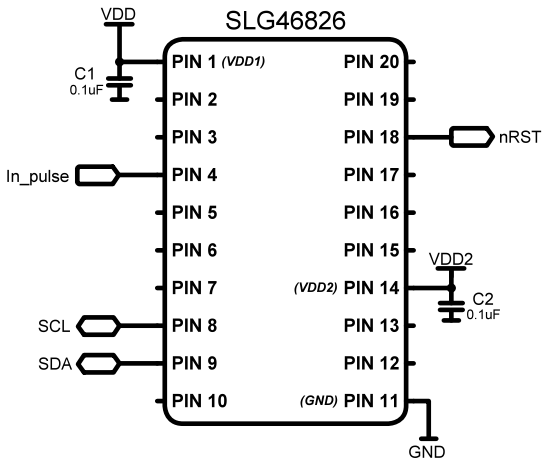


設計手順

1. 3 つの ACMP を使用して電源入力信号を検出します
2. 遅延として構成された CNT/DLY ブロックを使用して、デバウンス フィルターを実装します。
3. 論理セルを追加して、入力ソース間の優先順位を切り替えます。

アプリケーション: N パルス監視ウォッチドッグ

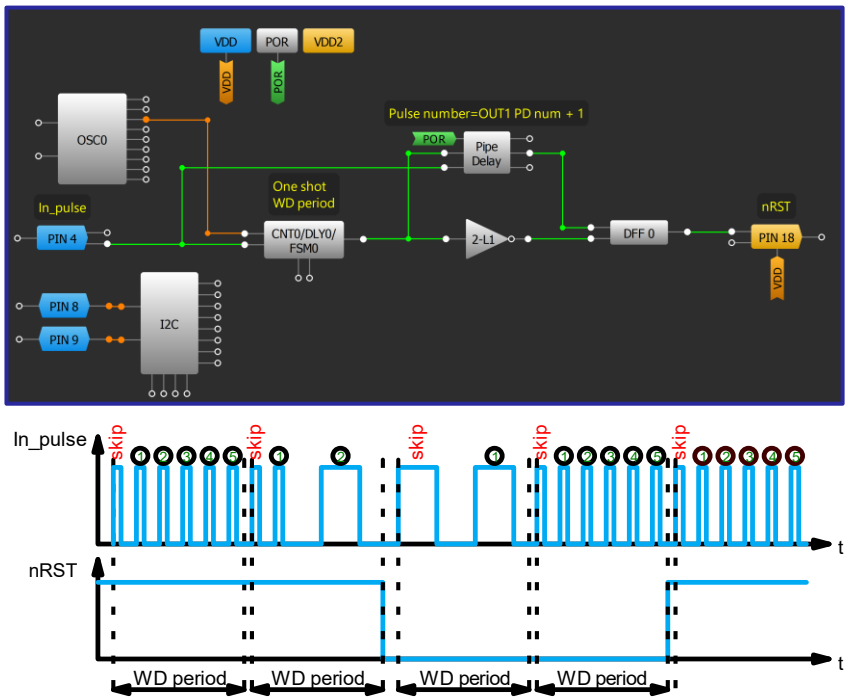
ウォッチドッグタイマーは、マイクロコントローラまたはマイクロプロセッサが定期的にパルスを送信しない場合に、システムリセット信号を自動的に生成するために使用されます。このアプリケーションは、ウォッチドッグ期間中に GreenPAK に送られるパルス数を監視します。数が事前に定義されたパルス数よりも少ない場合、システムリセット信号を発生します。



必要なコンポーネント

- 任意の GreenPAK

GreenPAK の図



設計手順

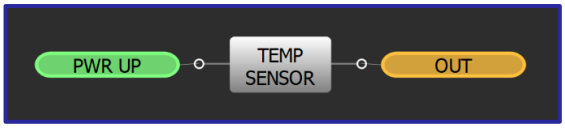
1. CNT0/DLY0/FSM0 を希望のウォッチドッグ期間でワンショットとして構成します。
2. パイプ遅延でパルス数を定義します (注: パルス数 = OUT1 PD num + 1)。
3. ワンショットの出力を反転し、DFF0 の CLK 入力に接続します。
4. Pipe Delay の nOUT1 を DFF0 の D 入力に接続します。

技法: 温度センサーブロックの使用

この技法は、内部に温度センサー マクロセルがある任意の GreenPAK で使用できます。

一部の IC には、摂氏温度に比例する出力電圧を持つアナログ温度センサー (TS) があります。 TS は、-40°C ～ 85°C の温度範囲で動作します。全温度範囲での誤差は ±0.85% を超えません。 TS 出力は、アナログ出力 (Analog Output)または ACMP のポジティブ入力に直接接続できます。 TS には 2 つの出力電圧範囲と 1 つの電源投入(Power Up)入力があります。 Power Up は、マトリックス入力を使用して、またはレジスタからオプションで有効にすることができます。 TS を I2C でアクティブにすることもでき、電圧範囲も I2C 経由で変更できます。一定温度での TS 出力電圧は、VDD の変化に対する変動が非常に小さくなっています (たとえば、SLG46826 では、出力電圧誤差はすべての温度で ±0.08% 未満です)。

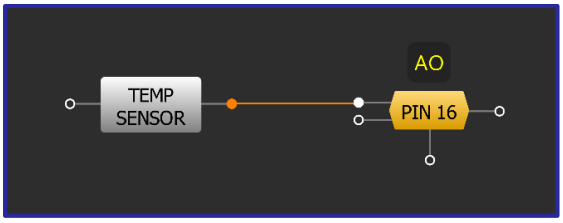
温度センサーマクロセル



TS の出力電圧は以下の式で求められます:

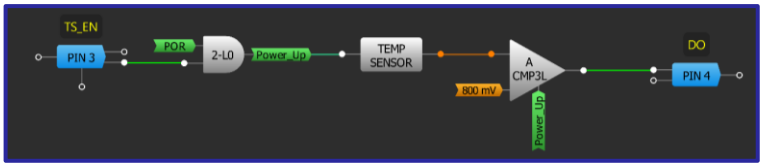
$$V_{ts} = K \times T + V0$$

ここで
V_{ts} - TS 出力電圧、K -係数、T – 温度(°C)、V_0 – 温度 0 °C の時の出力電圧



Analog 出力への接続

温度比例電圧信号は、アナログ出力 (PIN16) に適用できます。パワー ダウン ソース コンフィギュレーションは「From register」に設定する必要があります。温度センサーの出力信号は、ディスクリート出力で 2 状態信号を生成する ACMP ブロック (下図を参照) の基準電圧と比較できます。消費電力を削減するために、TS_EN が使用されます。 TS_EN は温度センサーを有効にし、ACMP3L をオンに切り替えます。パワーダウンソースは「マトリックスから」に設定する必要があります。



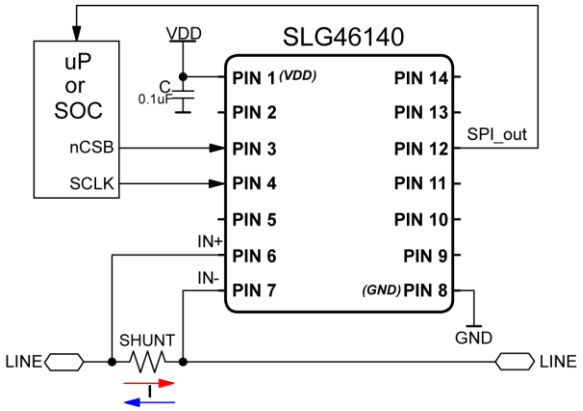
温度比較器

温度センサー付き GreenPAK では以下を実行できます :

- PCB コンポーネントの温度を測定する
- FET または BJT のケース温度を測定する
- SoC または内部のクローズドループアプリケーションにアラーム信号を発生する
- ADC、DAC、オペアンプ、およびその他の温度に依存する回路のエラーを最小限に抑える

アプリケーション: 外部センス抵抗による電流検出

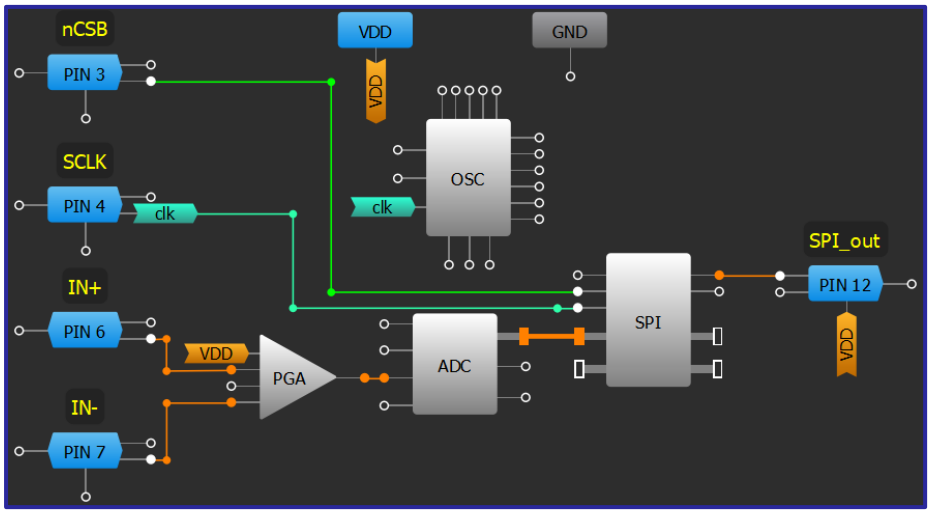
GreenPAK を使用して、検出抵抗の両端の電圧を検出することにより、デバイスを流れる電流を検出できます。このアプリケーションは、感知した値を表すシリアル コードを出力します。



必要なコンポーネント

- PGA/ADC/SPI を有する GreenPAK
- 1 つの抵抗

GreenPAK の図

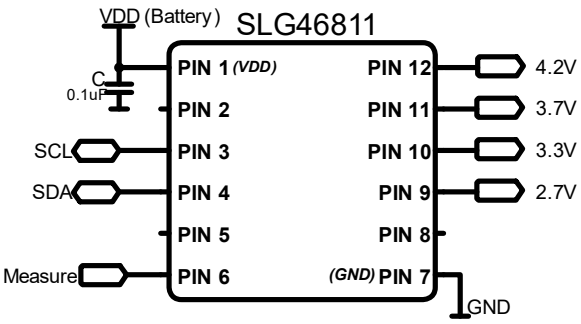


設計手順

1. PWR DOWN 入力から VDD を除去して、ADC をパワーアップします
2. PGA を「差動」モードに設定します。 ADC に、PIN6、および PIN7 が自動的に接続します
3. SPI を「P2S」モードに設定し、PAR 入力データ ソースを「ADC」に変更します

アプリケーション: 1 つのアナログ信号の 4 つのレベルを監視する

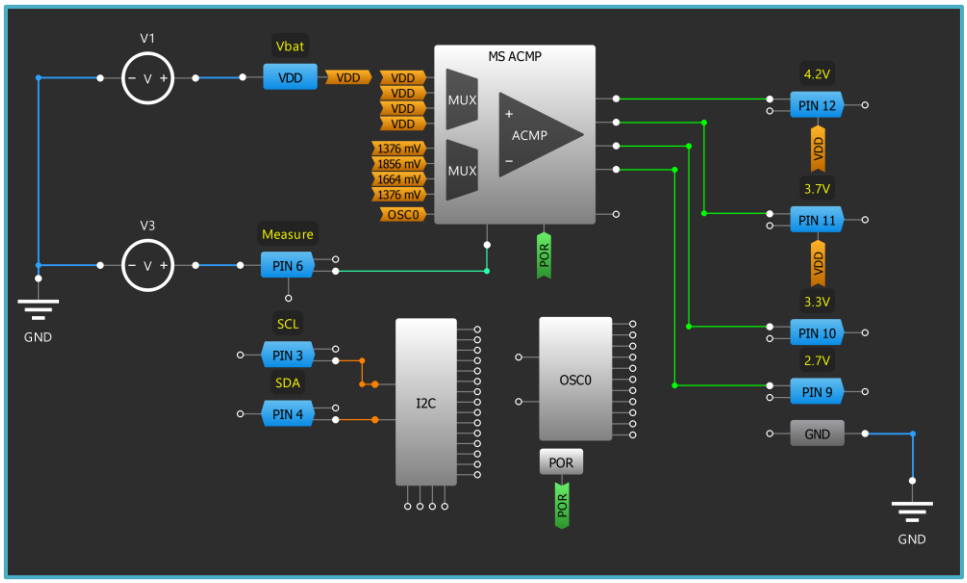
GreenPAK を使用して 1 つのアナログ信号の 4 つのレベルを監視すると、さまざまなアプリケーションで役立ちます。たとえば、バッテリー管理、液面制御、温度、光、近接、圧力、湿度の検知などに使用できます。



必要なコンポーネント

- SLG46811V または適切な数の ACMP を内蔵する任意の GPAK

GreenPAK の図

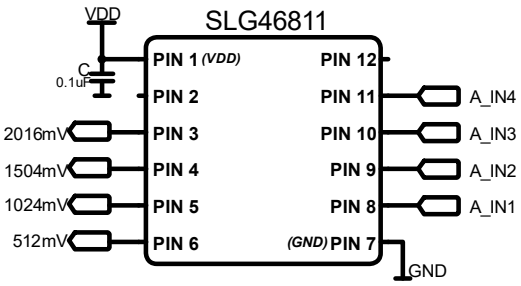


設計手順

4. MS ACMP を Multi-channel モードに設定し、4 チャネルを選択します。
5. MS ACMP をイネーブルするために、立上りエッジのアクティベーションを選択します。
6. 入力信号をチャネル 0 から 3 で同一にします。
7. PIN6 をデジタル入力として設定し、MS ACMP のイネーブル入力に接続します。 PIN6 に印加されるすべての立ち上がりエッジによって、MS ACMP は VDD 電圧を測定し、結果を PIN9 ～ PIN12 に出力します。
8. ACMP の閾値の値は、I2C で書き換え可能です。

アプリケーション: 4つのアナログ信号をモニターする

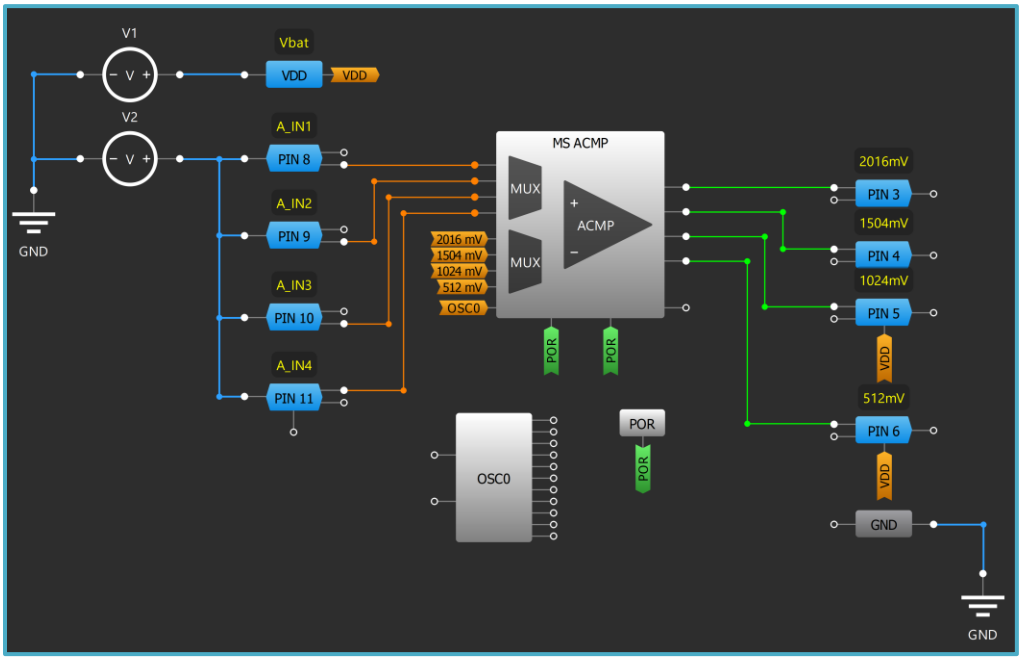
GreenPAK を使用して4つの個別のアナログ信号をモニターすることにより、ゆっくり変化する4つの電圧を持つアプリケーションで役に立ちます。



必要なコンポーネント

- SLG46811V または適切な数の ACMP を持つ任意の GPAK

GreenPAK の図



設計手順

1. MS ACMP を Multi-channel モードに設定し、4 channels を選択します。
2. High-Level Activation を選択し、POR (MS ACMP は継続してサンプリングを行います)を MS ACMP の ENABLE 端子に接続します。
3. IN-source を Channel 0 から Channel 3 とします。PIN8 から PIN11 の電圧が基準電圧と比較され、結果がそれぞれ PIN3 から PIN6 に出力されます。
4. I2C により ACMP の基準電圧の書き換えができます。

第 5 章：通信プロトコル

この章ではデバイス間の通信に関わるアプリケーションを紹介します。具体的には、I2C、シリパラ変換通信プロトコルなどのアプリケーションと技術に関わるものです。

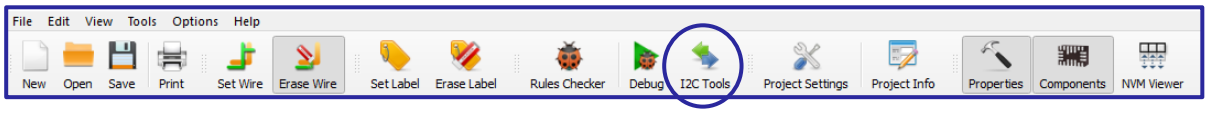
このセクションで利用可能な技法とアプリケーションの多くは、GreenPAK の I2C 機能に依存しています。GreenPAK 内の I2C についての詳細はチップのデータシートを参照してください。

技法：I2C を使用した設計の変更

この技法は任意の I2C 対応デバイスで使用できます。

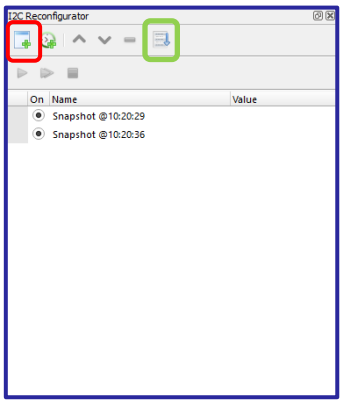
GreenPAK デバイスが I2C 対応である場合、その動作はプログラムされた後でも編集できます。ただし、デバイスは MTP 対 応でなければならず、電力供給の停止後も設計変更を保持するにはインシステムプログラミング（ISP）を使用する必要があり ます。この技法では設計を変更するために実行する必要のある I2C コマンドをすばやく決定する方法の概要を示します。

1. 初期設計を完了させます。この設計が IC の起動時に使用されます。
2. GreenPAK Designer で[I2C Tools]ボタンを選択して[I2C Reconfigurator]を開きます。

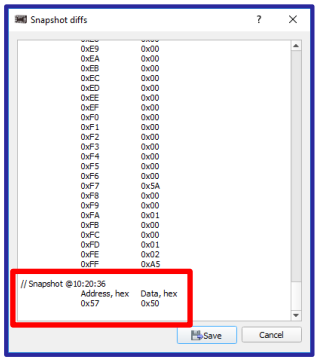


[I2C Tools]ボタン

3. [I2C Reconfigurator] でスナップショットボタン（赤色） を選択するか、 SHIFT+A を押します。これにより、現在の設計の I2C コマンドリスト「スナップ ショット」が取得されます。
4. 設計を次の構成に変更します。
5. 手順 3 の方法でスナップショットを取得し、2 番目のスナップショットを作成し ます。
6. スナップショットの差異ボタン（緑色）をクリックします。これにより、この設計を 作成するために必要な I2C コマンドが表示されます。これらは GreenPAK の起 動時にインスタンス化されるため、プログラムする必要はありません。
7. スナップショットの差異リストをスクロールダウンし、2 番目のスナップショットを 探します。1 番目のスナップショットと 2 番目のスナップショットで変更されてい る値のみが表示されます。
8. 右側の図で赤のボックスで示した値は、ACMP のしきい値を変更するため（また 発生しうるその他の変更のため）に I2C で送信するために必要な 16 進アドレスお よびデータ値と相互関係があります。

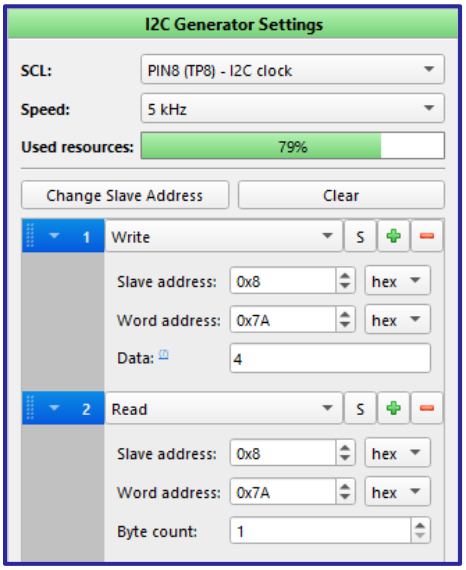


I2C Reconfigurator



スナップショットの差異

技法: I2C コマンドの生成



この技法は、I2C マクロセルを備えた任意の GreenPAK で使用できます。

I2C ジェネレータを使用すると、ユーザーはロジック ジェネレータに基づいて I2C 信号を作成できます。これは、SDA および SCL ラインとして機能する 2 つのロジック ジェネレータで構成されています。ユーザーは、定義済みの I2C プリミティブを組み合わせて必要な波形を生成し、SCL 周波数を選択できます。GreenPAK Advanced 開発プラットフォームでは 1k、2.5k、または 5kHz で、 GreenPAK Pro 開発プラットフォームでは 1k、2.5k、5k、10k、20k、50k、100k、200k、400k、1000 kHz のいずれかです。

I2C Generator を用いて I2C 信号を作成する手順 :

Debug ボタンを選択します

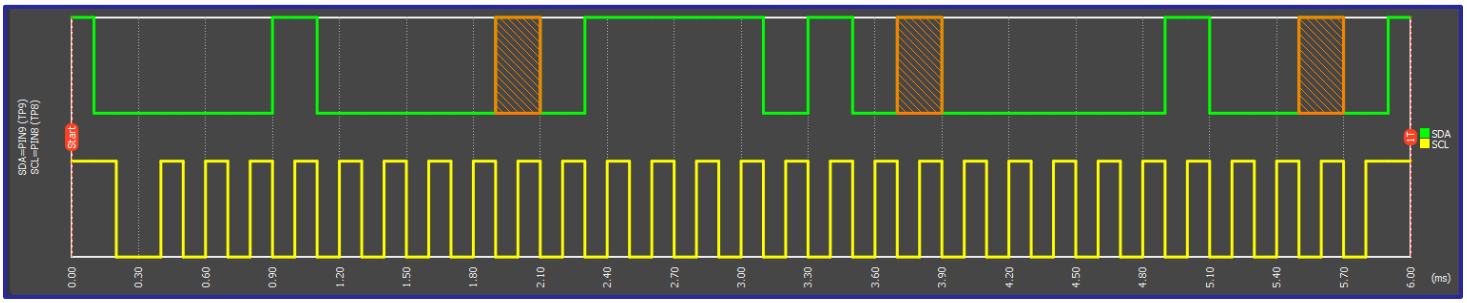
I2C ブロックの SDA 入力の外部コネクタ設定で「I2C ジェネレータ」を選択します

EDIT をクリックして Signal 画面に移動します

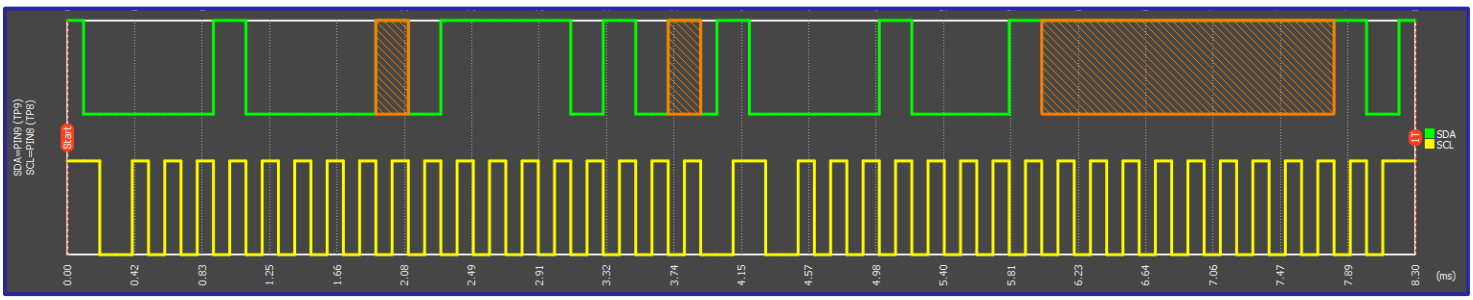
PIN8 を **SCL** として選択してクロックのスピードを設定します

Read または Write コマンドを選択します.

複合コマンドを開いて Slave アドレスと Word アドレスを設定します。 **Read** コマンドはバイト数を、**Write** コマンドは書き込むデータをセットします。



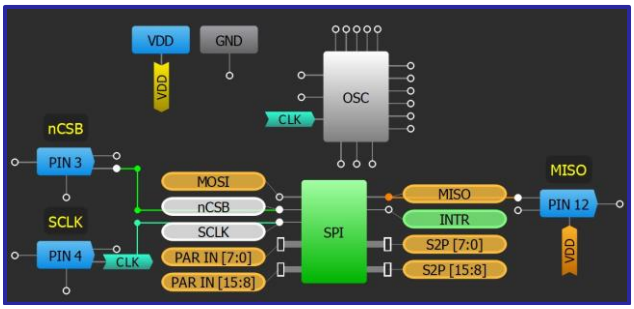
I2C Write コマンド



I2C Read コマンド

技法: Serial to Parallel Interface (SPI) ブロックの使用

この技法は、Serial to Parallel Interface (SPI) ブロックのもので、 SLG46140、SLG46620、 および SLG46621 で使用可能です。このブロックは、GreenPAK と SOC 間の通信に使用できる特別なマクロセルです。シリアルデータをパラレルに、またはパラレルデータをシリアルに変換できます。入力は、標準の SPI I/O 接続 (MOSI、MISO、nCSB、SCLK、および INTR) です。 nCSB はアクティブ LOW の チップ セレクトです。 SCLK は、SPI マクロセルに入力されるシリアル クロックです。



SPI macrocell

SPI は以下のようなブロックにデータを伝えるために使われます。

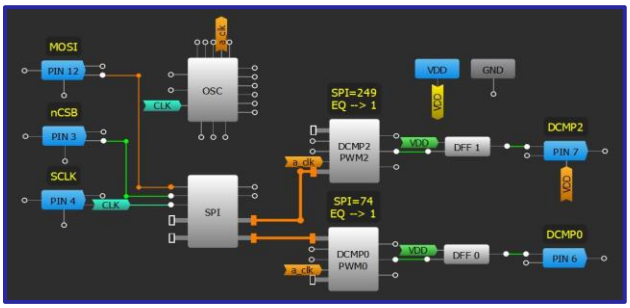
●FSM ●DCMP ●DAC (DCMP を通して)

SPI は以下からのデータを受けて、送信します。

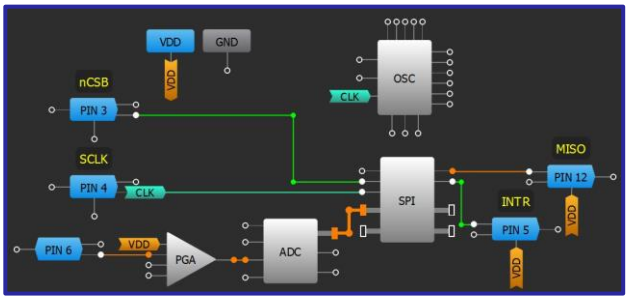
●ADC ●FSM

これらは、他のマクロセルとともに以下のような機能に使うことができます：

- パルス幅変調
- アナログとデジタルへの比較
- デジタルとアナログの比較
- DCMP による 2 つの結果の比較
- SDIO と LCD



serial から parallel モードへの SPI マクロセル



parallel から serial モードへの SPI マクロセル

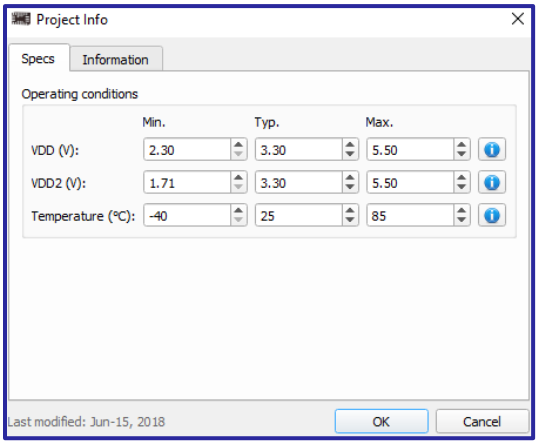
SPI は 8 ビットまたは 16 ビットで動作するように選択できます。 SPI マクロセルは、同じプログラム ファイルでシリアル データを送受信できないことに注意してください。 「S2P」または「P2S」モードに設定する必要があります。

「P2S」モードでは、データの送信が完了するたびに、INTR ピンが 1 クロック周期の間 High になります。それ以外については、GreenPAK に実装されている SPI は、一般的な標準を満たしています。クロック周波数は最大 2 MHz まで設定できます。 CPOL ビットでクロック極性を設定し、CPHA ビットでクロック位相を設定することもできます。 CPHA = 0 の場合、データはシリアルからパラレルにのみ送信でき、パラレルからシリアルには送信できません。 CPHA = 1 の場合、データはシリアルからパラレル、パラレルからシリアルの両方に送信できます。

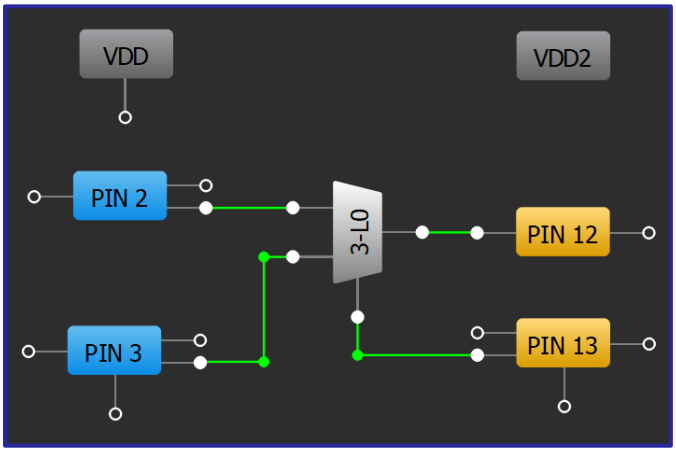
技法：レベルシフト回路

この技法は、SLG46826V などのデュアル電圧レールを備えた任意の GreenPAK で有効です。

システムレベルの設計で、異なる電圧レベルで動作する 2 つの信号 からのデータを組み合わせる必要があることがよくあります。たとえ ば、システムのアナログレールが 5.0V で動作し、デジタルレールが 3.3V で動作することがあります。 多くの GreenPAK IC では、この問題を解決するため、デュアル電圧 レールを使用します。異なるレールで動作する信号は GreenPAK への入力となって処理され、いずれかの電圧レールレベルで GreenPAK からの出力とすることができます。 デュアルレールパーツを使用して GreenPAK Designer の新しい設計 を開始するとき、両方のレールについて電圧範囲を入力するよう求め られます（図 21）。両方のレールで利用可能な範囲はパーツごとに異 なります。高電圧レールは VDD2 ではなく、VDD として指定する必要 があります。



デュアルレールプロジェクト情報



デュアルレールプロジェクト情報

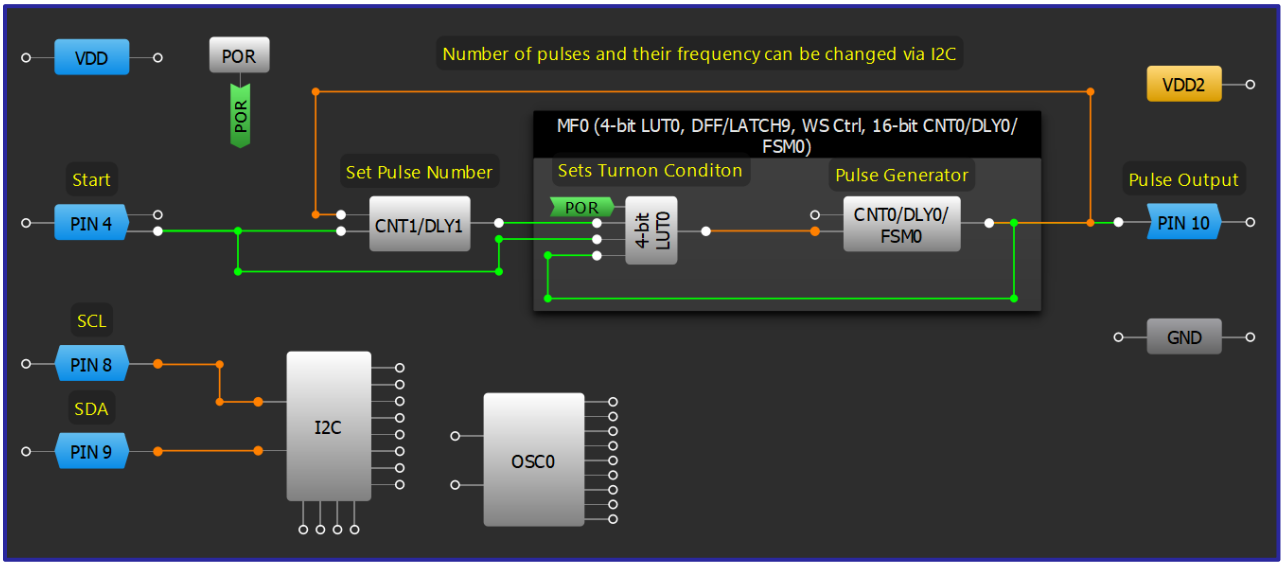
デュアルレールパーツで、第 1 レールおよび第 2 レール への GPIO 接続は GreenPAK Designer 内の IO PIN の色によって示されます。（図 22）。VDD は青いピン、 VDD2 は琥珀色のピンで示されます。GreenPAK マトリ クス内では、異なる電圧レベルの信号は同じ動作をしま す。

技法:プリセットされた数のパルスを送る

全ての GreenPAK.で適用可能です。いくつかの GreenPAK 内にある Multi Function ブロックを使うと要素数を削減できます。

多くの通信プロトコルでは、設定された数のビットを別の IC との間で送受信する必要があります。通常、これには、GreenPAK が送受信されるパルスの数を追跡する必要があります。たとえば、データを受信するシフト レジスタでは、ビット数を監視することにより、スキューによる誤りや絶え間ないシフトが続くことなく、期待されるデータが正しいレジスタにあることを保証する必要があります。

GreenPAK で所定のパルス数を設定するには、さまざまな方法があります。ここでは、拡張性が高く効率的な方法について説明します。この方法には、トランザクションごとにクロック スキューをリセットすることで、他の IC と GreenPAK 間のクロック スキューを制限するといった追加の利点もあります。下の図は、プリセットされた数のパルスを送信する一連のブロックを示しています。パルス数をセットするステージとパルスジェネレータのステージがあります。



プリセット パルス ジェネレータ設計

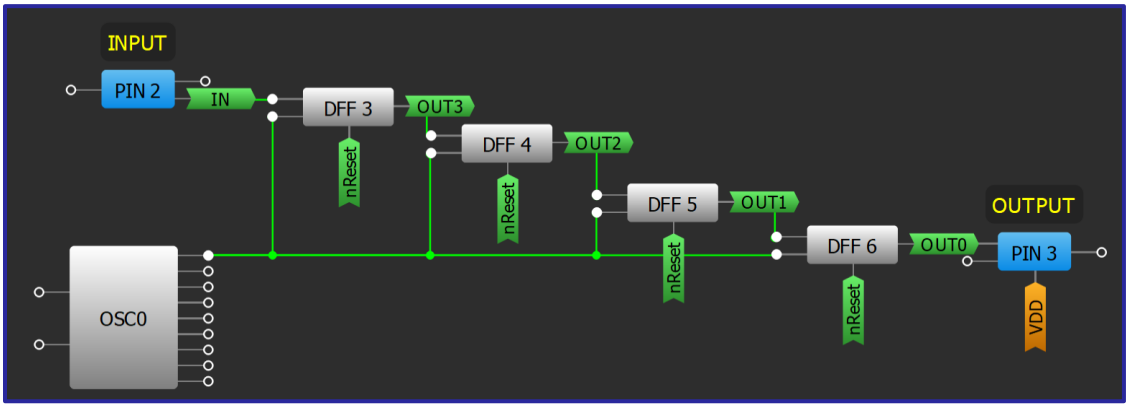
パルス数のステージは、パルスジェネレータの出力パルスをクロックとするワンショットブロックで構成されています。ピン 4 (開始) の立ち上がりエッジで、CNT1/DLY1 出力は、カウンタ データで設定されたパルス数だけクロックを受けるまで HIGH を維持します。設定したパルス数の後、LOW になります。

パルスジェネレータは MF0 でできています。 MF0 のうち、CNT0/DLY0 は反転出力の両エッジ遅延です。その遅延時間は、パルスジェネレータの周期を設定します。その出力は、CNT/DLY1 の出力が HIGH の間、CNT0/DLY0 からの信号を反転するためだけに構成された 4 ビット LUT0 にフィードバックされます。ワンショットパルスが終了すると、パルスジェネレータはパルスの送信を停止します。

技法：シフトレジスタの構築

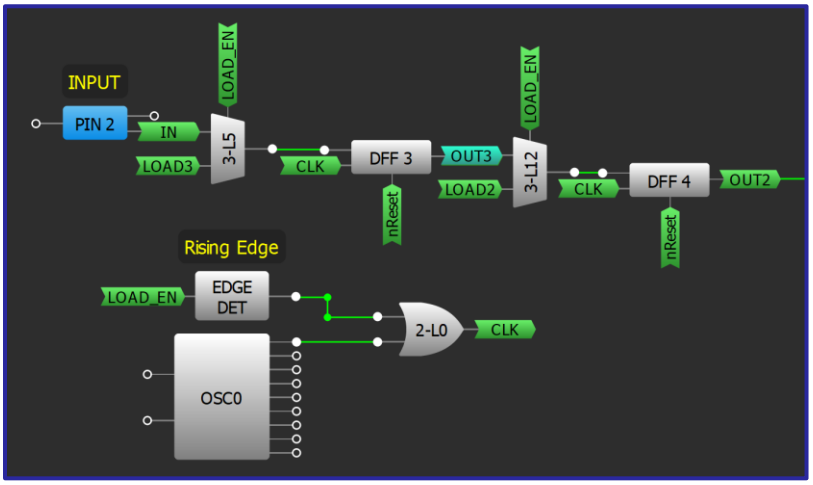
この技法は任意の GreenPAK 内で使用できます。シフトレジスタのサイズは特定の GreenPAK 内で利用可能なコンポーネン トによって異なります。

シフトレジスタはデータのシリアル化とデシリアル化のために不可欠なコンポーネントです。シフトレジスタはフリップフロップの チェーンで、順に結合したり、その出力に個別にアクセスしたりすることができます。それぞれが共有クロックへの接続を持ち、 クロックの立ち上がりエッジではレジスタによりデータがシーケンス内の次のフリップフロップにシフトされます。図 24 に基本的 な 4 ビットシフトレジスタを示します。D フリップフロップは共有リセット信号を使用してグローバルにリセットできます。



基本的なシフトレジスタ

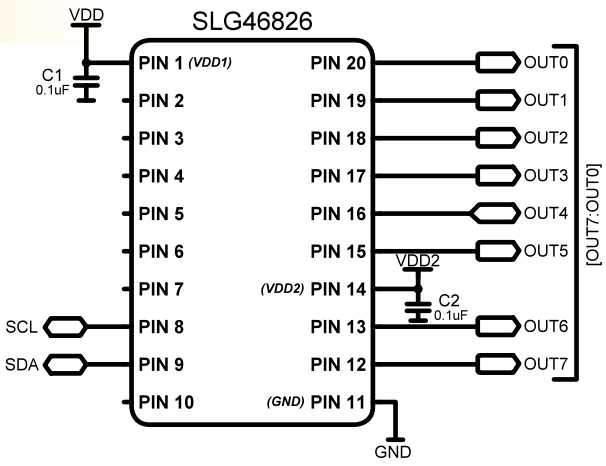
多くの場合、シフトレジスタを GreenPAK 内にロードする必要があります。そのために、各 DFF の前に MUX 標準ロジックセ ルを追加します。データのロードの準備が完了すると、各レジスタの値を確定するために MUX 選択入力（GreenPAK Designer では「S」）が切り替わり、DFF クロック入力が切り替わります。図 25 に、図 24 の基本的なシフトレジスタのビットの 2 に MUX を追加したものを示します。LOAD_EN は、ロードされた値を DFF3 および DFF4 に確定するための クロック 入力を共 有します。



シフトレジスタのロード

アプリケーション: I2C GPIO の拡張

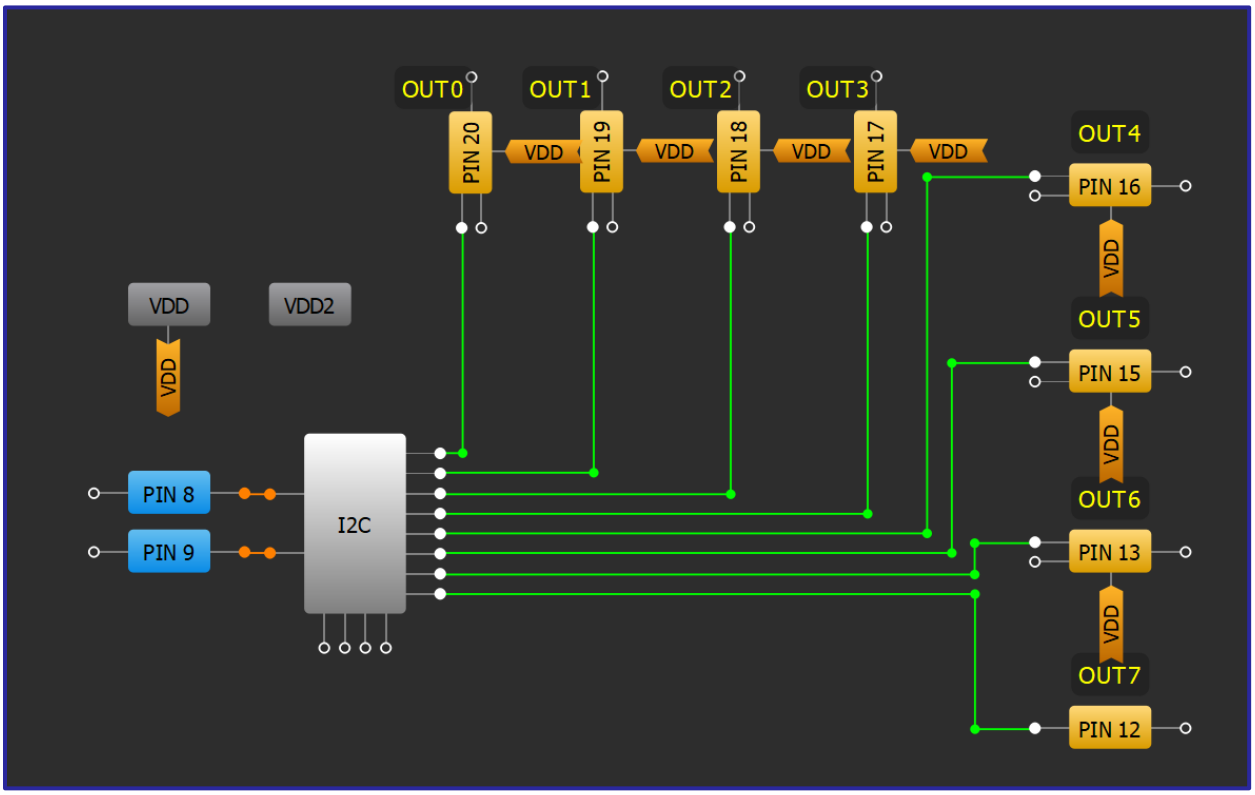
GPIO 拡張は、少数の信号線から発信されたデータを多数の信号線で送信されるデータに変更するために使用されます。1 つまたは複数のアドレスを、異なる IC で使用される複数の専用ラインに変更できるため、I2C はこのタイプのアプリケーションに共通な入力となります。



必要なコンポーネント

- I2C を有する任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

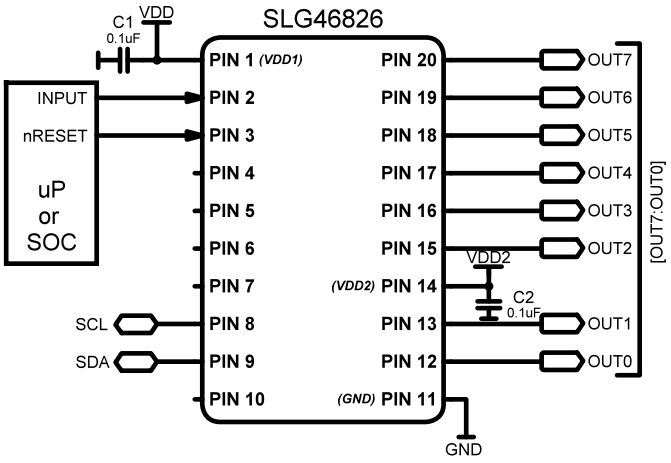


設計手順

1. GPIO ピンを出力として構成します
2. I2C 仮想入力に接続します
3. I2C仮想入力はGreenPAK のデータシートに従ってI2C仮想出力アドレスを使用して、個別または同時に変更することができます。

アプリケーション：シリアル/パラレル （外部クロック）

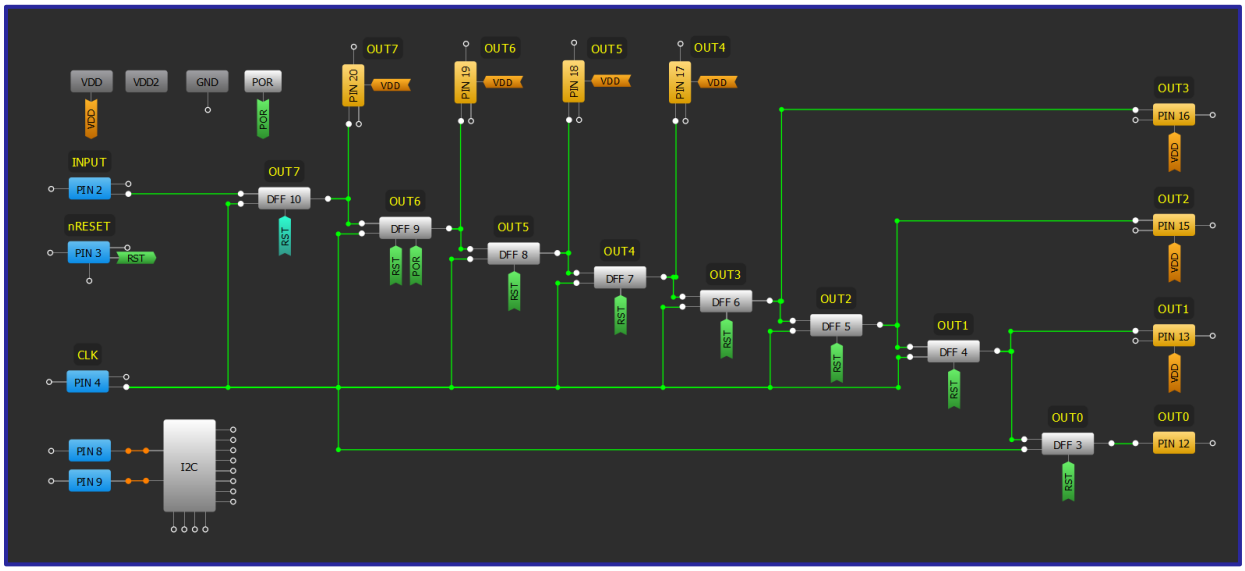
デシリアライゼーション IC はデータを 1 つのワイヤー上で 送信し、データを複数ビットとして読み取るた めに使用されます。データが不適切なタイミングでクロック 処理されるのを避けるため、同じデバイスにデータ通信ク ロックと同一の外部クロックを使用することもあります。



必要なコンポーネント

- 任意の GreenPAK
- IC と外部クロック出力

GreenPAK の図

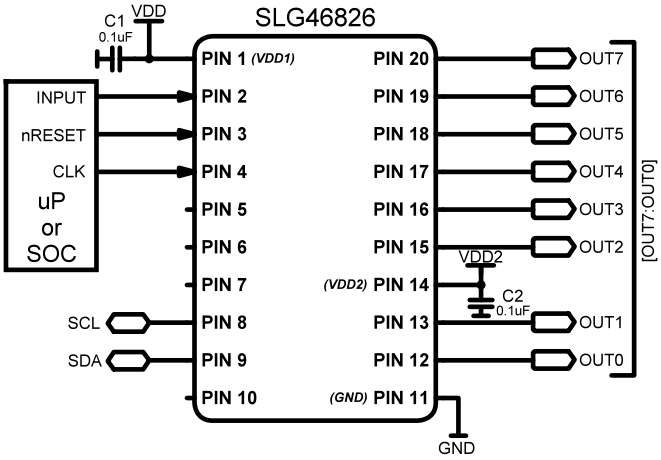


設計手順

1. 技法：シフトレジスタの構築に従ってシフトレジスタを構成します。
2. 内部クロック用の入力接続を追加し、シフトレジスタの CK 入力に接続します。
3. リセット機能用の入力接続を追加し、シフトレジスタの nRESET に接続します。

アプリケーション：シリアル/パラレル （内部クロック）

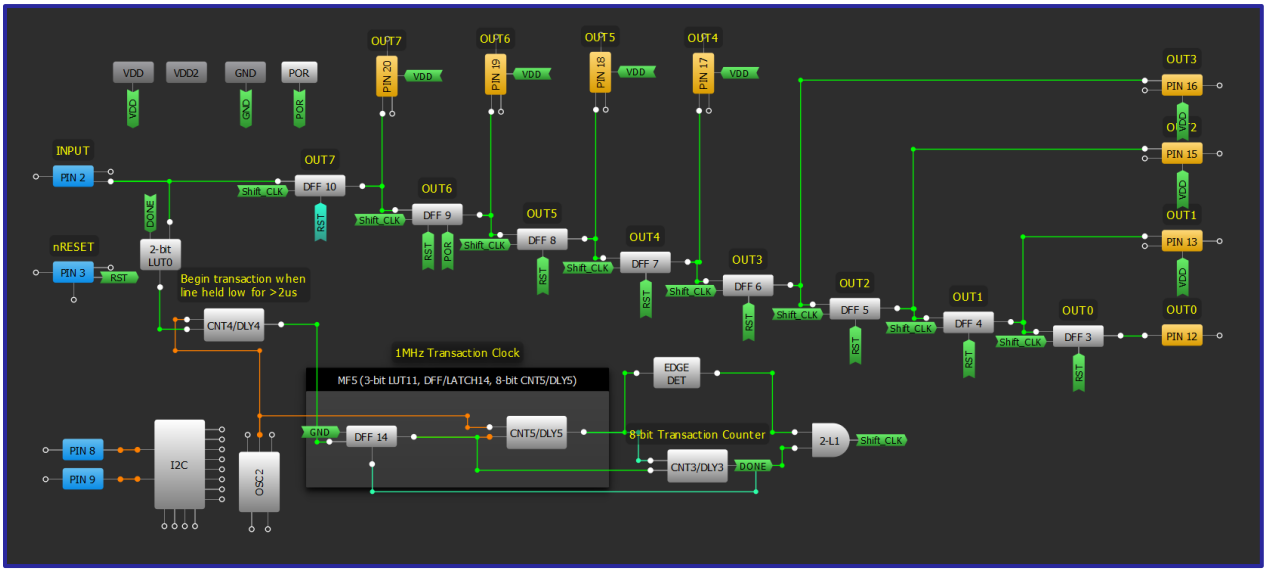
デシリアライゼーション IC はデータを 1 つのワイヤー上で送 信し、データを複数ビットとして読み取るために 使用されます。外部クロックの追加が行えない場合、事前に 規定された時間だけ保持される LOW 信号など、入力回線 でのアクションからトリガーされる内部オシレータを GreenPAK で使用することができます。



必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図



設計手順

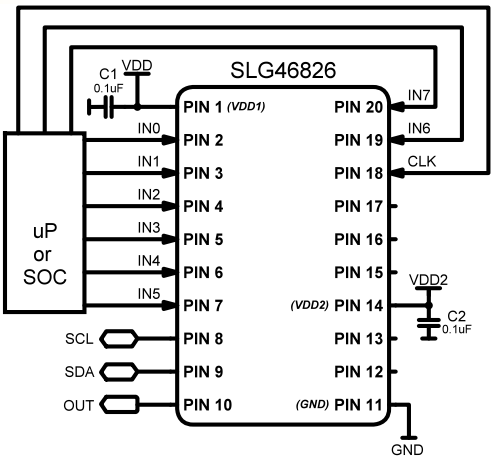
1. 技法：シフトレジスタの構築に従ってシフトレジスタを構成します。

2. リセット機能用の入力接続を追加し、シフトレジスタの nRESET に接続します。

3. シフトレジスタの段数に合わせて、プリセットパルス数を設定します。これについての概要は技法：プリセットしたパルス数 の送信を参照してください。

アプリケーション：パラレル/シリアル

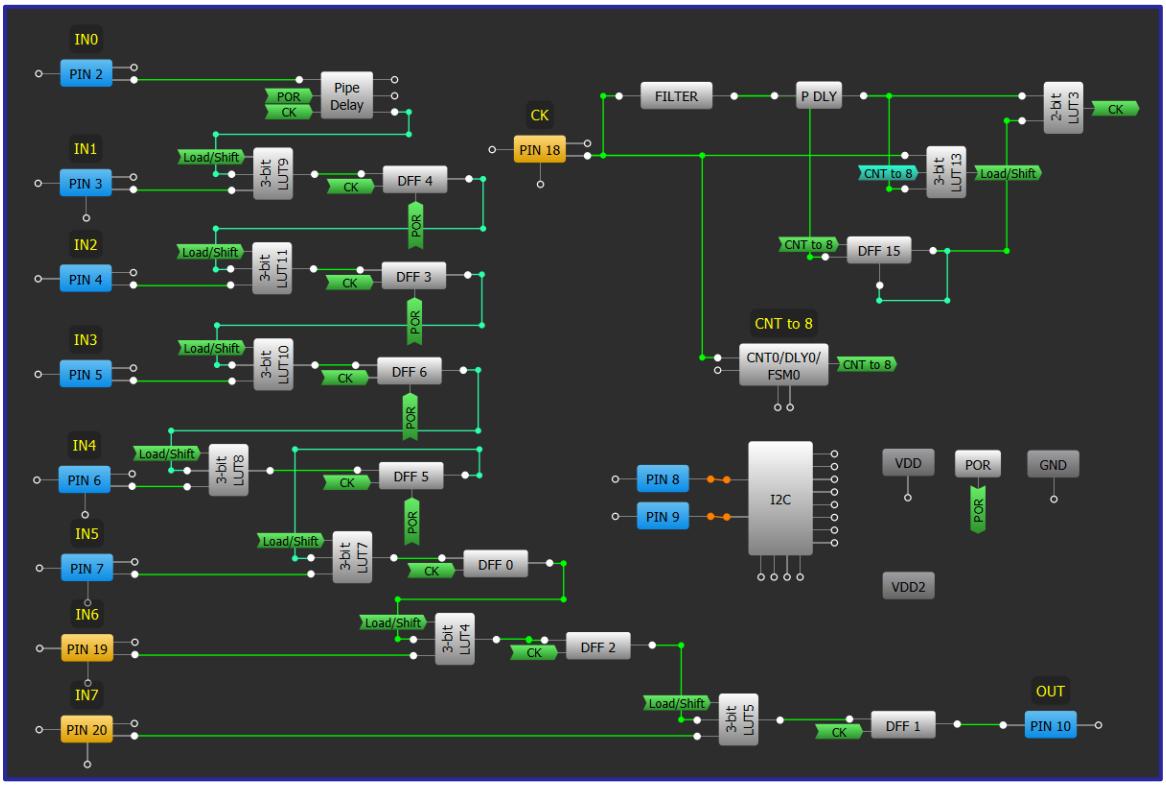
パラレル/シリアルコンバータは、一般的に 1 本または 2 本のワイヤーを使用 して 1 つの IC から別の IC にデータを送信するために使 用されます。ここでは内部または外部クロックを使用でき、パラレルビットの数は利用可能な GPAK I/O および内部ブロックの数によっ て制限されます。



必要なコンポーネント

- 任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

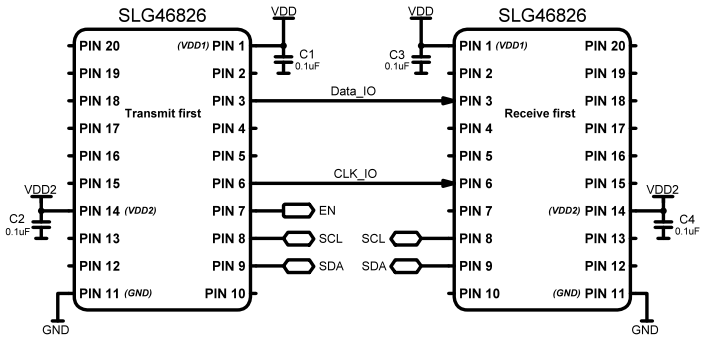


設計手順

1. 技法：シフトレジスタの構築に従ってシフトレジスタを構成します。
2. 各 DFF が MUX として動作するように、技法：LUT マクロセルを使用した標準ロジックの構成に従って 3 ビット LUT を追加し、構成します。
3. ロード/シフト機能を構成し、内部クロックを入力、出力に接続します

アプリケーション:双方向通信(送信優先)

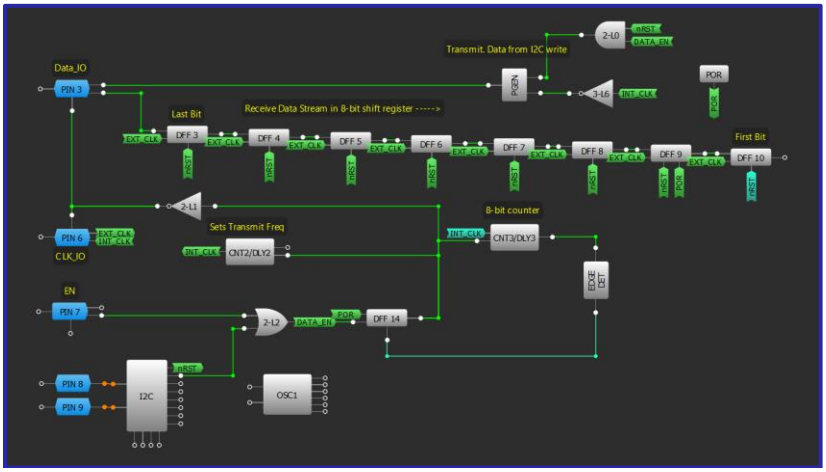
双方向通信システムは、基板面積が不足していたり、デバイス間の接続に使用される接点が限られているアプリケーション (ウェアラブル機器と充電器など) で重要です。送信優先設計は、このデバイスが常にトランザクションで最初に有効になることを示します。



必要なコンポーネント

- OE ピンを有する任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

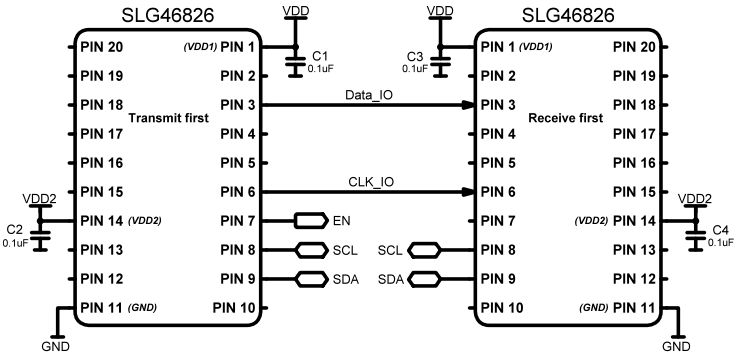


設計手順

1. 2 つの GPIO ピンを「デジタル入力/出力」として設定します。1 つはデータ ライン用で、もう 1 つはクロック用です
2. 「リセットカウンタ」として構成された CNT/DLY ブロックを使用して、内部クロック信号を作成します
3. 「技法: シフト レジスタの構築」に示すように、入力データを格納するようにシフト レジスタを構成します。シフト レジスタは I2C で読み取ることができます
4. PGEN を追加して外部送信データを送信します。 PGEN の内容は、I2C を使用して変更できます
5. DFF を構成して送信を有効にします。出力を送信クロック信号と I/O ピンの OE に接続します
6. CNT/DLY ブロックを追加して、DFF からの CLK 入力で送信を無効にします。 Edge Det ブロックを接続して、設定された数のクロック パルスの後に DFF をリセットします
7. 外部クロックをシフト レジスタに接続し、内部クロックを PGEN および I/O ピンに接続します

アプリケーション:双方向通信(受信優先)

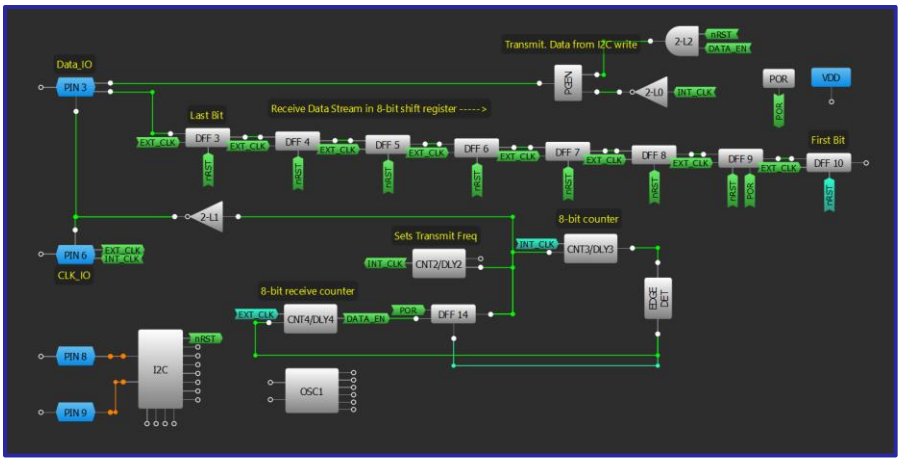
双方向通信システムは、基板面積が不足したり、デバイス間の接続に使用される接点が限られているアプリケーション (ウェアラブルと充電器など) で重要です。受信優先設計は、このデバイスが常にトランザクションで 2 番目に有効になることを示します。



必要なコンポーネント

- OE ピンを有する任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図

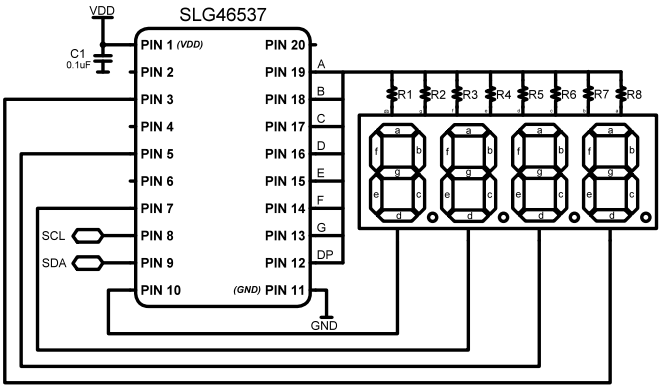


設計手順

1. 2 つの GPIO ピンを「デジタル入力/出力」として設定します。1 つはデータ ライン用で、もう 1 つはクロック用です
2. 「リセットカウンタ」として構成された CNT/DLY ブロックを使用して、内部クロック信号を作成します
3. 「技法: シフト レジスタの構築」に示すように、入力データを格納するようにシフト レジスタを構成します。シフト レジスタは I2C で読み取ることができます
4. PGEN を追加して外部送信データを送信します。 PGEN の内容は、I2C を使用して変更できます
5. DFF を構成して送信を有効にします。出力を送信クロック信号と I/O ピンの OE に接続します
6. CNT/DLY ブロックを「リセットカウンタ」として追加して、DFF からの CLK 入力で送信を無効にします。 Edge Det ブロックを接続して、設定された数のクロック パルスの後に DFF をリセットします
7. 外部クロックをシフト レジスタに接続し、内部クロックを PGEN および I/O ピンに接続します

アプリケーション: ASM と I2C を用いた 7 セグメント表示

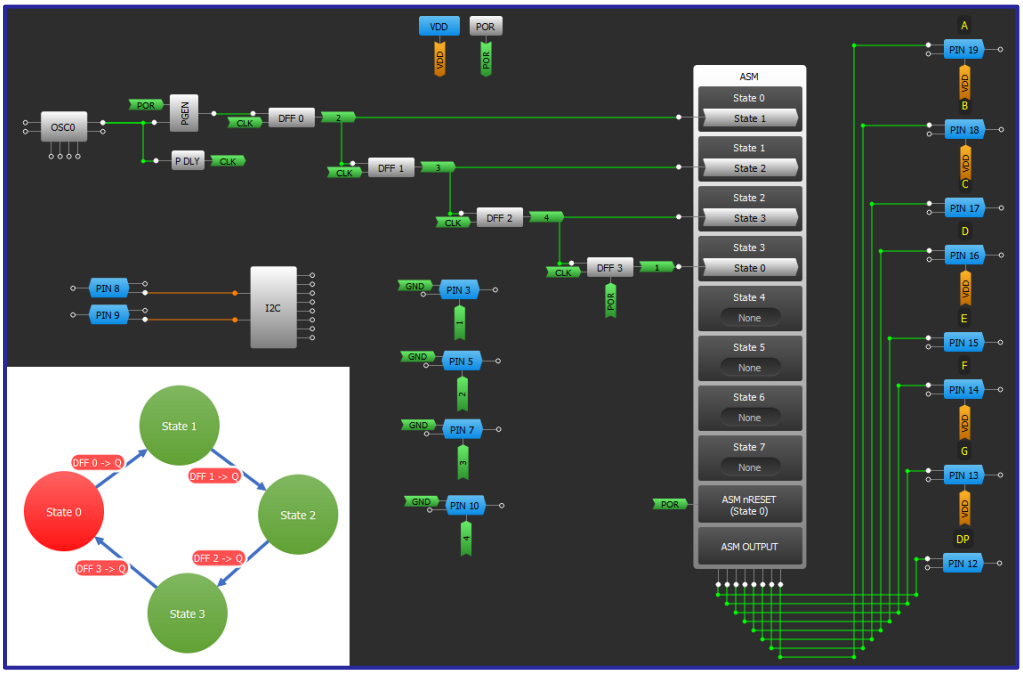
7 セグメント インジケータは、一般的な数値を表示します。 GreenPAK 非同期ステートマシンと I2C を使用して、表示する番号をセグメントに指示できます。提供する例は、4 桁の十進表示と互換性があります。



必要なコンポーネント

- I2C と ASM がある任意の GreenPAK
- 7 セグメント表示器
- 8 つの抵抗

GreenPAK の図

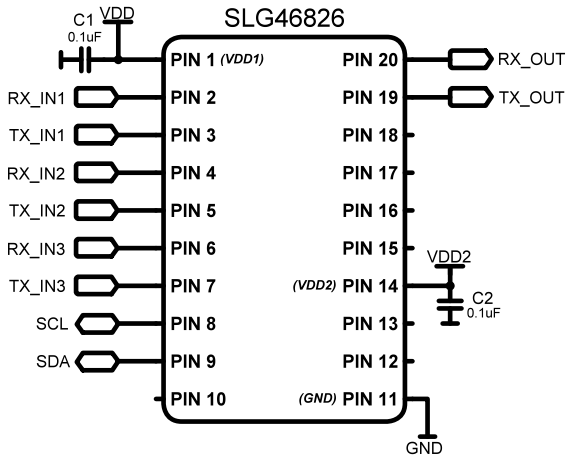


設計手順

1. GPIO ピンを出力として設定し、ASM 出力に接続します
2. 「技法: シフト レジスタの構築」を使用してシフト レジスタを追加します
3. 必要な桁数に応じて、 PGEN ブロックを使用してロジックジェネレータを作成します
4. 最初のデジタル シーケンスと一致するように ASM を追加して構成します
5. I2C を介して 1 つまたは複数の桁を ASM の状態に更新します

アプリケーション: I2C を用いた通信 MUX

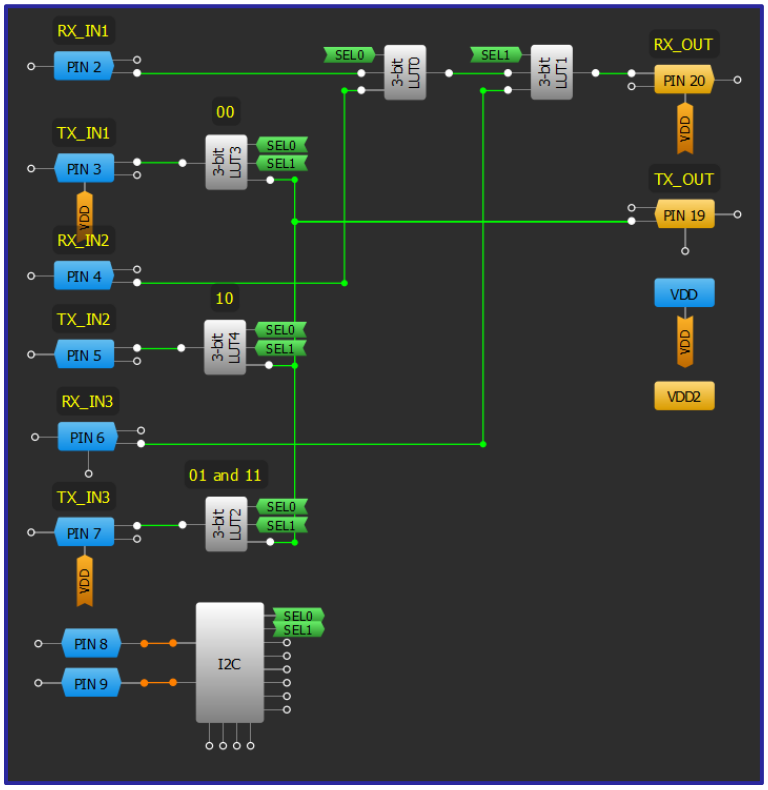
I2C Communication MUX は、設計者が複数の I2C 入力信号を組み合わせて単一の出力ラインに転送する必要がある場合に使用します。



必要なコンポーネント

- I2C を有する任意の GreenPAK
- 2 つの抵抗
- 2 つの容量

GreenPAK の図

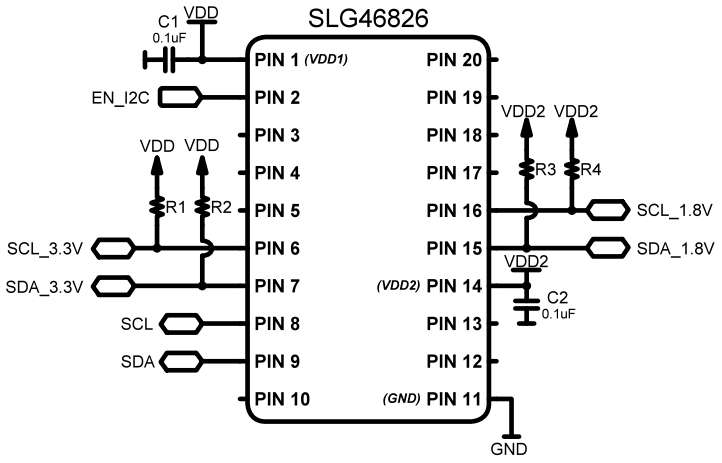


設計手順

1. LUT を使用して MCU からの TX 信号を共有するために 1:3 demux を構成し、外部 UART ポートに送信します
2. LUT を使用して 3:1 マルチプレクサを構成し、外部 UART ポートからの MCU への RX 信号を受信します
3. I2C を使用して入力ポートを選択します。

アプリケーション: I2C レベルシフター

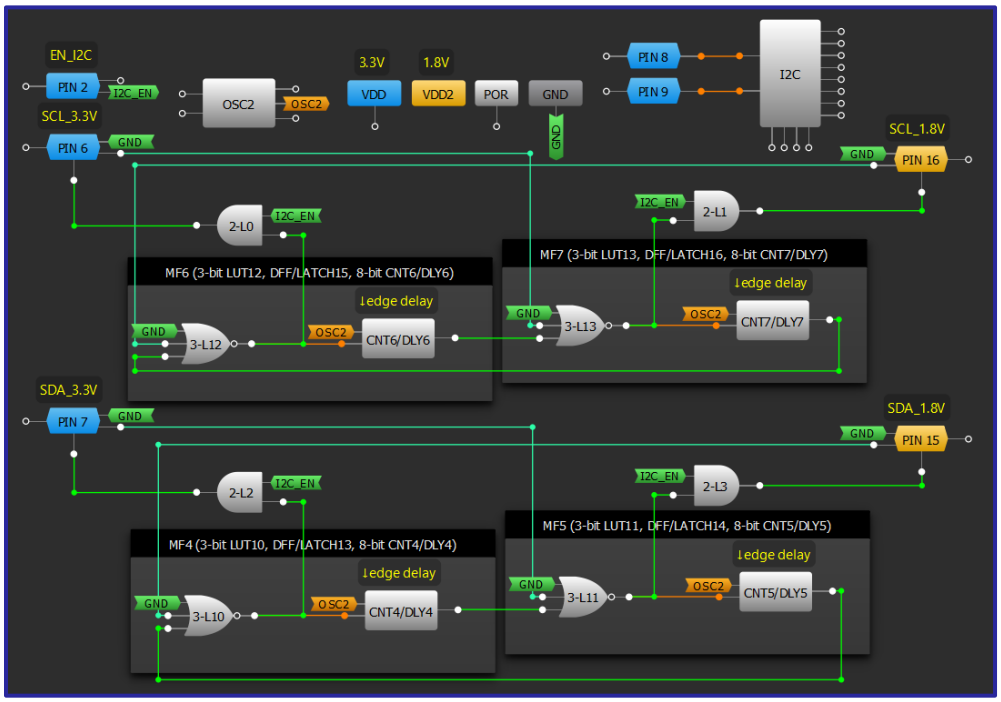
I2C レベル シフターは、2 つの I2C 対応デバイスが 2 つの異なる電圧レベルで相互に通信できるようにします。与えられた例では、レベルが 3.3V から 1.8V にシフトします。



必要なコンポーネント

- 任意の GreenPAK
- 4 つの抵抗

GreenPAK の図

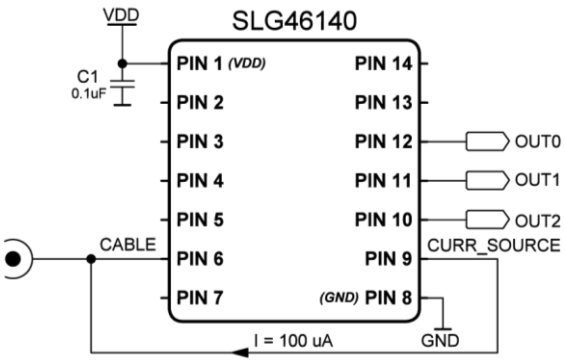


設計手順

1. 4 つの GPIO ピンをデジタル入力/出力として設定し、出力モードをオープンドレインの NMOS に設定します
2. 1 つのピンをイネーブル信号のデジタル入力として設定します
3. 各入力/出力に AND ゲートを追加します
4. 4 つの多機能ブロック (多機能ブロックが利用できない場合は 4 つの LUT と 4 つの CNT/DLY ブロック) を立ち下がりエッジ遅延に供給する NOR ゲートとして構成します
5. OSC2 を遅延ブロックのクロックソースとして選択し、「Force Power On」に設定します

アプリケーション:接続検出

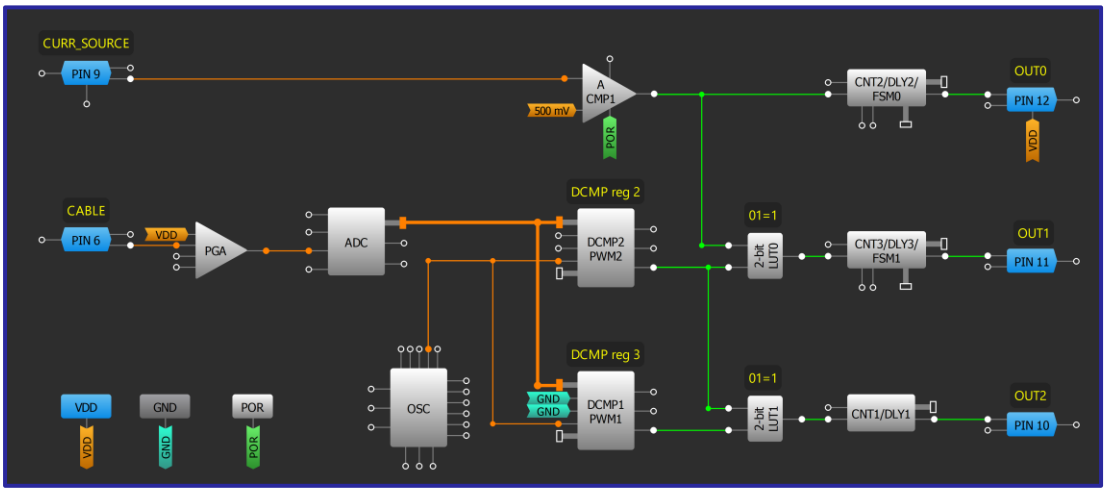
このアプリケーションは、ケーブルの抵抗に比例する電圧を測定することにより、ケーブルの存在を検出します。 ACMP の 100uA 電流源は、ケーブルの電圧降下を生成するために使用されます。この構成では、さまざまなワイヤの長さや接続された負荷に基づいて、どのタイプの接続が行われているかを判断することもできます。



必要なコンポーネント

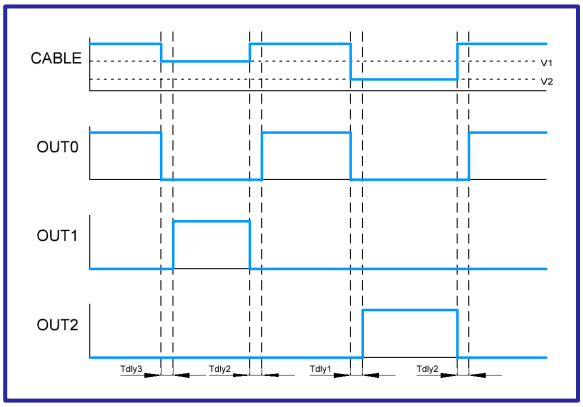
- ADC を有する GreenPAK

GreenPAK の図



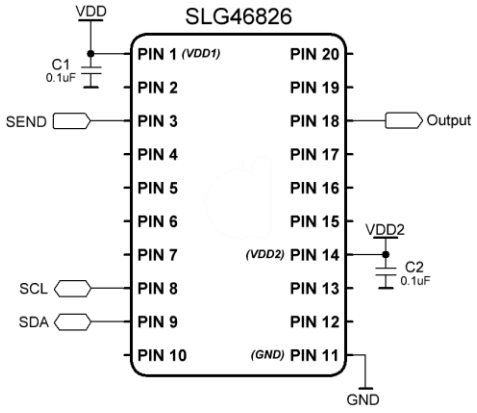
設計手順

1. ACMP1 で入力 100uA 電流源を有効にし、IN-ソースを構成します
2. ADC および PGA ブロックを構成します
3. DCMP/PWM ブロックを有効にします (SHARED PD 入力から VDD を削除します)。 DCMP/PWM パワー レジスタを Power on に設定します。IN+セレクタが ADC に接続され、IN-セレクタが内部レジスタに接続されていることを確認ください
4. DCMP/PWM ブロックのレジスタに必要な値を設定し、MTRX SEL 入力を構成します
5. 2 ビット LUT0 と 2 ビット LUT1 を構成します
6. CNT1-CNT3 を立ち上がりエッジ遅延モードに設定します
7. PIN10-PIN12 をデジタル出力 1x プッシュ プルとして構成します



アプリケーション: カスタムパターン ジェネレータ

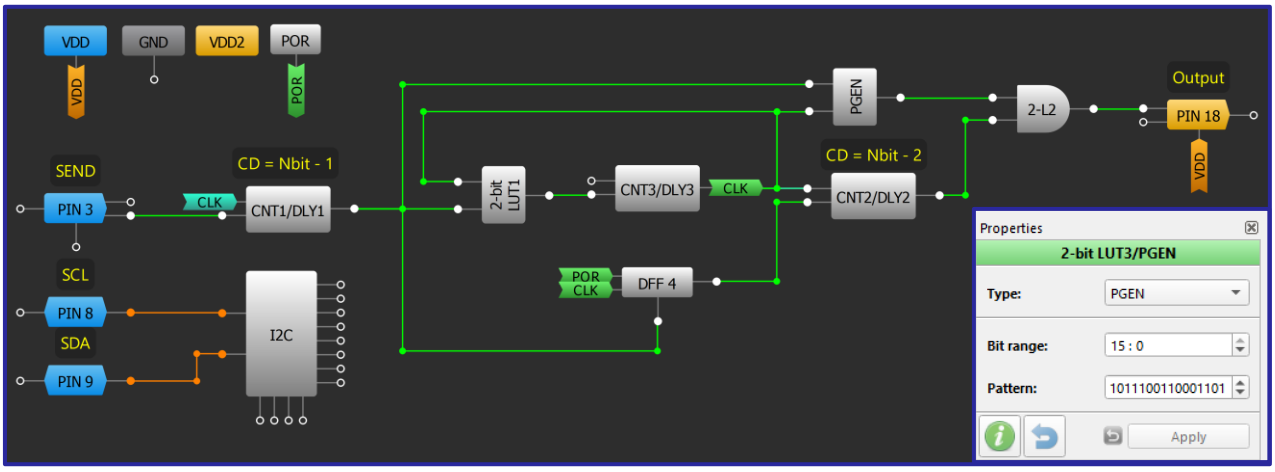
GreenPAK のパターン ジェネレータ (PGEN) は、論理 1 と 0 のパターン (最大 16 ビット) を格納し、内部マトリックスにシリアル送信できます。 このデザインは、入力の立ち上がりエッジの後に 16 ビット コードを出力します。 シーケンシャル ロジック アプリケーションで使用できます。



必要なコンポーネント

- PGEN を有する任意の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図



設計手順

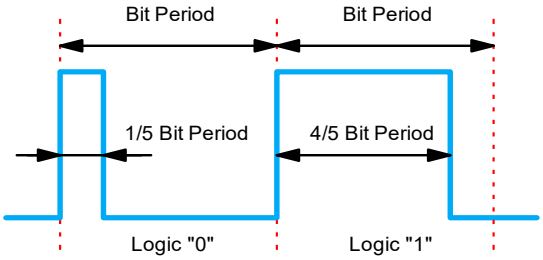
1. PGEN でカスタム N_bits パターンを構成します
2. CNT1/DLY1 モードをワンショットとして構成します。Counter Data を Counter Data=N_bits-1 に設定します
3. CNT2/DLY2 モードをワンショットとして設定し、Counter Data を Counter Data=N_bits-2 に設定します
4. CNT3/DLY3 を立ち上がりエッジ遅延として構成し、2 ビット LUT1 を使用してジェネレータを作成します
5. この設計は、I2C で GreenPAK を使用して PGEN データ、クロック周波数、およびカウンタ データを動的に変更することで改善できます。

技法: デューティーサイクル検出を使用したシリアルプロトコルの送信

この技法は、PGEN、カウンタ/遅延ブロック、および I2C で構成される任意の GreenPAK で使用できます。ここで説明されている GreenPAK を用いてプロトコルを読み取る方法については、「技法:シフトレジスタによるシリアルプロトコルの読み出し」と「技法：パイプ遅延を用いたシリアルプロトコルの読み出し」をお読みください。

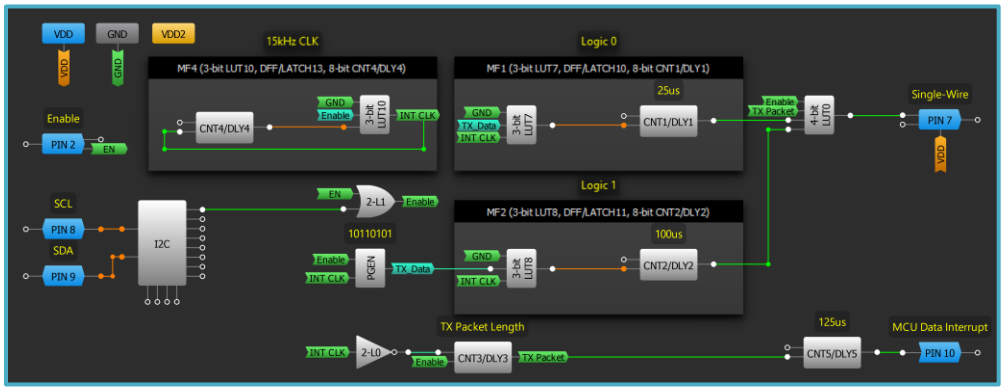
シリアルデータ伝送には、電力線、無線システム、MCUなどを含むさまざまな通信アプリケーションがあります。単線データ伝送は、別の GreenPAK または MCU が読み取るための個別のデューティ サイクルを使用してデータを転送できます。

デューティサイクル検出トポロジを下の図に示します。各ビットは、特定のデューティサイクル範囲を持つ周期的なパルスによって表されます。この設計では、パルス持続時間が周期の 1/5 以下の場合、処理されたビットは「0」に設定され、周期の 4/5 以上の場合は「1」に設定されます。ロジック「0」とロジック「1」のデューティサイクル範囲の間に十分な距離がある場合は、他のデューティサイクル範囲を使用できます。



Logic 0 と Logic 1 の検出

以下に示すデザインでは、データ送信は Enable 信号で開始されます。これは、外部 GPIO または I2C の仮想入力を介してサポートされます。 CNT4/DLY4 は、データ送信のビット周期を設定します。 MF1 と MF2 ブロックは、送信するデューティサイクルを決定します。送信するデータは、I2C または不揮発性メモリを介して PGEN に書き込まれます。PGEN は最大 16 ビットを送信できますが、この設計では単線出力 8 ビットのデータ送信を示します。 I2C 経由で PGEN を介してさまざまなパターンを書き込む方法の詳細については、「アプリケーション: カスタム パターン ジェネレータ」を参照ください。



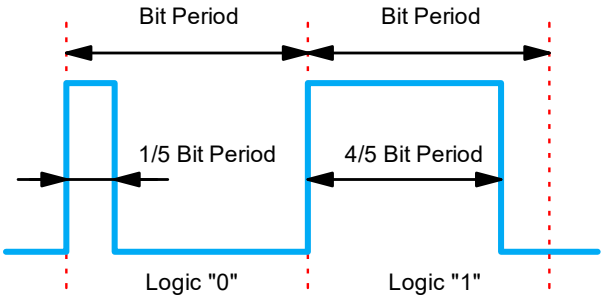
データ パケット長は CNT3/DLY3 によって決定され、この場合は 8 ビットを送信するように設定されています。 4 ビット LUT0 は、データが送信信号のデューティサイクルに組み込まれた後、シングルワイヤ出力でデータを送信します。データ送信が完了したら、I2C を介して PGEN を再書き込みできます。

ユーザーが送信を終わりにしたい場合、CNT5/DLY5 は割り込み信号を生成して、送信されたパッケージが完了したことを外部 MCU に示します。

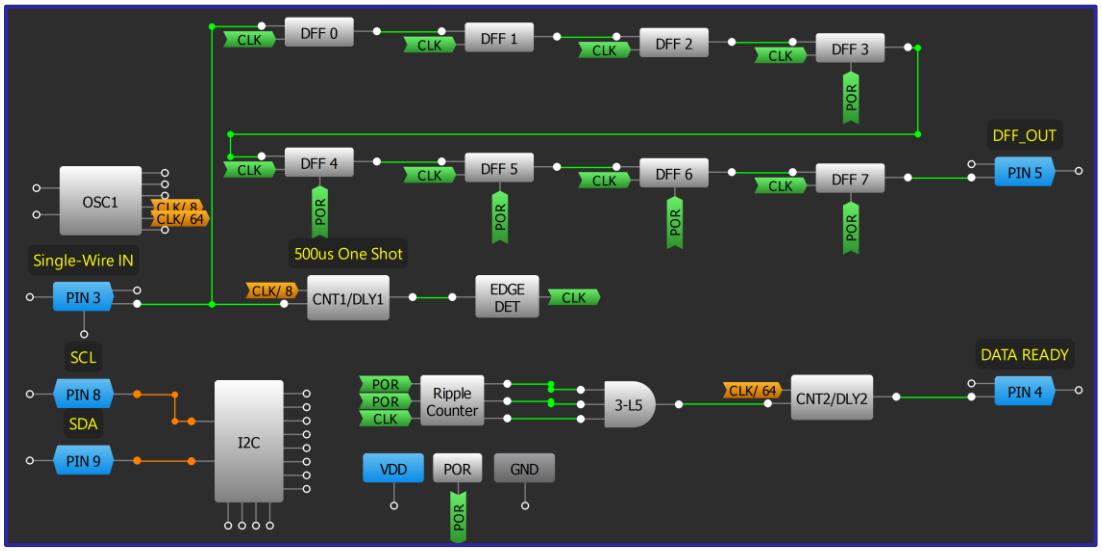
技法: シフトレジスタによるシリアルプロトコルの読み出し

この技法は、十分な数の DFF ブロック、1 つのリップル カウンタ、および 3 つの CNT/DLY ブロックを備えた任意の GreenPAK 内で使用できます。「技法:デューティサイクル検出を使用したシリアルプロトコルの送信」で説明されている単線プロトコルを読み取る 1 つの方法について説明します。

シリアル データ伝送には、電力線、無線システム、MCU など、さまざまな通信アプリケーションがあります。シフト レジスタは、単線伝送用の汎用性の高いシリアル レシーバとして使用できます。



上の図は、「技法: デューティサイクル検出を使用したシリアルプロトコルの送信」で説明した単線トポロジーを示しています。立ち上がりエッジでの遅延エッジ検出は、信号周期 (ビット長) の途中でサンプリングするために使用されます。この特定の設計のシフト レジスタがクロックされると、パルスがビット周期の 1/5 以下の部分で信号が LOW になり、パルスがビット周期の 4/5 以上の部分で HIGH になります。



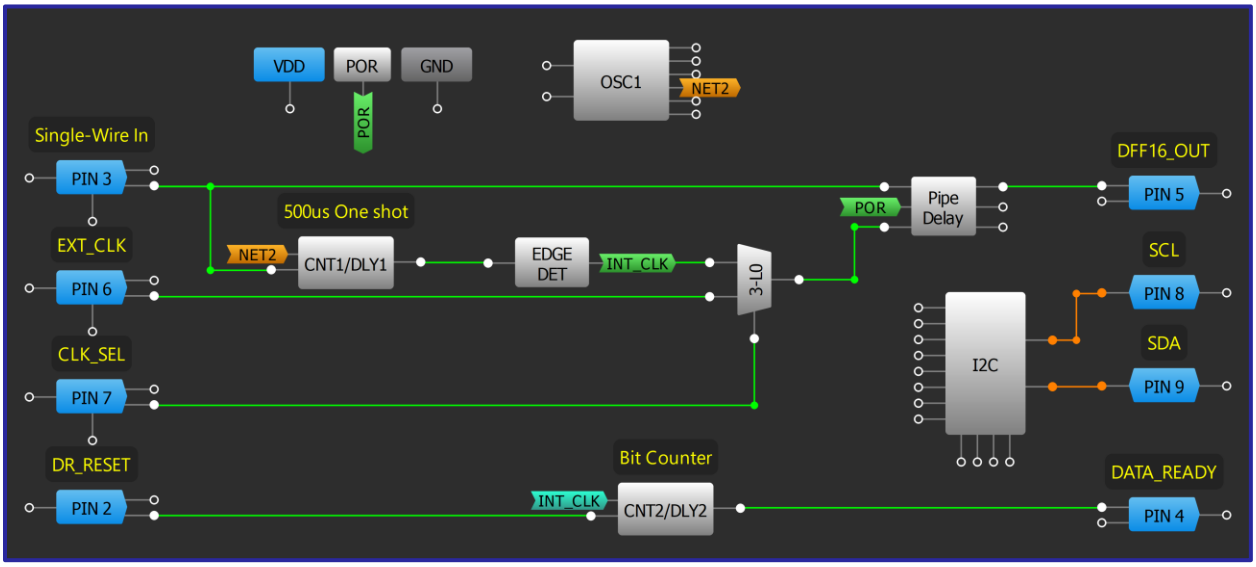
上図は、8 ビットの増加の単線データを取得する 1 つの方法を示しています。シフトレジスタがクロックされるたびに、リップルカウンタもインクリメントされます。リップル カウンタが 8 つのパルスを受信すると、CNT2/DLY2 は DATA READY にハイパルスを出力して、単線伝送が完全に読み取られ、I2C による読み取りの準備が整ったことを MCU に通知します。この設計では、CNT1/DLY1 が立ち上がりエッジを 500µs 遅らせるため、1ms 周期で送信を読み取りますが、これはカウンタ値を調整することで簡単に変更できます。

技法:パイプ遅延によるシリアルプロトコルの呼び出し

この技法は、任意の GreenPAK 内で使用できます。「技法: デューティ サイクル検出を使用したシリアル プロトコルの送信」で説明されている単線プロトコルを読み取る方法を説明します。

シリアルデータ伝送には、電力線、無線システム、MCU を含めて、さまざまな通信アプリケーションがあります。シングルワイヤ伝送用のシリアルレシーバとして、リソースを要するシフト レジスタの代わりに単一の Pipe Delay ブロックを使用できます。ただし、シフトレジスタ方式とは異なり、このパイプ ディレイ方式はデータをシリアルにしか出力できず、パラレル出力機能がないことに注意する必要があります。

パイプ遅延は、3 つの利用可能な出力を持つ 16 ビットのシフト レジスタと考えることができます。 2 つは、Pipe Delay の 16 個の内部 D フリップフロップのいずれかを出力するように構成可能です。以下に示すこの 16 ビットの実装では、パイプ遅延の OUT0 は 16 番目の DFF 出力を表すように設定されています。MCU が外部クロックでパイプ遅延を動かすための入力は、 Single-Wire In 入力と切替られるようになっており、ビット カウンタをインクリメントせずにパイプ遅延のデータをアンロードできます。

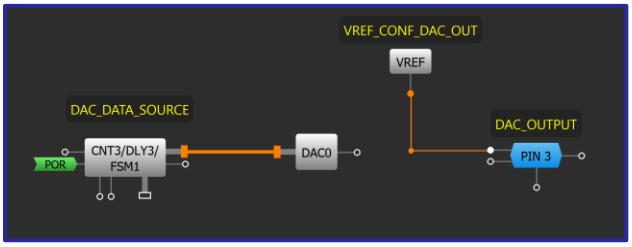


この設計は、「技法: シフトレジスタによるシリアル プロトコルの読み出し」と同じ遅延データ サンプリング メカニズムを使用しますが、リップルカウンタではなくカウンタブロックを使用してデータ ビットを保持します。このカウンタは、データ ビットが検出されるたびに増加します。 16 ビットに達すると、次のデータ ビットが送信されるまで DATA_READY ピンを HIGH に設定します。適切な動作を確保するために、データが送信される前にこのカウンタブロックをリセットする必要があることに注意することが重要です。これは、POR によって内部的に、または入力によって外部的に行うことができます (このデザインのように)。

技法: D/A コンバータ (DAC)の使用

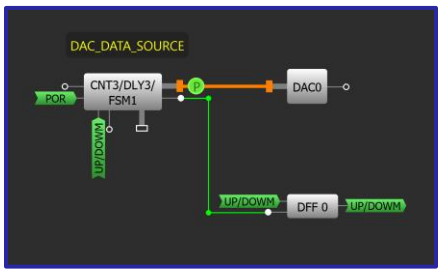
この技法は SLG46140、SLG46620、および、SLG46621 の GreenPAK で使用可能です。.

一部の GreenPAK デバイスには、デジタル-アナログ コンバータ (DAC) が含まれています。これらは、100 ksps の最大サンプリング速度で動作する 8 ビット DAC です。 DAC の微分非直線性は 1LSB 未満で、積分非直線性は 1LSB 未満です。 DAC 出力から 端子 への抵抗は 1 kΩ です。負荷抵抗は 10kΩ 以上、負荷容量は 100pF 以下を推奨します。通常、DAC 出力範囲は 0V ～ 1V ですが、SLG46620/1 では DAC1 出力範囲は 50mV ～ 1.05V です。
レジスタ、SPI、または FSM のいずれかを DAC の入力として構成できます。 DAC の出力は、VREF の出力ピン、PGA、または ACMP に設定できます。
一部の IC では、DAC0 は PGA マクロセルの疑似差動モードの一部として使用されます。したがって、PGA が疑似差動モードの場合、DAC0 は使用できません。また、DAC1 は ADC マクロセルと共有されます。したがって、ADC を使用する場合、DAC1 を使用することはできません。 DAC マクロセルの出力を VREF マクロセルに接続するには、このピンをアナログ入力/出力として構成し、VREF のソース セレクタを DAC に構成する必要があります。



鋸歯状波形生成器

DAC を使用して、上に示す単純な鋸歯状(Sawtooth)ジェネレータを作成できます。この場合、DAC0 は FSM1 をデータソースとして使用します。 DAC 出力は、PIN3 (DAC_OUTPUT) に接続された VREF に接続されます。出力信号の周期と分解能は、FSM1 のカウンタデータとクロック周波数によって設定されます。また、トグル DFF を追加し、それを FSM の UP 入力に接続して、以下に示す三角波発生器を構成することもできます。カウンタデータ値は、DFF0 の出力に応じて、時間とともに上下に変化します。



三角波生成器

DAC は、「差動」および「疑似差動」モードで ACMP または PGA のネガティブ入力のリファレンスとして使用できます。ACMP リファレンスが DAC を使用する場合、ユーザーは ADC ブロックを使用せずにアナログ信号をディスクリート データと比較できます。

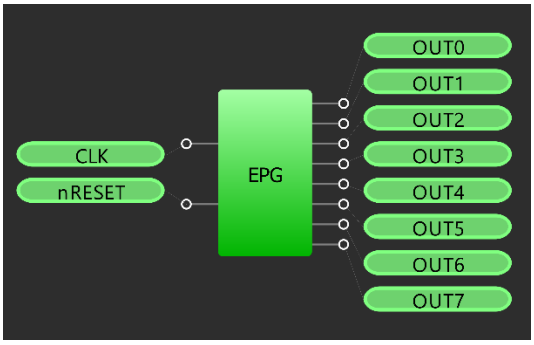
GreenPAK の DAC は

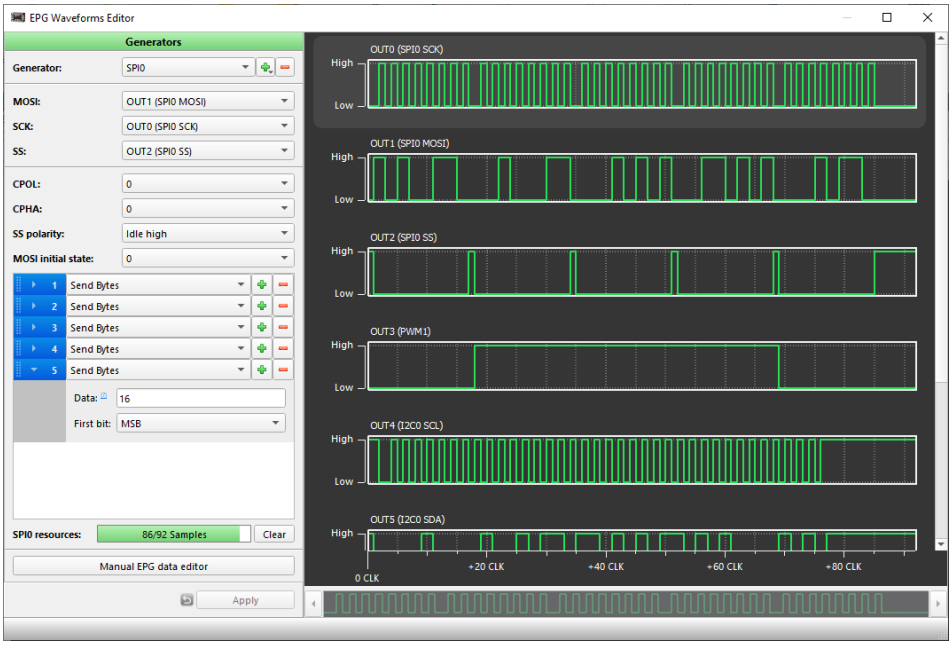
- 波形生成器として使用することができます
- SPI に制御される基準電圧源を生成できます
- 温度、湿度、および、その他のデジタル値をアナログ電圧に変換することで、さまざまなコンバータで使用できます。

技法: EPG

拡張パターンジェネレータ（EPG）は 92 バイトの領域で出力を生成できます。
不揮発性メモリ（NVM）からデータを取得後、入力 CLK 信号が立ち上がるたびに 1 バイトずつ出力します。さらに、EPG はその出力を I2C の仮想入力と共有します。クロックの最大周波数は 1MHz です。
システムの電源投入後、EPG は nReset から入力される信号に応じて異なる動作をします。nReset にアクティブ Low が入力された場合、EPG は初期値を出力します。反対に nReset にアクティブ High が入力された場合、EPG はユーザが定義したパターンを出力します。この機能により、ユーザは EPG の出力をカスタマイズすることができます。
EPG は、オーバーフローモードで CLK に信号が印加されている間動作し続けます。また、バウンダリーモードでは最後のバイトに到達したとき動作を停止します。

EPG 波形エディタでは、SPI、I2C、PWM、マニュアルなど様々な定義済みのジェネレータから選択し、それに応じて出力を割り当てることができます。これらのジェネレータはそれぞれ調整可能な設定範囲を備えており、ユーザが希望するパターンを簡単に作成できます。

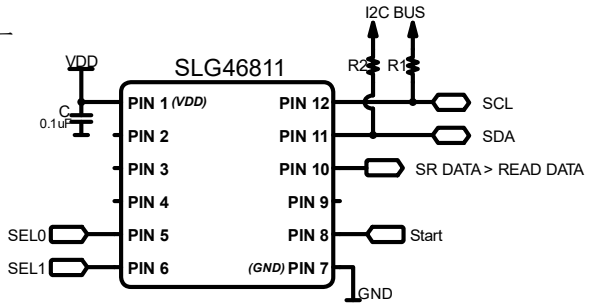




リソースバーはパターンで使用されているビット数を視覚的に示してくれます。これによりユーザーは、現在のパターンに割り当てられているメモリとリソースの量を明確に理解できます。

アプリケーション: ACK チェックとデータ比較を備えた I2C マスタ側 Read コマンド

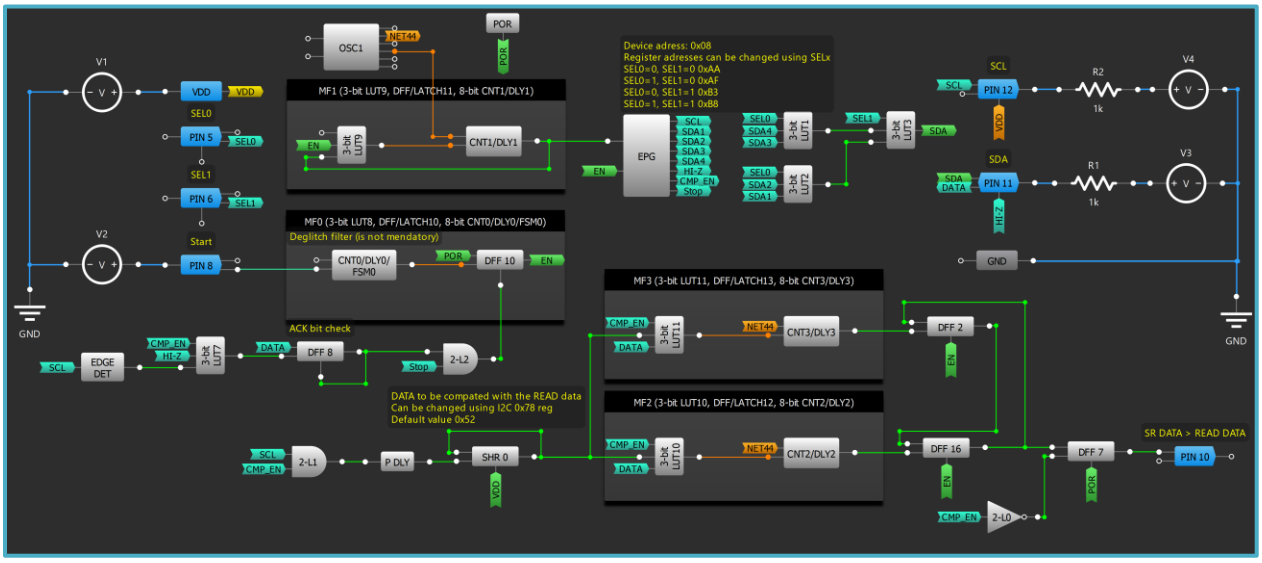
このアプリケーションでは、SLG46811 を使用して、I2C スレーブデータを Read、参照データと比較することができるシンプルな I2C マスタを構築する方法を示します。



必要なコンポーネント

- SLG46811V
- 抵抗　2 個

GreenPAK の図

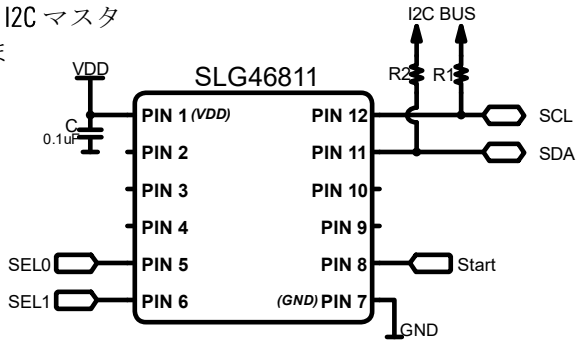


設計手順

1. EPG Generator (Technique：EPG を参照)を設定して、I2C Read コマンドを作成し、いくつかの SDA および SCL パターンを設定します。
2. EPG に 1 つのチャネルを追加して、SCL 9 クロックごとに 受信したデータ(ACK ビット)をチェックする回路を作ります。
3. Application: 8-bit Multiplexer で説明されているように、４ビットのマルチプレクサを構成します。SEL0，SEL1 でどのデータ(SDA)を出力するか選択します。
4. CNT1/DLY1 と LUT9 を使用して、EPG のクロック源とします。
5. I2C パターンを開始する信号のデグリッジフィルタを CNT0/DLY0 で構成します。
6. このデザインは I2C コマンド中に ACK ビット チェックが存在するかどうかを監視し、I2C スレーブが応答しない場合、送信を停止します。
7. I2C スレーブから受信したデータは SHR 0 に保存されている参照データと比較されます。比較中の各クロックで、SHR0 のデータがシフトされ、受信したデータと比較されます。DFF7 は、読み取りコマンドまで比較結果をストアします。

アプリケーション: ACK チェックを備えた I2C Master Write Command 機能

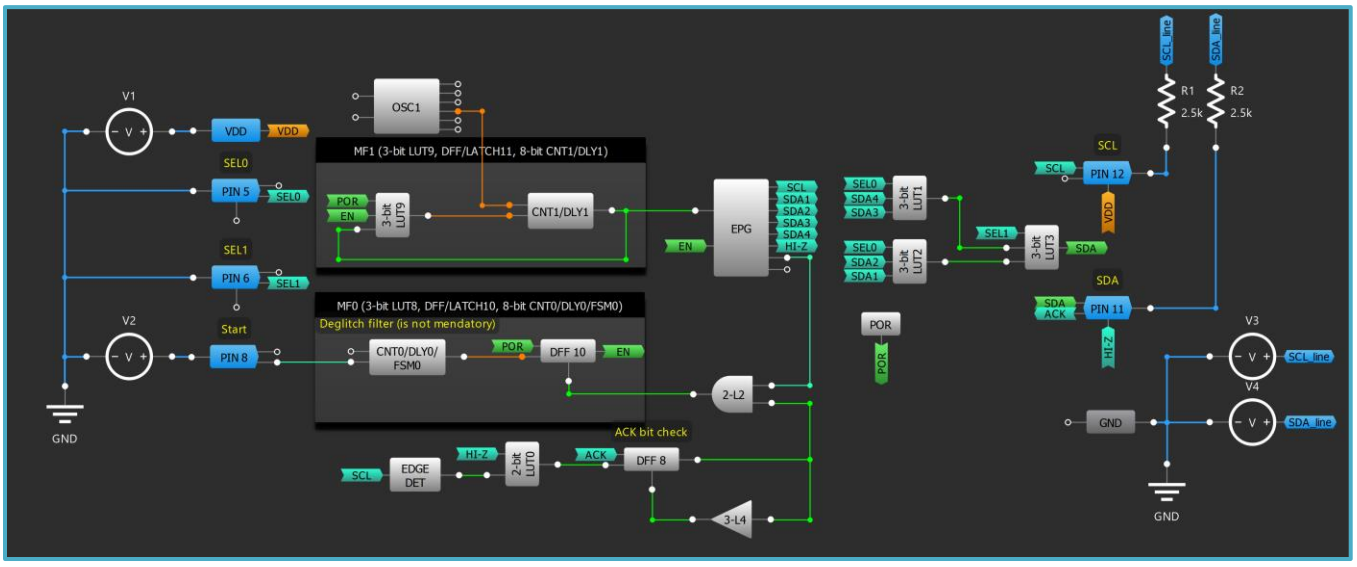
このアプリケーションでは、SLG46811 デバイスを使用して簡単な I2C マスタを構築する方法を示します。構築された I2C マスタは、1 バイトまたは複数バイトを書き換えることができます。よって複雑な実装が不要なシステムに最適なソリューションです。



必要なコンポーネント

- SLG46811V
- ２抵抗

GreenPAK の図

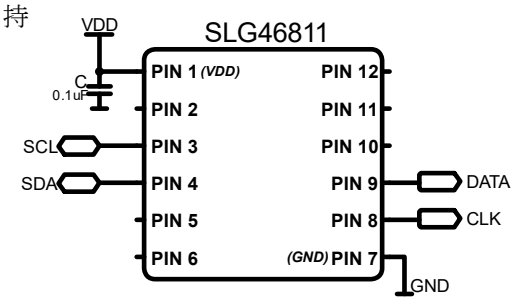


設計手順

1. EPG Generator (EPG テクニックを参照)を設定して I2C 書き込みコマンドを作成し、複数の SDA および SCL パターンを設定します。
2. EPG に チャネルを追加して、SCL で 9 クロックごとに ACK ビットをチェックします。
3. 「Application: 8-bit Multiplexer」で説明されているように、4 ビット マルチプレクサを作成します。SEL0 と SEL1 で、出力するデータ(SDA)を選択します。
4. CNT1/DLY1 および LUT9 を使用して EPG のクロックとして機能する周波数発生器を作成します。
5. CNT0/DLY0 で I2C パターンの開始トリガー信号のデグリッチ・フィルタを設定します。
6. I2C スレーブが応答しない場合、I2C コマンド中に ACK ビット チェックが存在するかどうかが監視され、送信が停止します。

アプリケーション: シフトレジスタを使用した I2C パターン生成

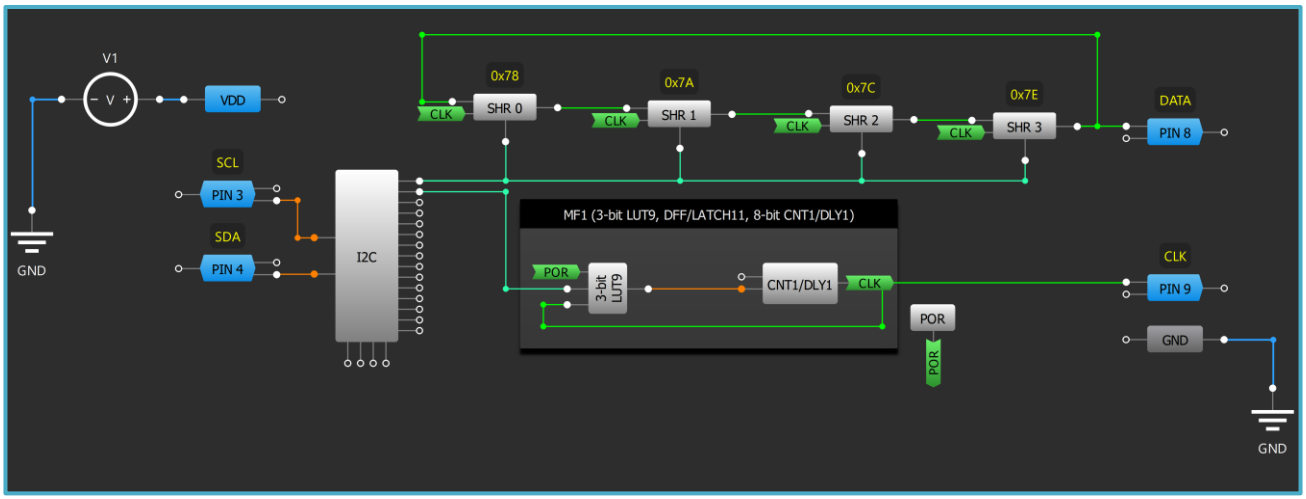
このアプリケーションは、SLG46811 を使用して、最大 32 ビットの容量を持つ単純なパターンジェネレータの構築方法を示します。



必要なコンポーネント

- SLG46811V のような Shift Registers(SHR)を搭載した GreenPAK

GreenPAK の図

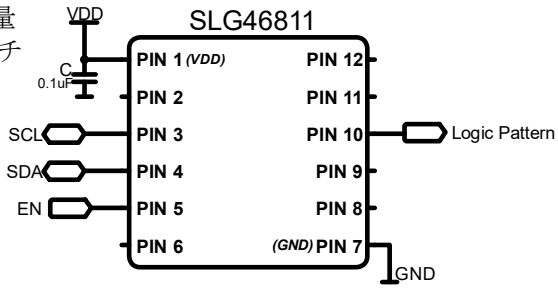


設計手順

1. シフトレジスタ(SHR)の設定をし、直列に接続します。
2. SHR3 の出力を SHR0 の D 入力に接続します。
3. MF1 の CNT1/DLY1 と LUT9 を使用してクロック生成回路を構成します。
4. I2CVirtual input から SHR0-SHR3 の Reset 入力に接続、SHR のクリアまたはパターンの送信を開始する回路を構成します。
5. CNT1/DLY1 はクロックを出力する PIN9 と接続します。

アプリケーション: EPG を使った長いパターン

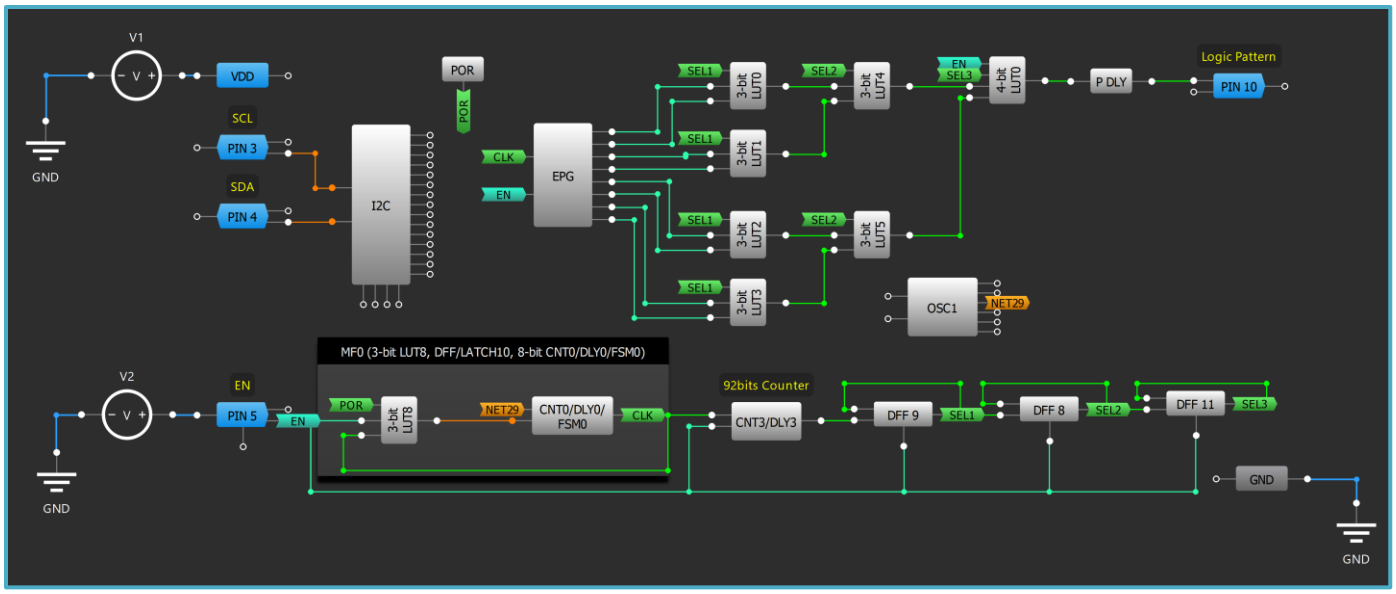
このアプリケーションでは、SLG46811 を利用した最大 736 ビットの容量を持つパターンジェネレータの構成を示します。このようなアプローチは、コスト効率とエネルギー効率の高いアプリケーションにとって有益であることが分かっています。



必要なコンポーネント

- SLG46811V

GreenPAK の図

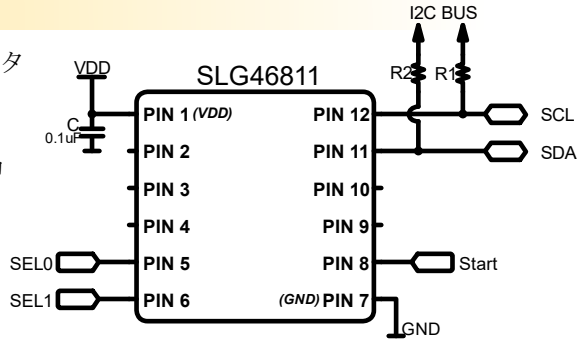


設計手順

1. EPG ジェネレータを構成し、(EPG テクニック参照) 出力するデータを設定します。
2. 「アプリケーション：8 ビットマルチプレクサ」で説明したように、8 ビットマルチプレクサを作成し、EPG の適切な出力に接続します。
3. CNT0/DLY0 を LUT0 をベースに、EPG のクロックとなる周波数ジェネレータを作成します。
4. 周波数ジェネレータにより生成された 92 クロックをカウントするように、CNT3/DLY3 を構成します。
5. 3-bit カウンタとなる DFF8, DFF9 と DFF11 は、EPG から出力されたデータを選択します。「テクニック：ビットストリームの多重化」を参照して下さい。
6. EN 端子 PIN5 を加え、H のときパターンを開始、L でストップとします。

アプリケーション: SPI マスターの基本

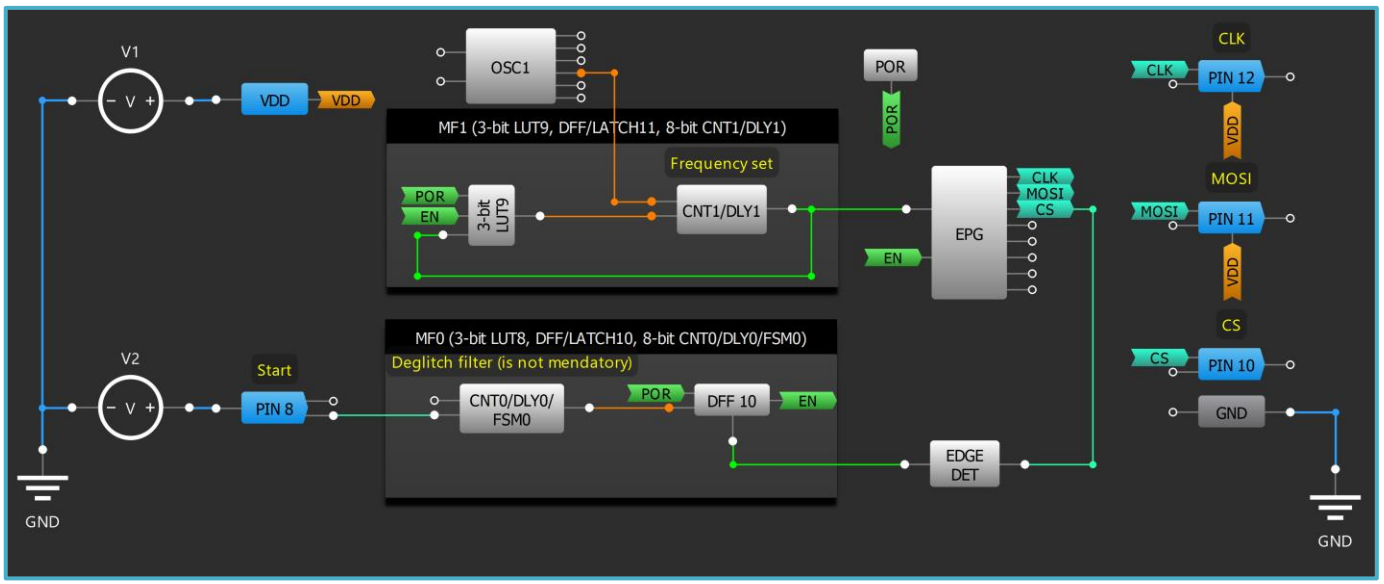
このアプリケーションでは、SLG46811 を用いて簡単な SPI のマスターをどう作るかをご覧頂きます。このデバイスを使った SPI マスターは、ひとつまたは複数バイトの書き換えを行うことができ、複雑な実装を必要としないシステムにとって有益なソリューションを提供します。



必要なコンポーネント

- SLG46811V

GreenPAK の図



設計手順

1. EPG ジェネレータ (EPG テクニック参照) を使って SPI ジェネレータを構成し、MOSI, SCL, CS 信号を端子に割り付けて下さい。
2. CNT1/DLY1 と LUT9 を使って、周波数発生器を作成して、EPG のクロックとします。
3. CNT0/DLY0 によるディグリッチフィルタが DFF10 にクロックを入れることで、SPI パターンにトリガが掛かります。
4. DFF10 をリセットするには、反転出力の EDGE DET を使います。CS が high になると EDGE DET の反転出力が EN 信号をリセットし、パターンをストップします。

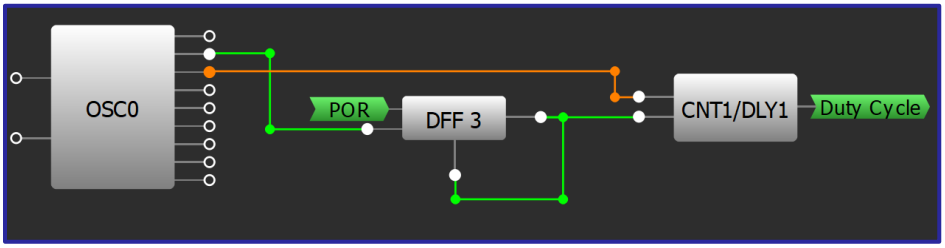
第 6 章: パルスベースの制御

この章では信号のパルス幅を制御するアプリケーションを紹介します。これは最も一般的には PWM に関係するもので、LED コントローラ、モータコントローラ、サウンドによく使われます。

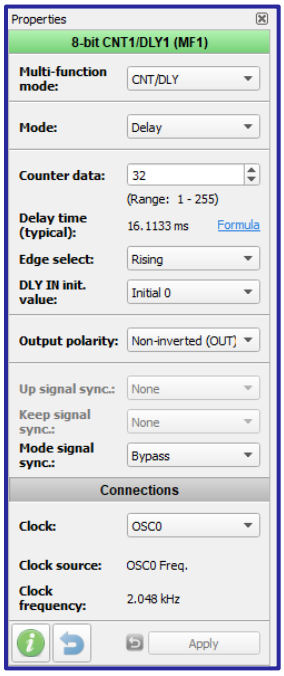
技法：一定のデューティサイクル設定

この技法は任意の GreenPAK で有効です。

一定のデューティサイクルを設定するには、1 つの CNT/DLY ブロック、1 つのオシレータ、1 つの DFF が必要です。マクロセルは図 27 に示すように構成する必要があります。

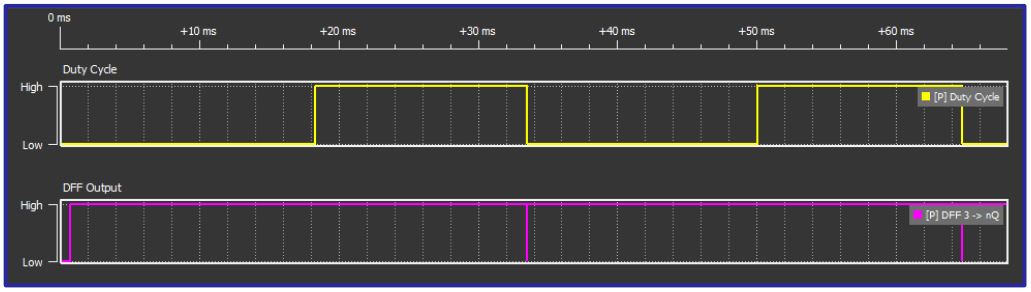


簡単なデューティサイクルの構成



CNT/DLY の構成

オシレータにより周期が決まり、DFF は立ち上がりエッジを検出し、CNT/DLY ブロックに よりデューティサイクルが決まります。オシレータからの立ち上がりエッジが DFF によって登録さ れる場合、CNT/DLY ブロックに Low パルスが送信されます。これにより CNT/DLY 出力 Low が 設定され、出力は遅延の Counter data の値が合うと始めて立上ります。 DFF の初期構成から、Q output polarity を Inverted (nQ)に変更し、DFF の出力を接続しま す。これにより、立ち上がりエッジディテクタとしての動作が可能になります。CNT/DLY 出力は、立ち上がりエッジ がクロックで検出されるまで High が保持され、検出されると Low に落ちます。この用途のために、FILTER/EDGE DET を構成することもできます。 オシレータ OUT0 または OUT1 は、周期を生成するために DFF のクロック入力に接続されま す。周期は常にデューティーサイクルよりも長くなければなりません。この例では、'OUT1' second divider by により周期が OSC/64 に設定されます。 CNT/DLY ブロックの Counter data オプションによりデューティサイクルが設定されます。遅延時間により、信号の Low の期間が設定さ れます。デューティサイクルは次のように計算されます。: $D = \frac{T_{period} - T_{delay}}{T_{period}}$

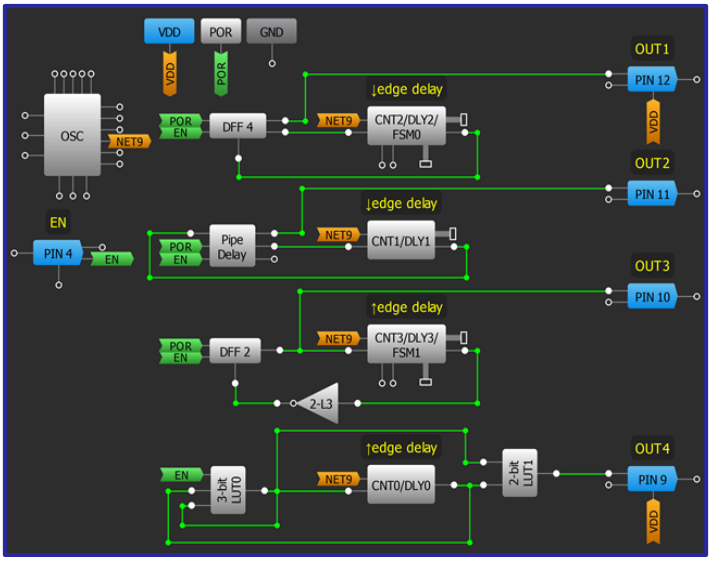


負荷サイクル= 50%のシミュレーション

技法: ワンショット実装

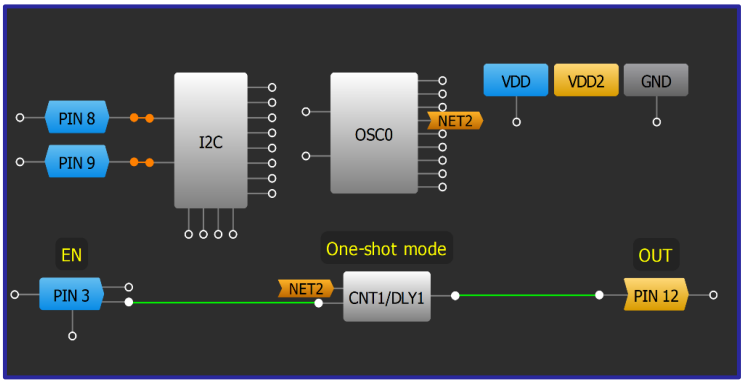
この技法は、任意の GreenPAK で使用できます。

ワンショット回路は、あらかじめ定義された期間で出力パルスを生成します。 回路はパルスを生成すると、安定した状態に戻り、再びトリガーされるまでパルスを生成しません。 リセット機能、ウォッチドッグタイマー、および他の多くのアプリケーションにとって非常に重要なコンポーネントです。 ワンショットは GreenPAK で簡単に実装できます。 以下の図は、入力 EN の立ち上がりエッジによってトリガーされるワンショットインパルスを作成するいくつかの方法を示しています。 パルスの持続時間は、使用する DLY ブロックのカウンタデータ値を変更することによって調整できます。



異なるワンショット回路の実装

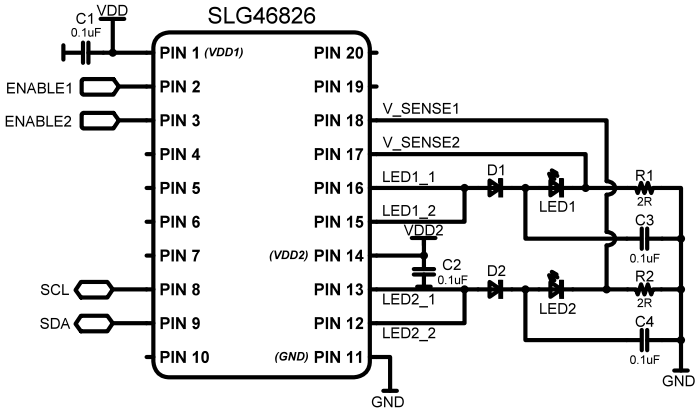
多くの GreenPAK(下の図を参照)では、1 つのショットを実装するのに使用する DLY ブロックは 1 つだけです。ユーザーが行う必要がある唯一のことは、ブロックプロパティウィンドウのモードを「ワンショット」に切り替え、検出するエッジを選択することです。立ち上がり、立ち下がり、または両エッジに設定できます。



SLG46826 でのワンショット実装

アプリケーション：定電流 LED ドライバ

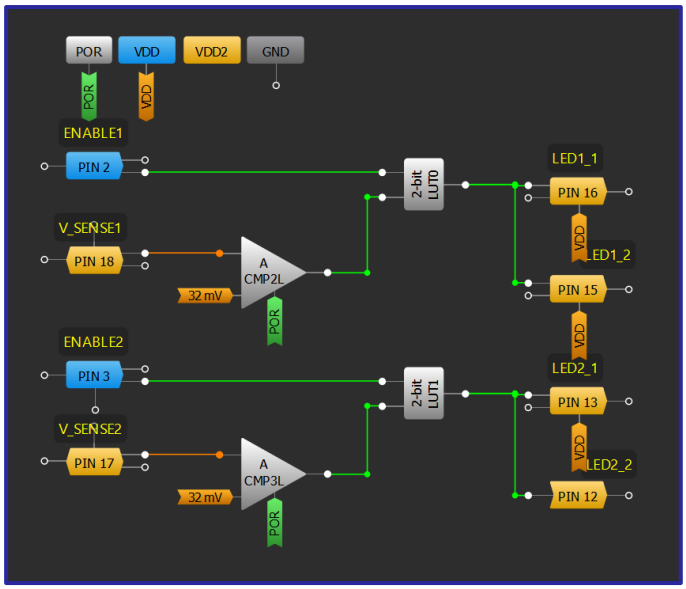
LED ドライバは LED への電流を供給し、調整します。バーンアウト/熱暴走を防止するため、電力を安全なレベルに維持します。



必要なコンポーネント

- 任意の GreenPAK
- 4 つのコンデンサ
- 4 つのダイオード（2 つの LED、2 つのシリコンダイ オード）
- 2 つの抵抗

GreenPAK の図

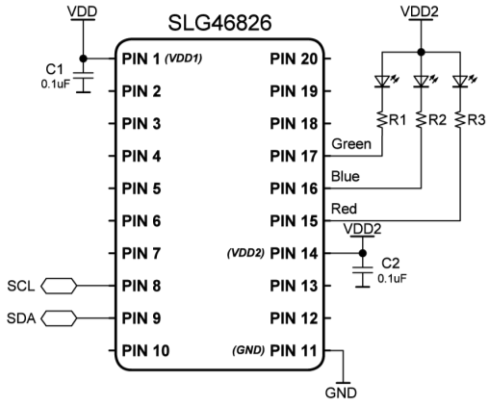


設計手順

1. ACMP の IN- 入力を適切なしきい値（検出電流に対応）に設定します。
2. LUT を構成して LED 出力を有効化します。
3. LED1_1 および LED1_2 をシリコンダイオードの陽極に接続し、シリコンダイオードの陰極を LED の陽極に接続します。
4. LED の陰極と接地の間に抵抗を接続します。
5. LED の陽極と接地の間にコンデンサを接続します。
6. LED2 の出力について手順 3～5 を繰り返します。
7. V_SENSE1 および V_SENSE2 をそれぞれ LED1 および LED2 の陰極に接続します。

アプリケーション: I2C による RGB LED 制御

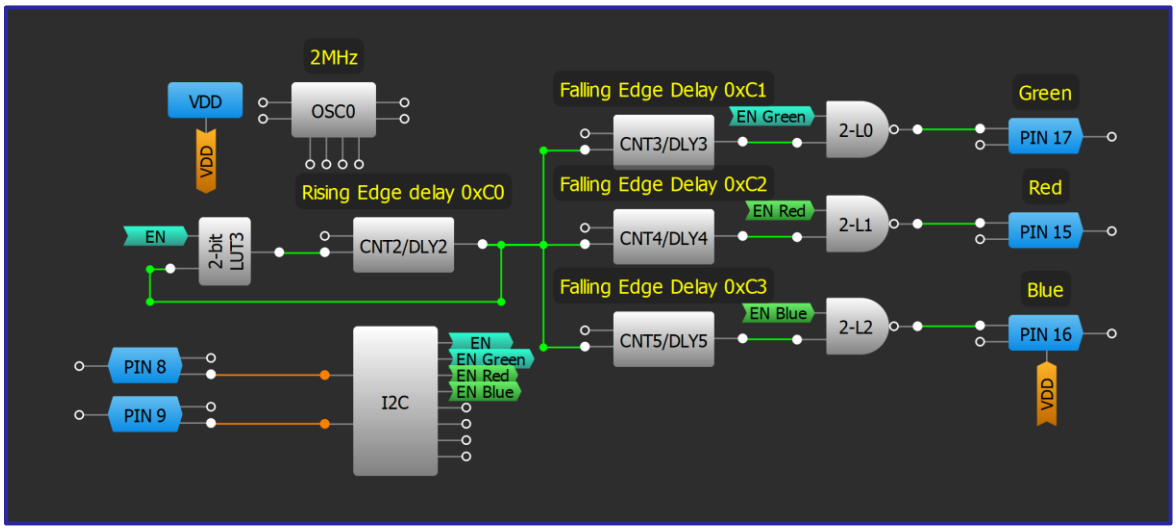
RGB LED は、LED 表示システムの複雑な制御に使用され、GreenPAK で制御できます。このアプリケーションでは、デューティサイクルを変更して異なる色を生成する簡単な方法として I2C が使用されます。



必要なコンポーネント

- I2C を搭載した任意の GreenPAK
- RGB LED
- 3 つの抵抗

GreenPAK の図



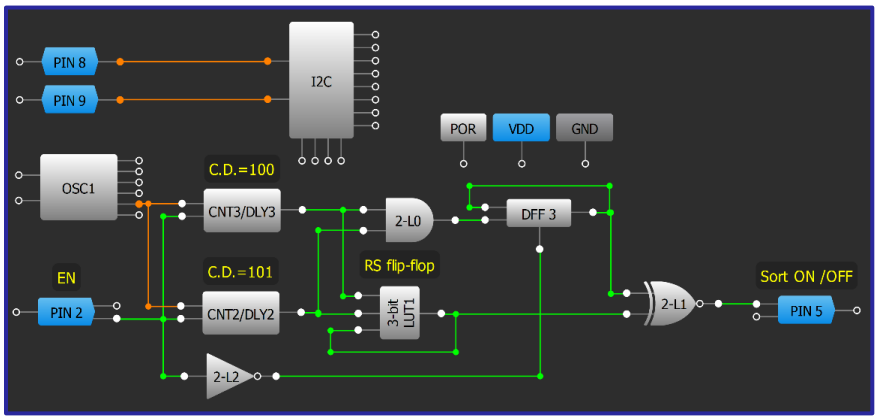
設計手順

1. GPIO ピンを RGB カソード接続用のオープンドレイン出力として設定します。
2. LUT ロジックと **CNT/DLY2** を追加して、**EN** 信号付きのジェネレータを作成します。
3. CNT/DLY ブロックを立ち上がりエッジ遅延に設定します。
4. 「技法: LUT マクロセルを使用した標準ロジックの設定」を使用して、出力ごとに LUT を追加および設定します。
5. 各 LUT 出力を所望の出力ピンに接続します。
6. I2C 仮想入力は、I2C 仮想出力アドレスを使用して個別にまたは同時に変更できます。
7. CNT/DLY ブロックのカウンタデータは、I2C を使用して個別にまたは同時に変更することができます。

技法: ブリージング LED パターンの作成

この技法は、任意の GreenPAK 内で使用できます。独立したソフト ON/OFF チャンネルの数は、特定のパーツ内で使用可能なカウンタの数によって異なります。

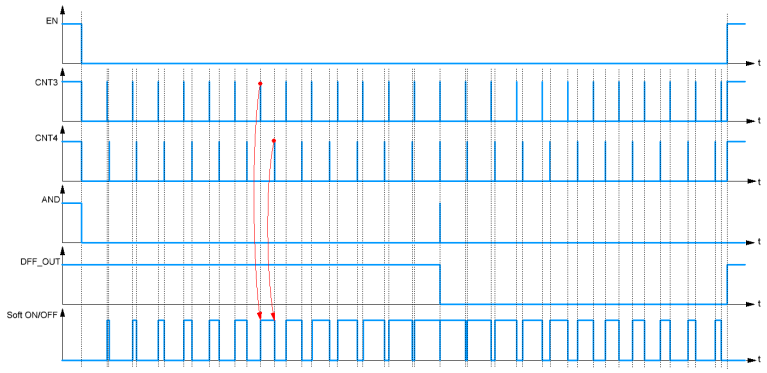
ブリージング LED パターンは、2 つのカウンタ間の一貫したペースの違いによって生成できます。 各カウンタは、プログラムされた周期の 1 クロックサイクルに対して 1 個のハイパルスを出力します。2 つの CNT/DLY ブロックは、異なるカウンタデータ設定でプログラムされ、それらの出力間に小さなオフセットを生じます。これらの出力信号は、デバイス内のフリップフロップをセットおよびリセットするために使用されます。次の図は、**CNT2/DLY2** が ON 期間を設定し、**CNT3/DLY3** がデューティ サイクルを設定する基本的な実装を示しています。



LED ブリージング実装

上の図の実装では、PWM の周波数は **CNT2** によって設定され、次の式で計算できます。

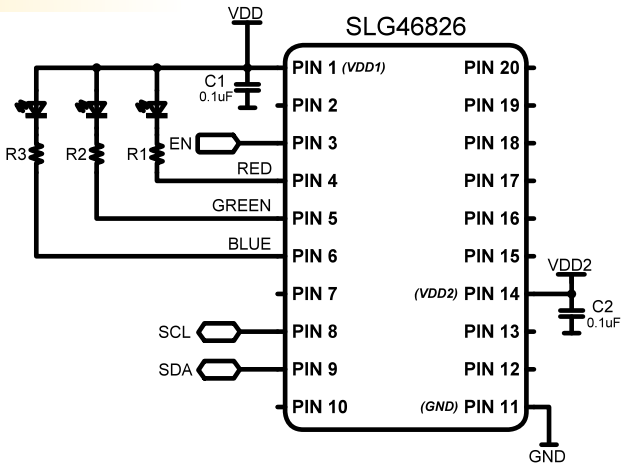
$$f_{PWM} = \frac{f_{OSC}}{(Data_{CNT2}+1)}$$



小さなオフセットの効果は、下図の波形で示されています。 PWM サイクルは、カウンタの出力が一致すると終了します。これにより、AND ゲートと DFF フロップで短いハイインパルスが発生します。 NXOR ゲートは PWM を反転させ、ソフトオフを提供します。 PIN2 はイネーブル信号で、これが HIGH の間、カウンタはハイレベルリセットされています。

アプリケーション: ブリージング RGB LED

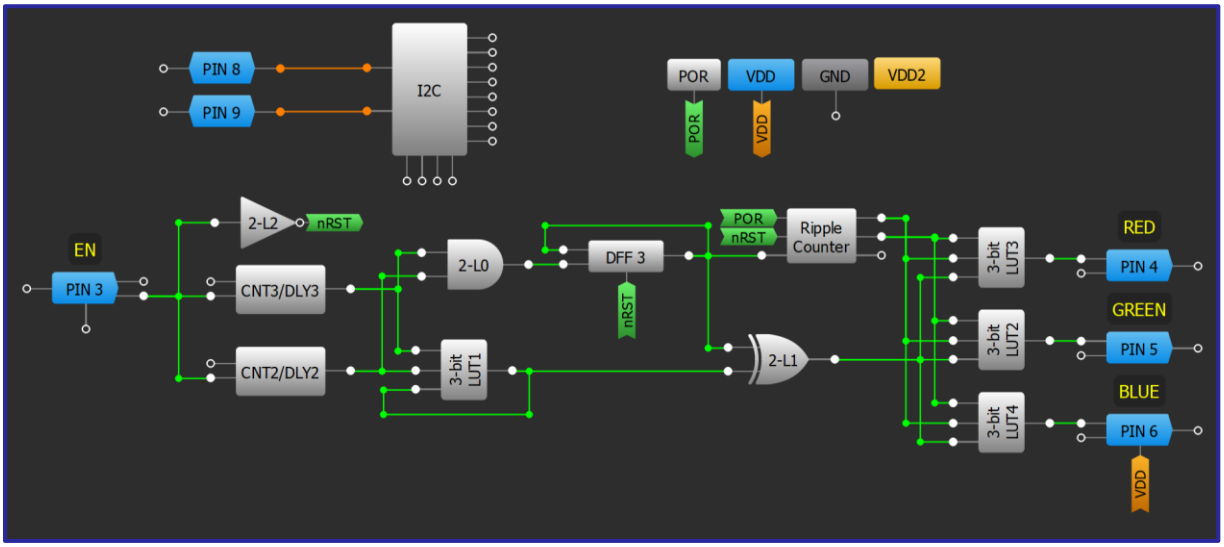
RGB LED は、LED 表示システムの複雑な制御に使用され、GreenPAK で制御できます。これらは、ブリージングパターン用のソフト ON/OFF 回路と組み合わせることができます。



必要なコンポーネント

- 任意の GreenPAK
- 1 つの RGB LED
- 3 つの抵抗

GreenPAK の図

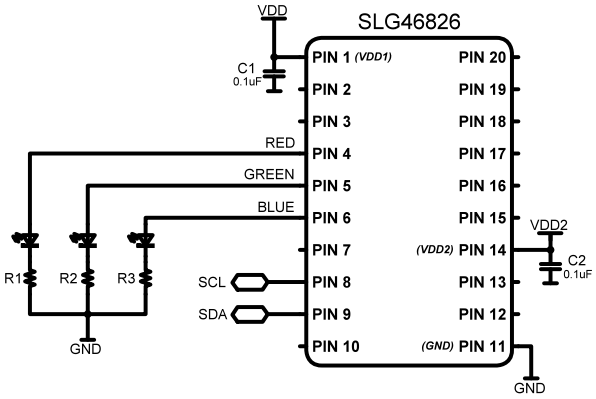


設計手順

1. GPIO ピンをオープンドレイン NMOS 出力として設定します。
2. 「技法:ブリージング LED パターンの作成」に示すようにソフト ON/OFF 回路を作成します。
3. リップルカウンタの設定 - Functionality mode を Range:SV-EV cycle (SV=1、EV=3)に設定します。
4. LUT をまとめてデマルチプレクサとして設定します。
5. RGB ブリージングを開始/停止するためのイネーブル(**EN**)信号を追加します。

アプリケーション: I2C によるブリージング RGB LED 制御

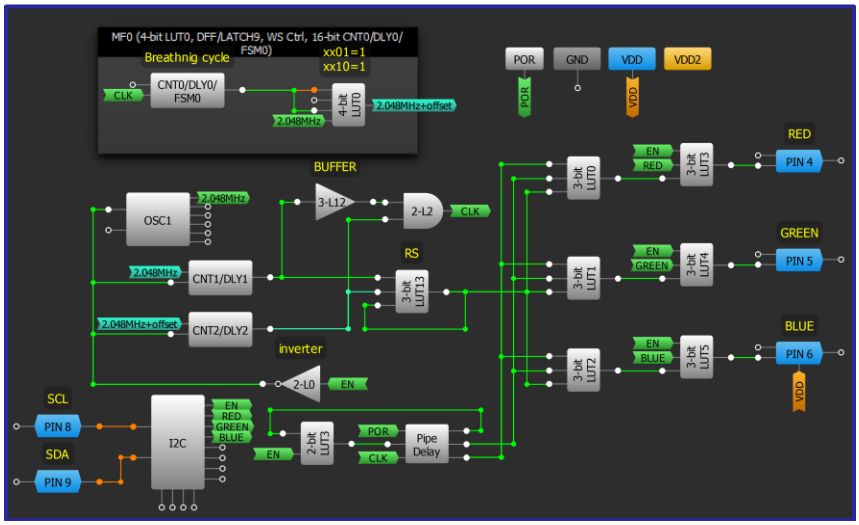
RGB LED は、LED 表示システムのより複雑な表示のために使用されます。これらは、ブリージングパターン用のソフト ON/OFF 回路と組み合わせることができ、さらに I2C で制御することができます。CNT0 カウンタデータを変更すると、ブリージング周期が変更されます。



必要なコンポーネント

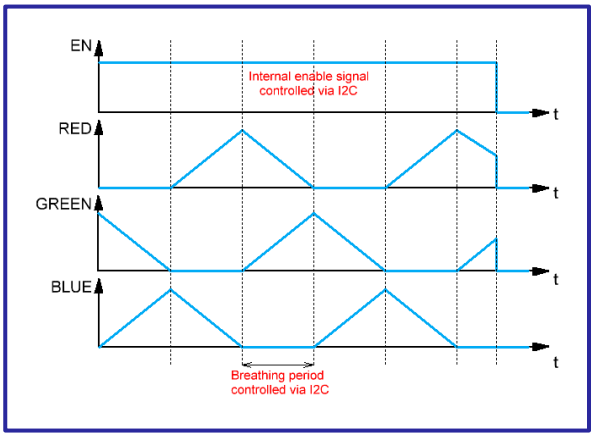
- I2C を搭載した任意の GreenPAK
- 1 つの RGB LED
- 3 つの抵抗

GreenPAK の図



設計手順

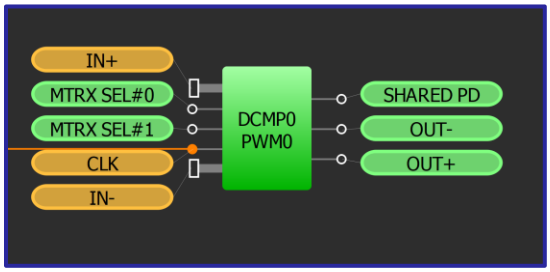
1. 「技法:ブリージング LED パターンの作成」に示すようにソフト ON 回路を作成しますが、代わりに **CNT1** と **CNT2** のカウンタデータを同じ値に設定します。
2. **CNT1** と **CNT2** の間に小さなオフセットを追加する MF0 を追加します。
3. I2C 経由で制御される EN 信号によって切り替えられるマルチプレクサとして LUT3 ～ LUT5 を設定します。
4. LUT0 ～ LUT2 を、タイミング図に従ってブリージング信号を渡すデマルチプレクサとして構成します。



技法: PWM モードでの DCMP/PWM マクロセルの使用

この技法は、SLG46140、SLG46620、およびSLG46621 で使用可能な DCMP ブロックに関するものです。

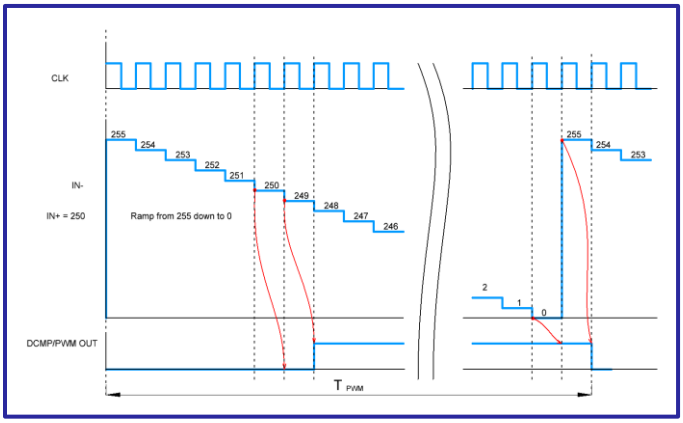
DCMP/PWM マクロセルの概要



DCMP/PWM マクロセルは、2 つの 8 ビット値または生成された PWM 信号を比較するために使用されます。 IC ごとに独立して動作できる 3 つの DCMP/PWM ブロックがあり、各 DCMP/PWM には PWM 信号の生成に使用できる 2 つの 8 ビット入力(**IN+**、**IN-**)があります。 入力 **MTRX SEL#0** および **MTRX SEL#1** は、スタティック PWM 生成時に、使用可能な 4 つのレジスタのうちの 1 つを選択するために使用されます。 入力 **SHARED PD** は、デバイスの電源オンまたはオフに使用されます。 PWM 出力デューティサイクル範囲は、0%～99.61%または 0.39%～100%の範囲に設定できます。

PWM 信号の作成

PWM 発生器の 1 つの入力は、周期的に変化する線系のカウンタデータです。これは、255 から 0 まで、またはその逆にカウントされるカウンタからのものです。



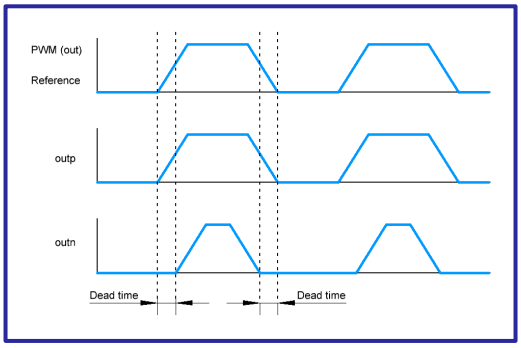
PWM モードの DCMP/PWM

もう一方の入力は、少なくとも 1 つの PWM 信号期間(PWM ランプカウンタ期間)の間、安定している必要があります。これは、 SPI、ADC、FSM ブロックからのデータ、 または DCMP/PWM の内部レジスタからのデータである場合もあります。

右の図は、**IN-**が 255 から 0(PWM ランプカウンタ)までカウントされる CNT/DLY ブロックに接続され、**IN+**ソースが 250 に設定された内部レジスタである場合の DCMP/PWM の動作を示しています。

IN+設定は、マクロセルの動作のキーです。静的 PWM 値は、内部レジスタを使用して作成することができます. PWM ダイナミックフィードバックは ADC を使用して行うことができます。MCU 制御 PWM は、SPI インターフェースを使用して設定することができます。

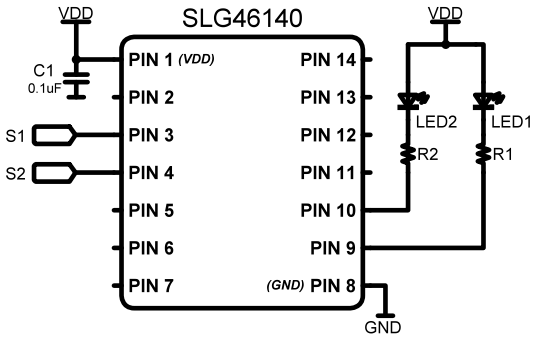
出力 **OUT-** および **OUT+** には、10 ～ 80 ns のデッドバンド時間があり、DCMP/PWM のプロパティウィンドウで設定できます。



OUT- と OUT+　のデットバンド時間

アプリケーション: PWM の選択

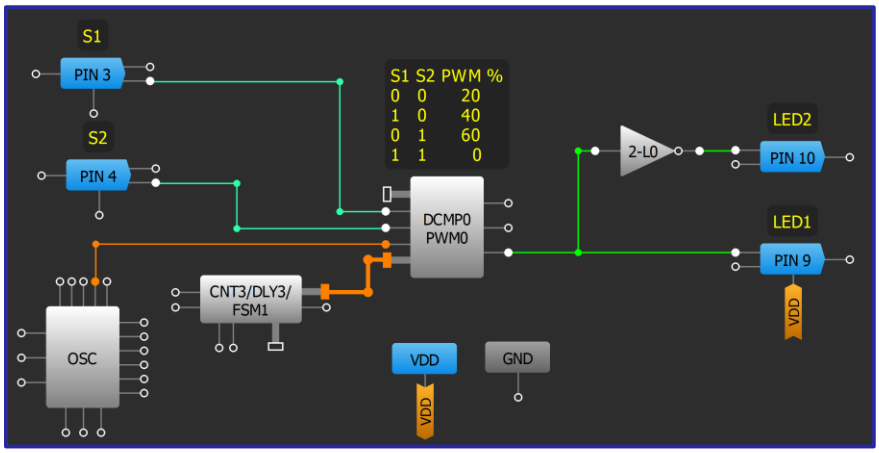
PWM の選択は、LED の輝度の調整やファン速度の制御などの機能によく使用されます。この実装では、2 つの LED が 2 つのスイッチの入力に基づいて個別の輝度レベルに調整されます。



必要なコンポーネント

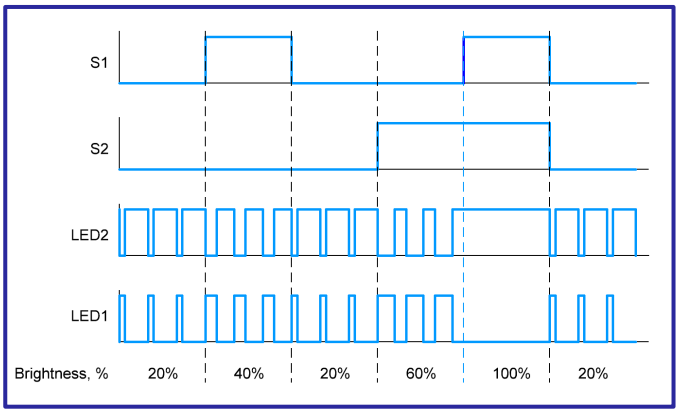
- DCMP 搭載の GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図



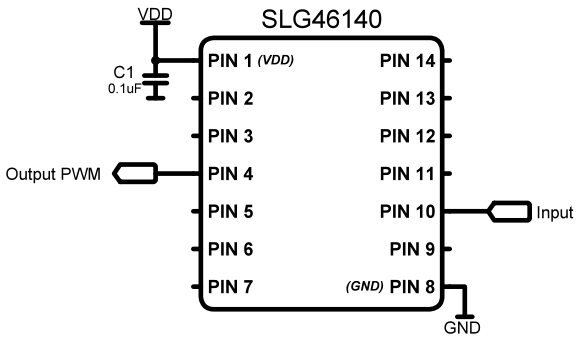
設計手順

1. **SHARED PD** 入力から VDD を除去して、DCMP を使用可能にします。DCMP/PWM パワーレジスタを「パワーオン」に設定します。**IN+ セレクタ**を「マトリックスで選択したレジスタ」に設定します。**IN-セレクタ**を "FSM1 [7:0]"に設定します。 DCMP0 のレジスタを設定します。レジスタ 0 - 51。 レジスタ 1 - 102。 レジスタ 2 – 154; レジスタ 3 – 0。
2. カウンタ/FSM として設定された CNT/DLY ブロックを追加します。カウンタ データを 255 に設定します。
3. RC OSC 電源モードを"強制電源オン"に設定します。
4. **PIN10** と **PIN9** を設定してオープンドレイン NMOS にします。
5. LUT をインバータとして追加します。
6. 入力ピンを DCMP/PWM ブロックの **MTRX SEL** ピンに接続します。



アプリケーション: ACMP と DAC を使用した PWM ジェネレータ

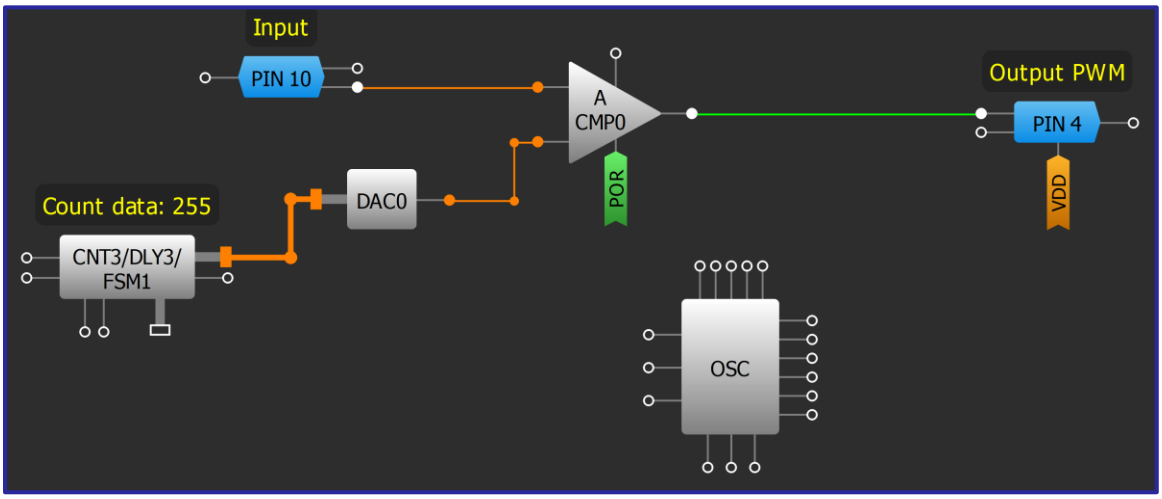
PWM ジェネレータを使用して、DC モータや LED などのデバイスを制御できます。この実装では、ACMP が DAC0 の信号と比較するアナログ信号を使用します。 CNT3 は、DAC の値を生成するために使用されます。



必要なコンポーネント

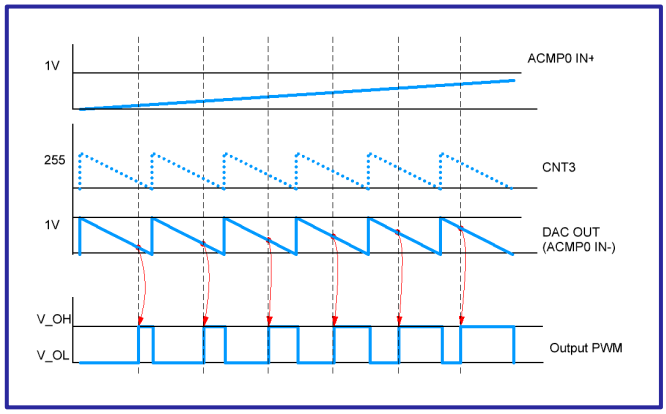
- DAC を搭載した GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図



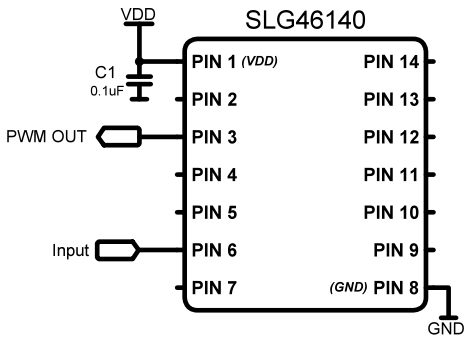
設計手順

1. **ACMP0 PWR UP** 入力を POR に接続します。**IN-source** を "Ext. Vref (DAC0 out)" に設定します。
2. **DAC0** のパワーオン信号を「パワーオン」に設定し、**入力選択**を「DCMP1 の入力から」に設定します。
3. FSM 互換の CNT/DLY を "カウンタ/FSM" モードに設定し、カウンタデータ = 255 にします。
4. RC OSC 電源モードを「強制電源オン」に設定します。



アプリケーション: ADC を使用した PWM ジェネレータ

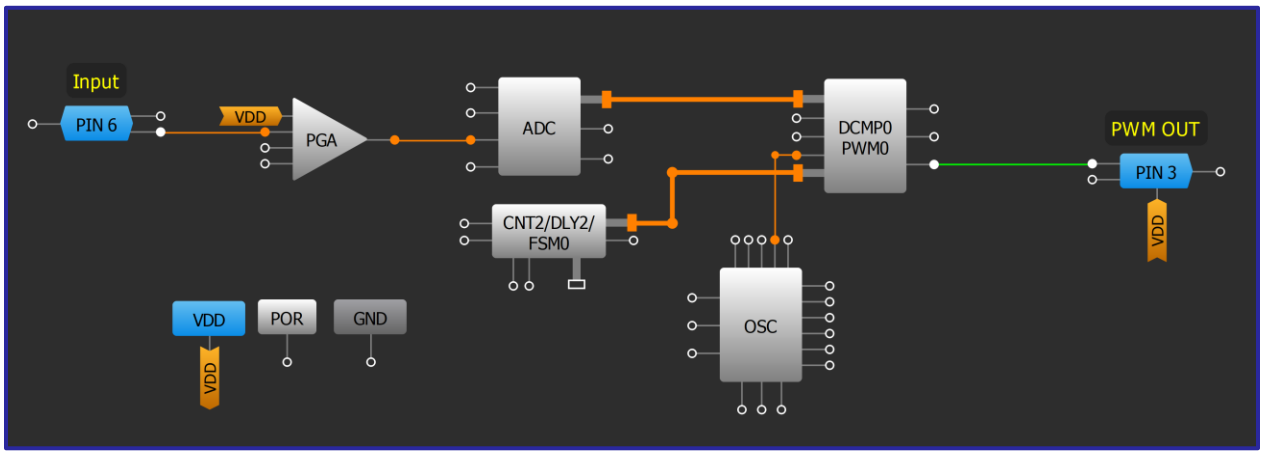
PWM ジェネレータを使用して、DC モータや LED などのデバイスを制御できます。この実装では、ADC に接続されたアナログ信号を使用して、PWM0 の CNT2 の値と比較します。CNT2 値がデジタル化されたアナログ信号よりも小さい場合、PWM0 の出力は High になります。CNT2 値が 0 の後、PWM0 の出力は Low になります。



必要なコンポーネント

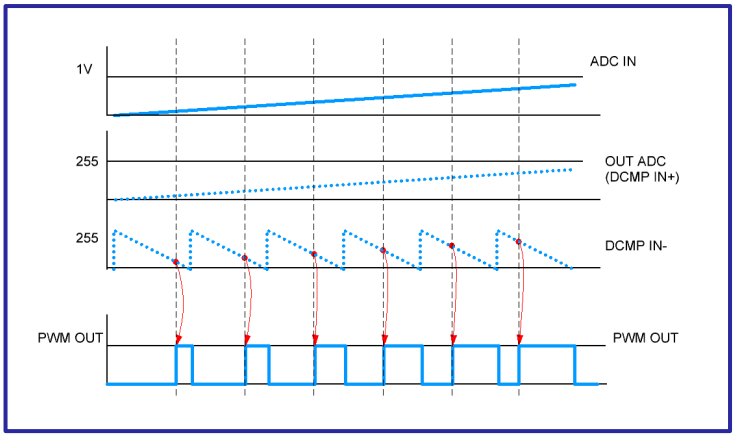
- ADC を搭載した GreenPAK
- それ以外のコンポーネントは不要です

GreenPAK の図



設計手順

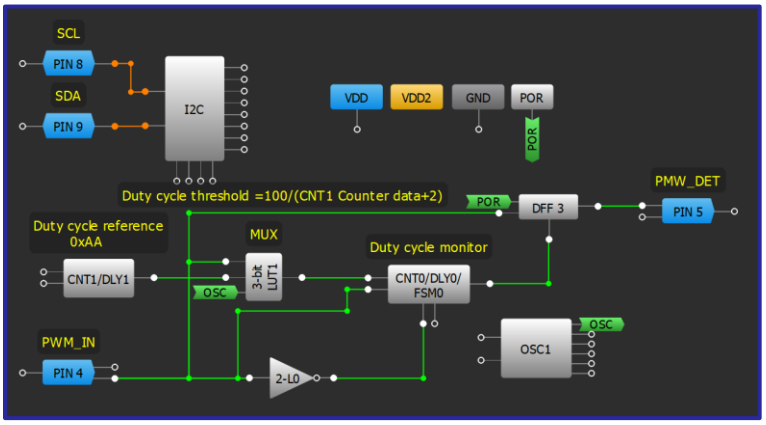
1. ADC の **PWR DOWN** 入力から VDD を取り外し、PGA Power on signal を「パワーオン」に設定します。
2. **PIN6** を「アナログ入力/出力」に設定します。
3. **SHARD PD** 入力から VDD を削除して、**DCMP0/PWM0** を設定します。DCMP/PWM パワーレジスタは電源をオンにします。**IN+セレクタ**が「ADC [7:0]」に接続され、**IN- セレクタ**が「FSM0 [7:0]」に接続されていることを確認します。
4. **4 ビット LUT1/14 ビット CNT2/DLY2/FSM0** を、カウンタ データが 255 に等しい「カウンタ/FSM」として設定します。
5. **DCMP0/PWM0 OUT+** を出力ピンに接続します。



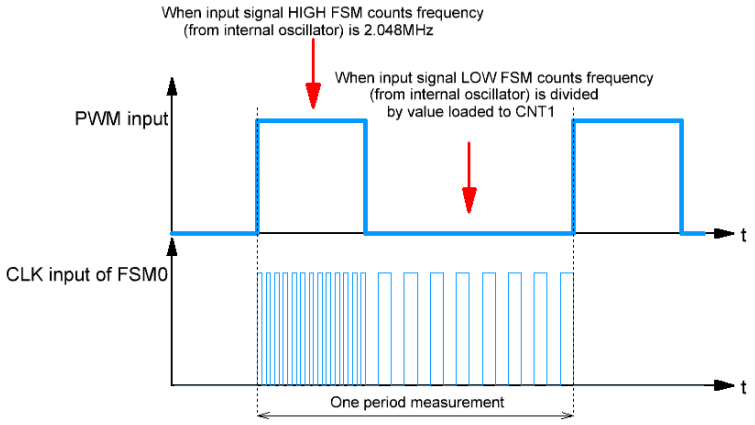
技法: デューティ サイクル検出

この技法は、任意の GreenPAK で使用できます。 入力周波数範囲は最大 FSM カウンタデータによって制限されるため、16 ビット FSM を使用することをお勧めします。 PWM 検出入力周波数は、精度を向上させるためにデューティサイクル基準周波数よりもはるかに遅くする必要があります。

デューティサイクル検出は、過負荷保護、DC/DC 変換、サーボモータ制御、プロトコル検出などの機能にとって重要です。このデザインは、FSM ブロックを備えた GreenPAK を使用して簡単に実装できます。(以下の例では、SLG46826 チップを使用していますが、下の図を参照してください)。



上記の実装では、**PIN4** が HIGH になると、**FSM0** は内部発振器によってクロックダウンされ、カウントダウンを開始します。FSM0 は、PIN4 の立ち上がりエッジによって 65535 に設定されます。**PIN4 が LOW** になると、**FSM0** は内部発振器からの周波数を **CNT1** のデータ値で割った値でカウントアップを開始します。**FSM0** が 65535 に達すると、DFF3 は PIN4 からの次の立ち上がりエッジによって LOW に設定され、デューティサイクルが設定されたしきい値を下回ったことを示します。

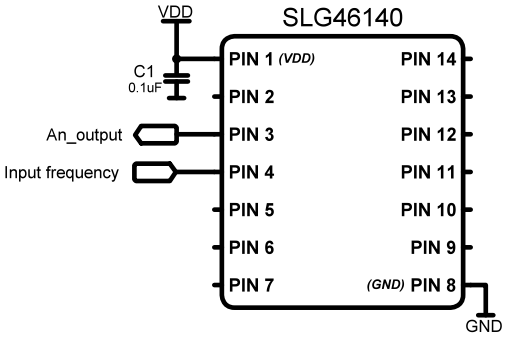


デューティサイクル基準周波数は、I2C を介して **CNT1** カウンタデータ値を変更することによって調整することができます。デューティサイクルしきい値を計算するには、次の式を使用します。

$$\textit{デューティサイクルしきい値} = \frac{100}{\text{CNT1 カウンタデータ} + 2}\ (\%)$$

アプリケーション: 周波数アナログ電圧コンバータ

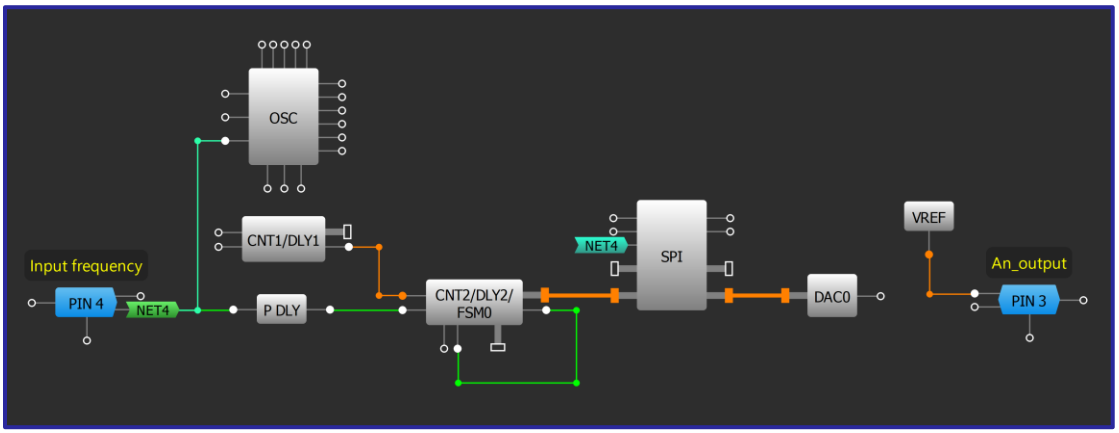
このアプリケーションは、入力周波数をアナログ電圧に変換します。入力周波数は事前定義された範囲にあり、それは、設計コンポーネントによって選択および調整することができます。出力アナログ電圧は一定で、要求に応じて変更することができます。

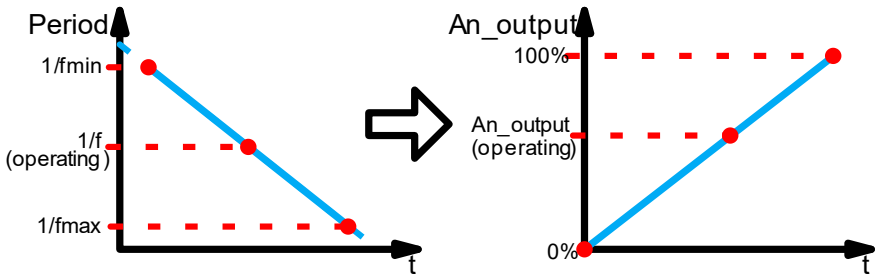


必要なコンポーネント

- SPI と DAC を搭載した任意の GreenPAK

GreenPAK の図





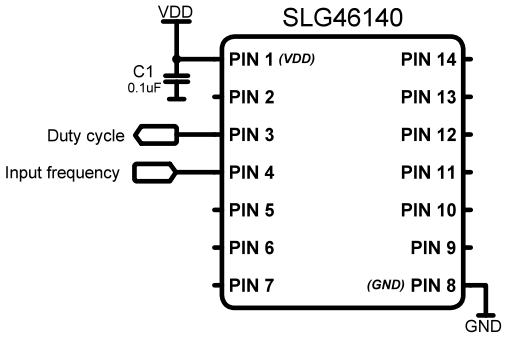
設計手順

1. SPI を「ADC/FSM Buffer」モードに設定し、**PAR input data** ソースを「FSM0 [15:8]」に変更します。
2. **FSM0** ブロックを「Set (counter value)」に設定し、クロック ソースを **CNT1** に変更します。
3. DAC **入力セレクタ**を「From DCMP1's input」および VREF **ソースセレクタ**を「DAC0 out」に設定します。
4. 入力周波数範囲と出力アナログ電圧を求めるには、次の式を使用します。

$$f_{min} = \frac{f_{osc}}{(CNT1+1)\cdot FSM0} \qquad f_{max} = \frac{f_{osc}}{(CNT1+1)} \qquad An_output_{operating} = V_{ref_max} - \frac{f_{min}\cdot V_{ref_max}}{f_{operating}}$$

アプリケーション: 周波数デューティサイクルコンバータ

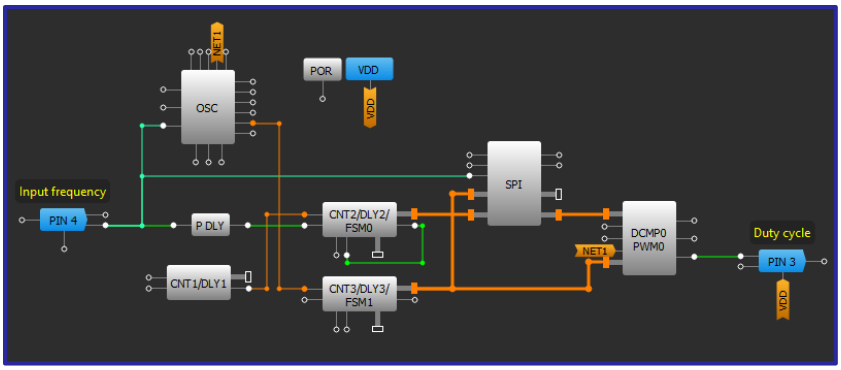
このアプリケーションは、入力周波数を特定のデューティサイクルに変換するために使用できます。入力周波数は事前定義された範囲内にあり、カウンタを調整することで変更できます。 出力 PWM 周波数は一定ですが、所定の要件に応じて変更することができます。

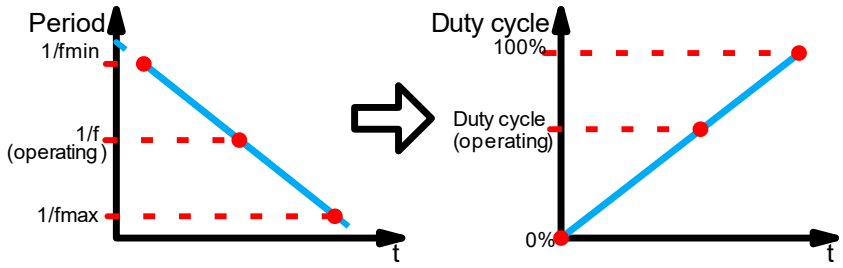


必要なコンポーネント

- SPI と DCMP を搭載する任意の GreenPAK

GreenPAK の図





設計手順

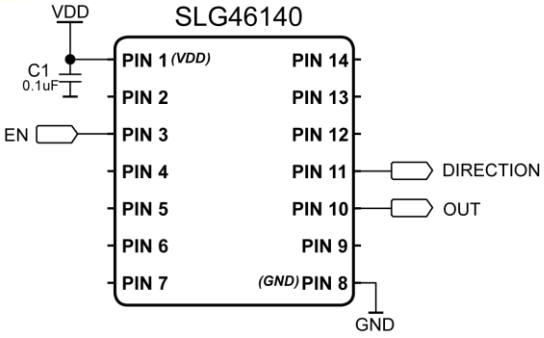
1. SPI を「ADC/FSM buffer」モードに設定し、PAR 入力データ ソースを "FSM0[15:8] FSM1[7:0]" に変更します。
2. **FSM0** ブロックを「Set (counter value)」に設定し、クロックソースを「8-bit CNT1/DLY1(OUT)」に変更します。
3. "SPI [15:8]" データと "FSM1 [7:0]" データを比較するために **DCMP0** を設定します。
4. 入力周波数範囲と動作デューティサイクルは、次の式を使用して計算します。

$$f_{min} = \frac{f_{osc}}{(CNT1+1)\cdot FSM1} \qquad f_{max} = \frac{f_{osc}}{(CNT1+1)}$$

$$Duty\ Cycle_{operating} = \left(1 - \frac{f_{min}}{f_{operating}}\right) \cdot 100\%.$$

アプリケーション: リニア周波数変調

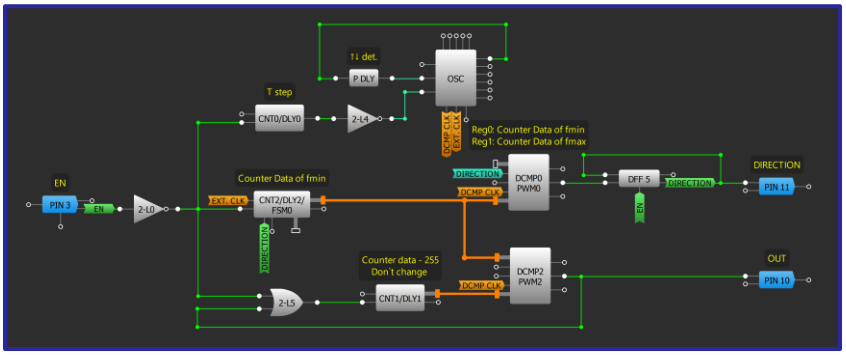
このアプリケーションは、特定の時間内に fmin から fmax へ、またはその逆に徐々に変更することができる周波数を生成するために使用することができます。 周波数は、広い周波数範囲ではリニアに変化しません。

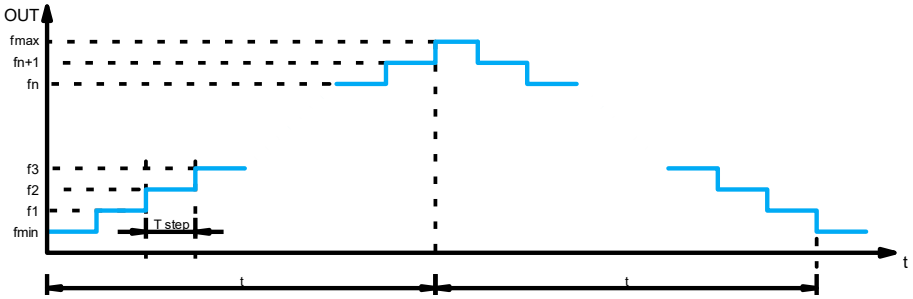


必要なコンポーネント

- DCMP を備えた任意の GreenPAK

GreenPAK の図





設計手順

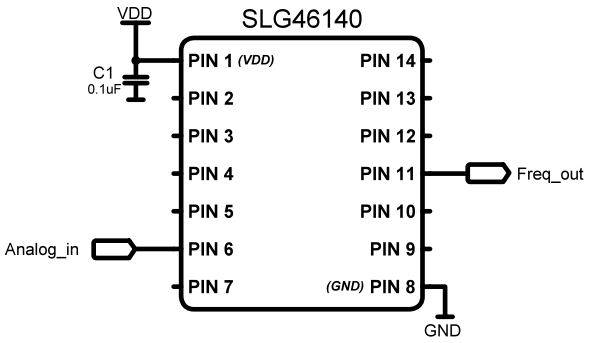
1. 所望の GPIO ピンで構成します。
2. 上記のように、LUT、P DLY、DFF、DCMP、CNT/DLY/FSM ブロックを追加、接続、および設定します。
3. 最小および最大周波数の Counter Data、および立上がり/立下り時の周期は次のように計算されます。

$$Counter\ Data\text{max}\ (min) = 255 - (\ fosc\ f\text{max}\ (min) - 1)$$
$$Tstep = t \cdot fmax \cdot fmin\ /fosc \cdot (fmax - fmin)$$

4. 特定の DCMP セレクタに適切なカウンタ データを設定します。
5. ステップ時間 Tstep となるよう、適切な **CNT/DLY0** カウンタ データを設定します。

アプリケーション: 電圧制御発振器

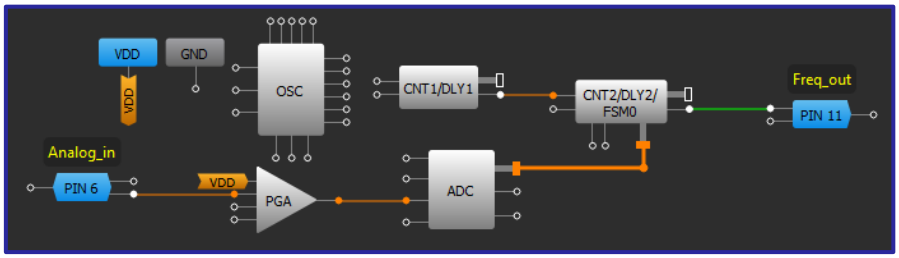
このアプリケーションは、入力アナログ電圧を周波数に変換するために使用できます。周波数は FSM0 によって生成され、カウンタデータは ADC によって設定されます。FSM0 のカウンタデータは、周波数分割係数を決定します。周波数範囲を広げるために、CNT1 は周波数前置分周器として使用されます。

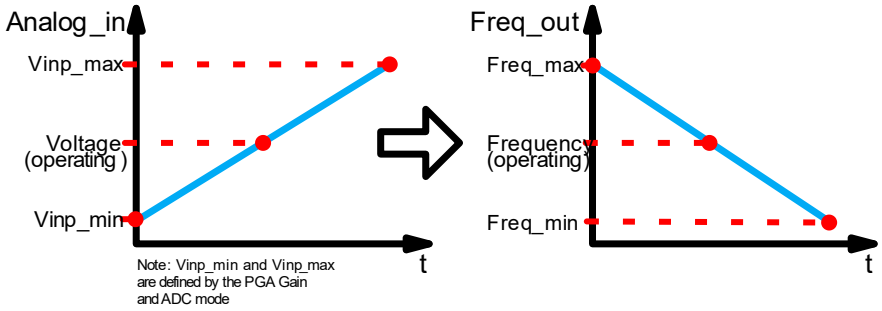


必要なコンポーネント

- ADC を搭載した任意の GreenPAK

GreenPAK の図





設計手順

1. **PIN6** を「Analog input/output」として設定します。
2. **FSM0: FSM データソース**の接続を "ADC" に変更し、**クロック**を "8-bit CNT1/DLY1(OUT)" に変更します。
3. ADC ブロックの電源を入れ、PGA **ゲイン**を「x0.25」から「x1」に変更します。
4. 出力周波数範囲と動作周波数を求めるには、次の式を使用します。

$$fmin = \frac{fosc \cdot Vinp_max}{2 \cdot (CNT1+1)} \qquad fmax = \frac{fosc}{256 \cdot (CNT1+1)} \qquad Frequency(operating) = fmax \cdot \frac{Vinp_max}{Voltage(operating)}$$

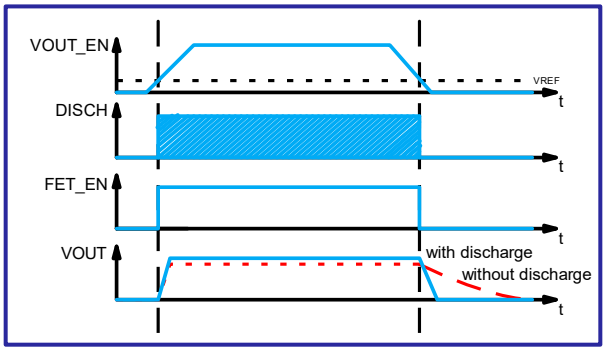
第 7 章: Power Management

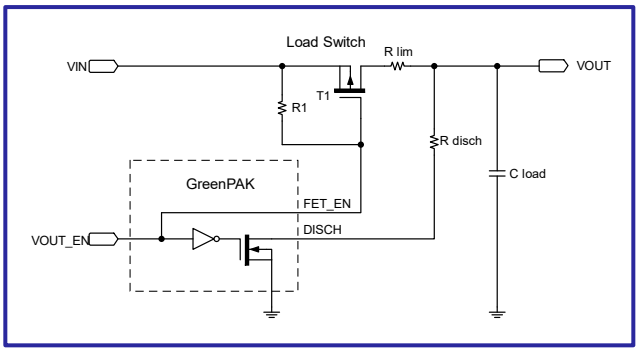
この章では電子システムの電力使用を管理するアプリケーションを紹介します。電力管理を行うアプリケーションにはチャージポンプ、LDO、放電回路などがあります。

技法: 出力放電

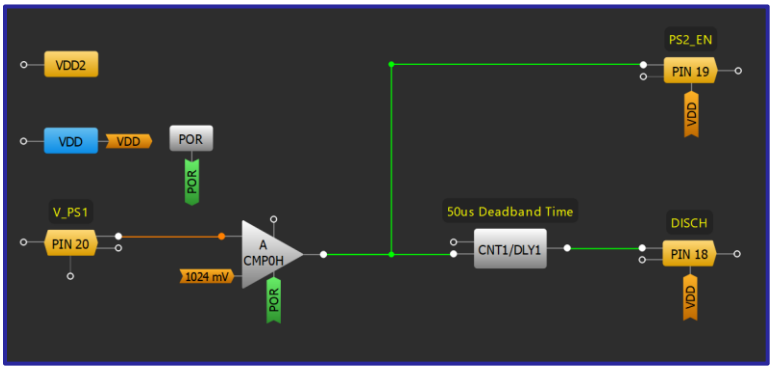
この技法は、任意の GreenPAK で使用できます。

出力放電回路は、出力ピンをフローティングやシステムの電圧低下から防ぐ方法です。これにより、デバイスを無効にする必要がある場合に出力ピンが既知の「Zero」状態に設定されることを保証しながら、動作中にはリークがないようにします。





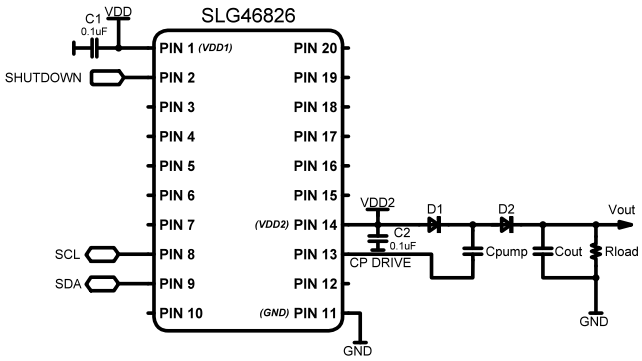
上の図は、負荷スイッチ出力 **VOUT** 用の急速放電回路の実装を示しています。GreenPAK のオープンドレイン NMOS 出力のドレインは **VOUT** に接続されています。 NMOS のゲートは GPIO 内で反転しているため、マトリックス内の **VOUT_EN** 信号を反転させる必要はありません。 **VOUT_EN** が HIGH の場合、負荷スイッチがオンになり、リークを防ぐために GND への放電経路がオープン（非接続）となります。**VOUT_EN** が LOW の場合、負荷スイッチはオフになり、GND への放電経路は閉じられ（接続）、**VOUT** を急速に放電します。 FET を流れる電流を制限し、放電を制御するために、2 つの抵抗 R_{lim} と R_{disch} が追加されます。



上の図は、先行して電源のレベルを監視する電源シーケンサー内の条件付き出力放電回路の例を示しています。 **PS2_EN** をプッシュプル出力として、**DISCH** をオープンドレイン NMOS 出力として設定します。**V_PS1** が 1024mV を下回ると、この回路は電源をオフにし、出力を GND に放電します。50µS のデッドバンド時間が **CNT/DLY1** で加算されます。

アプリケーション：チャージポンプ

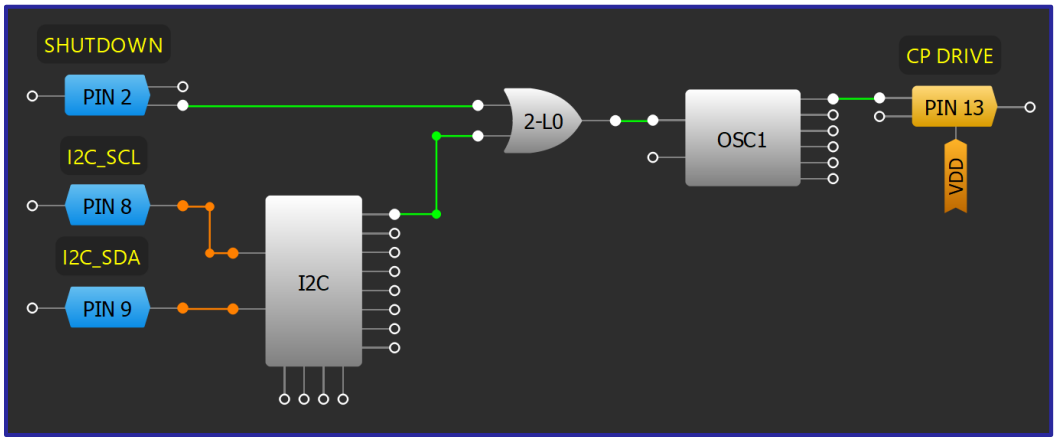
チャージポンプは、エネルギー電荷充電用のキャパシタを用いて、異なるレベルの電圧を生成する DC-DC コンバータです。特定のインターフェース回路、センサー などで使用する追加の電圧レベルを供給します。最良の性能を得るために、ショットキーダイオードの使用を推奨します。



必要なコンポーネント

- 任意の GreenPAK
- 2 つのコンデンサ
- 2 つのダイオード

GreenPAK の図

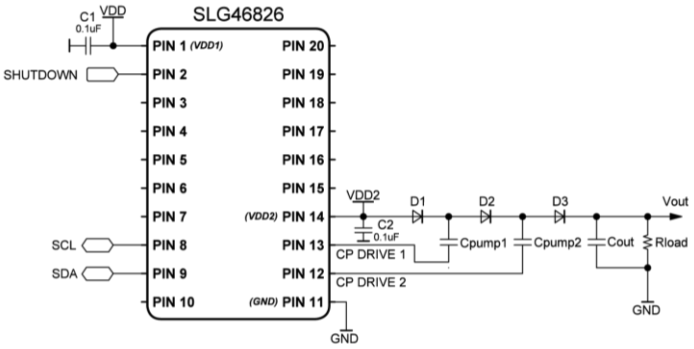


設計手順

1. 適切な出力の周波数が得られるように、OSC1 で分周比を設定します。
2. IO または I2C（利用可能な場合）により、シャットダウン機能が利用できるようにロジックを構成します。
3. VDD と CP_DRIVE の間にダイオード（D1）と Cpump を接続します。
4. D2 の陽極を D1 の陰極に接続します。
5. D2 の陰極と接地の間に Cout および Rload を並列で接続します。.

アプリケーション：2 ステージの チャージポンプ

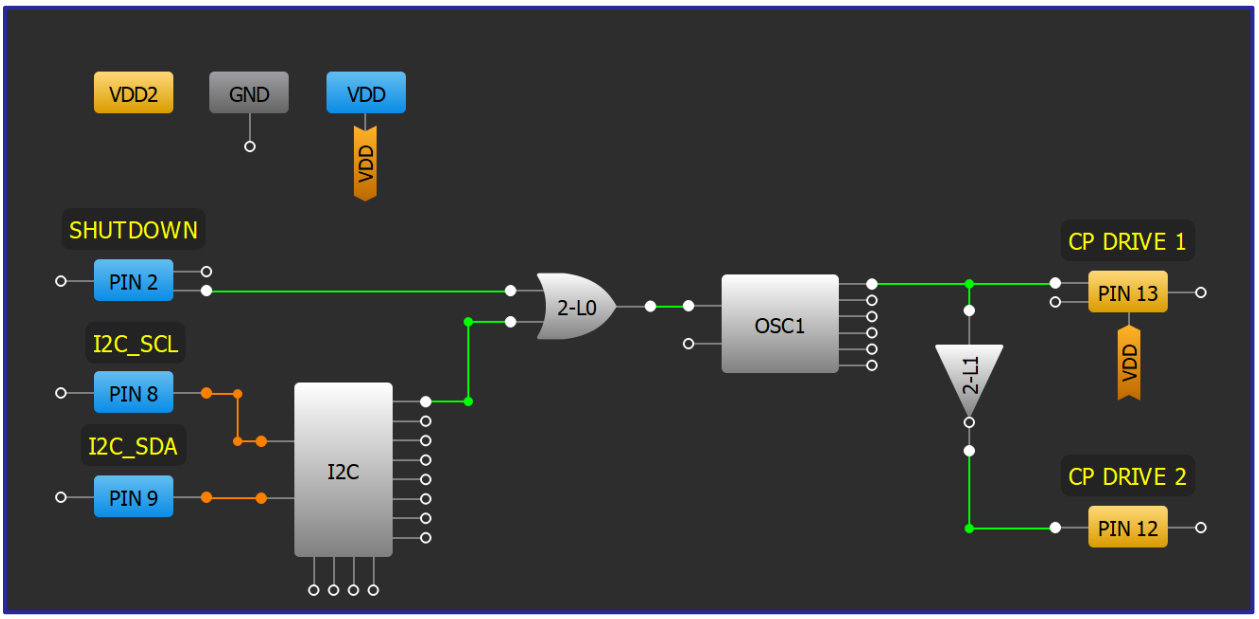
マルチステージのチャージポンプは、出力電圧 を入力電圧の 2 倍以上に押し上げることができ ます。最大のパフォーマンスを得るため、ショット キーダイオードを推奨します。



必要なコンポーネント

- 任意の GreenPAK
- 3 つのコンデンサ
- 3 つのダイオード

GreenPAK の図

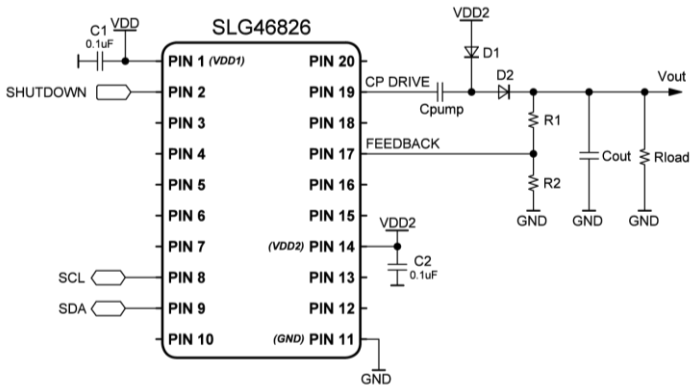


設計手順

1. アプリケーション：チャージポンプに従って、基本的なチャージポンプの設計を作成します。
2. インバータを追加するためのロジックを構成します。インバータを出力ピンとオシレータの間に接続します。
3. 追加のダイオードとコンデンサをチャージポンプの出力ステージと第 1 ステージの間に接続します。新しいコンデン サとダイオードは第 1 ステージのコンポーネントと同じタイプおよび値でなければなりません。.

アプリケーション: 出力調整付きチャージポンプ

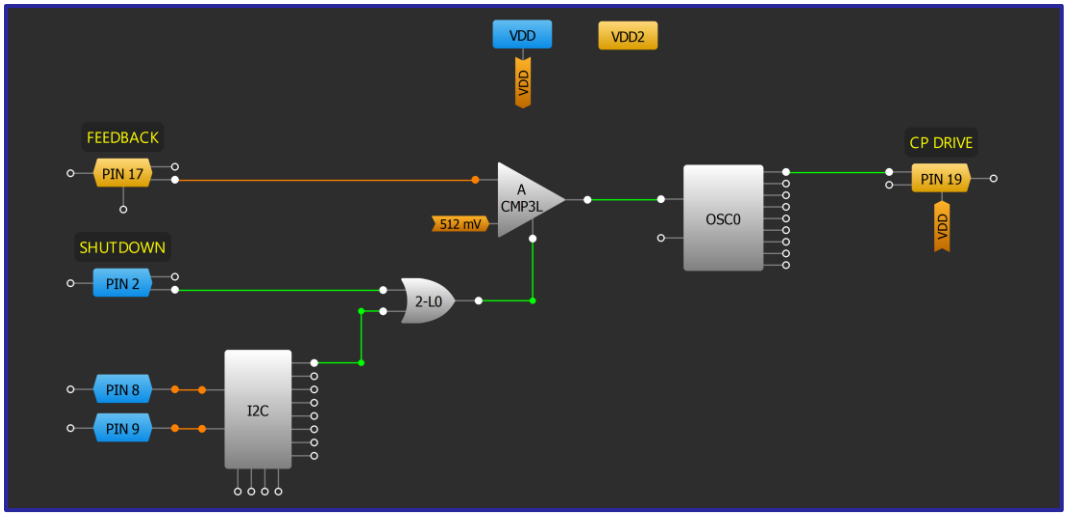
チャージポンプは DC-DC コンバータで、コンデンサを使用してエネルギー電荷を蓄え、異なる電圧レベルを生成します。出力調整回路を備えたこのチャージポンプは、I2C を介して出力電圧レベルを変更することができます。最高の性能を得るために、 ショットキーダイオードの使用を推奨します。



必要なコンポーネント

- ACMP つき任意の GreenPAK
- ２つのコンデンサ
- ２つのダイオード
- 2 つの抵抗

GreenPAK の図

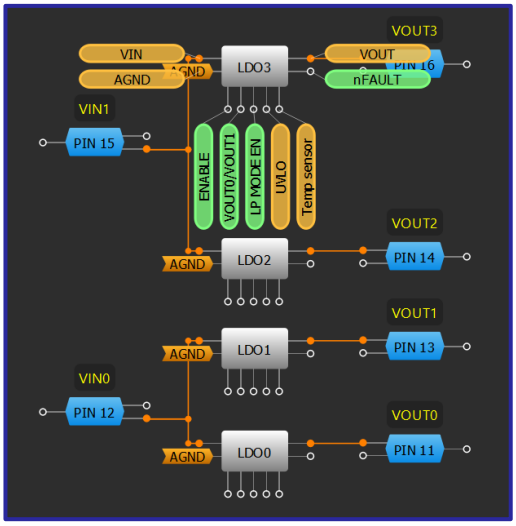


設計手順

1. 「アプリケーション:チャージポンプ」の手順で基本的なチャージポンプを作成します。
2. **ACMP3L** の **IN+ source** を **PIN17(FEEDBACK)**に設定し、**IN- source** を所望の Vref に設定します。
3. 2 つの抵抗 R1 と R2 を使用して分圧器を作成し、その出力を **PIN17** に接続します。
4. 出力電圧は、次の式を使用して計算できます。 $V_{out} = V_{ref} \cdot \frac{R1}{R2}$.
5. GreenPAK は、**ACMP3L** の **IN-source** の Vref を I2C で調整することにより、出力電圧を変更することができます。帰還抵抗分圧比を変更して、出力電圧を別の値に調整することもできます。

技法: LDO レギュレータ

この技法では、SLG46580、SLG46582、SLG46583、およびSLG46585 に内蔵された低ドロップアウト(LDO)レギュレータマクロセルについて説明します。



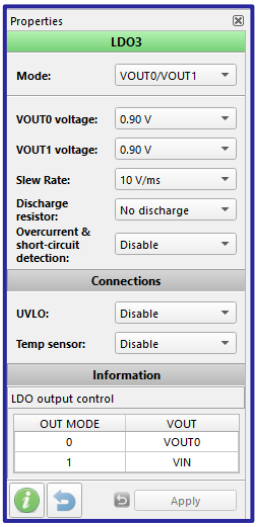
低ドロップアウト(LDO)レギュレータは、負荷インピーダンスの変化や電源電圧の変動にもかかわらず、最小限のドロップアウト電圧で電源の出力電圧を安定した値に維持します。これにより、エネルギー消費の大きいバッテリーに依存するポータブルデバイスや、周期的なリップルに対応しなければならない RF システムに役立ちます。一部の GreenPAK デバイスには、LDO レギュレータのマクロセルが装備されています。

各 LDO マクロセルは、0.90V～4.35V の範囲で 32 の出力電圧レベルを設定できます。また、2 つの異なる出力電圧(**VOUT1** と **VOUT2**)をプログラムすることができます。**VOUT1/VOUT2** 入力の状態によって、どちらの電圧レベルが出力されるかが選択されます。各出力電圧レベルの最小 VIN 値と VDD 値については、SLG46580 のデータシートを参照してください。

LP MODE EN 入力を LOW (デフォルト) に設定すると、ハイパワー (HP) モードが選択され、HIGH に設定すると、低消費電力 (LP) モードが選択されます。HP モードは最大の定格出力電流を処理できますが、LP モードは静止電流が小さく、小さな定格内でより高い効率を提供します。LDO が HP モードで安定するのは、各 LDO の VOUT に 2µF 以上のコンデンサが接続されているときに限定されます。

プロパティ ウィンドウで、各 LDO のスルーレートを変更して、ソフトスタートを設定することができます。各 LDO には、出力で 300Ω の放電抵抗を有効にするオプションがあります。各 LDO は、全て HP モード中に 210mA の過電流制限と、出力電圧が 0.5V を下回った場合に、電流を 20mA に制限する短絡検出を有効にすることもできます。

LDO で **UVLO** 接続を有効にすると、ACMP がその VIN を受けて、特定の低電圧ロックアウト (UVLO) しきい値 (ACMP IN- level) を下回った場合に LDO をシャットダウンするように設定されます。



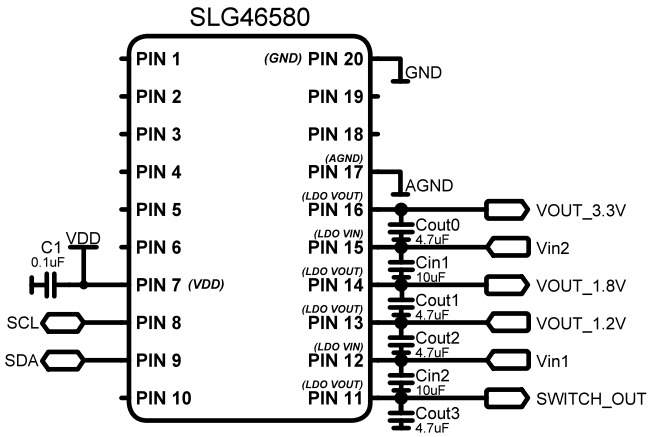
LDO レギュレータを設定する際には、デバイスの発熱による制約を考慮する必要があります。 デバイスの定格消費電力は、周囲温度 85°C で 0.6W です。**Temp sensor** 接続を有効にすると、ACMP2 は GreenPAK の温度センサーに接続され、適切にプログラムされた温度を超えた場合に LDO をシャットダウンし、冷却後に LDO を再起動できます。

デバイス仕様

- SLG46580/SLG46585:
- 4x LDO レギュレータ;
- Imax: HP モード = 150mA, LP モード = 100µA
- SLG46582:
- 2x LDO レギュレータ;
- HP モード Imax = 300mA, LP モード Imax = 200µA
- SLG46583:
- 1x LDO レギュレータ
- HP モード Imax = 600mA, LP モード Imax = 400µA

アプリケーション: フレキシブル電源アイランド

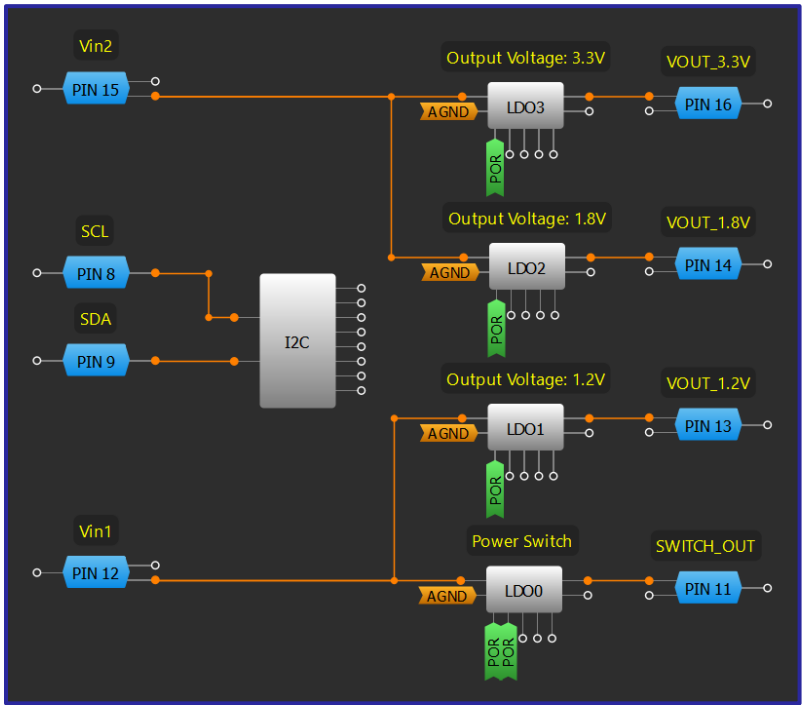
フレキシブル電源アイランドは、設計者が電源システムをシステム全体に分散できる電源領域の小さな「島」に分割するのに役立ちます。特定のシステム要件を満たすために、さまざまなレベルの安定化した電圧を供給できます。



必要なコンポーネント

- 内部 LDO をもつ任意の GreenPAK
- 7 つのコンデンサ

GreenPAK の図

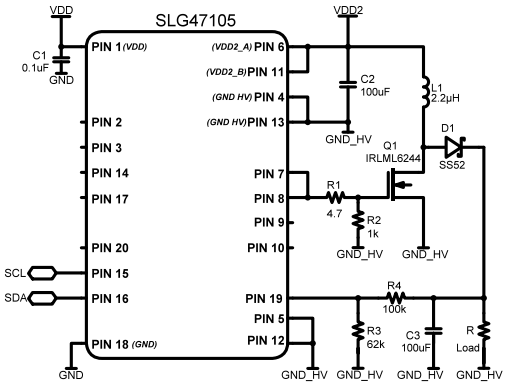


設計手順

1. 各 **LDO** チャンネルの **VOUT0 電圧**と **VOUT1 電圧**を設定します。
2. POR を各 **LDO** の **ENABLE** ピンに接続します。
3. LDO を「VOUT0/PWR switch」モードに設定し、**OUT MODE** ピンを HIGH に設定して Power Switch 出力として設定します。

アプリケーション: 昇圧コンバータ

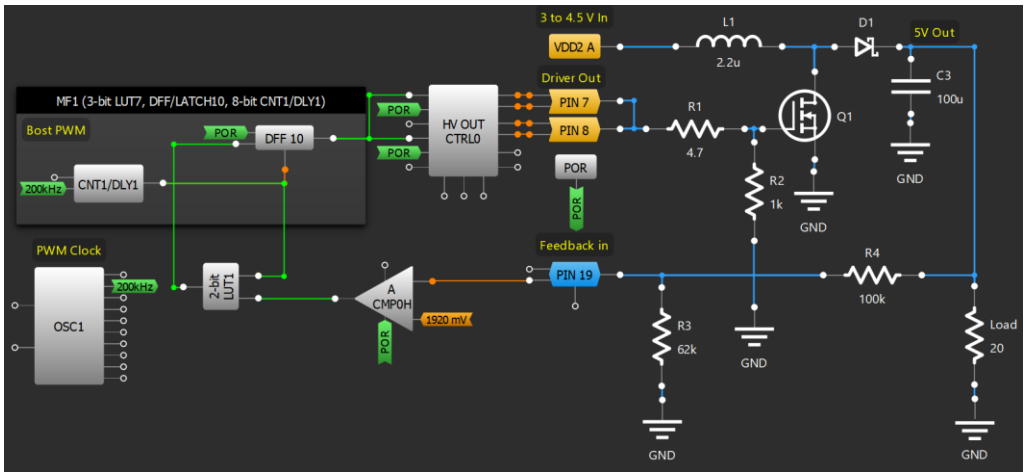
このアプリケーションでは、SLG47105 を昇圧コンバータのドライバとして使用します。このデザイン例では、VDD2 電圧を 3V から 5V に昇圧します.



必要なコンポーネント

- SLG47105
- MOSFET
- Schottky diode
- Inductor
- Capacitor　3 個
- 抵抗　4 個

GreenPAK の図

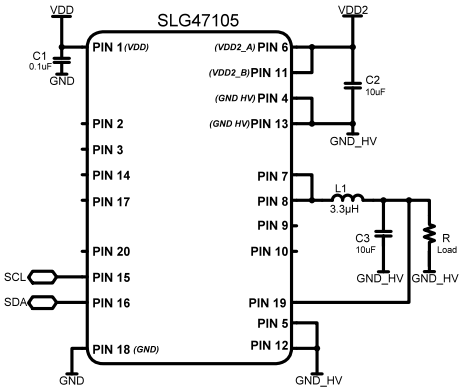


設計手順

1. GreenPAK 外部コンポーネントも含め、上記回路図に従って回路接続をします。外部コンポーネントの値は property から設定ができます。オレンジのゴムバンド（配線）についてはこの後設定します。
2. HV OUT CTRL0 の設定： Slew rate を “Fast for pre-drive” 、HV-OUTmode は “Half-Bridge”に設定します。
3. 端子設定：PIN7 と PIN8 の Output mode OE＝1 を“HIGH and LOW side on”に設定します。
4. ACMP0H の設定：IN- source の値を “1920 mV” に設定します。これにより出力電圧(昇圧後)を 5V に調整します。
5. 2bit LUT1 の設定：LUT を設定します(IN1＝1、IN0＝０のときのみ出力１)
6. CNT1/DLY1、OSC1 の設定： Delay モードに設定します。Cunter data は 51、Edge detect mode は Falling、Output polarity は Inverted に設定します。Connect DLY IN には OSC1 の Flex-Div OUT を接続します。OSC1 の Flexible divider 値は 125 に設定します。

アプリケーション: 降圧 Converter

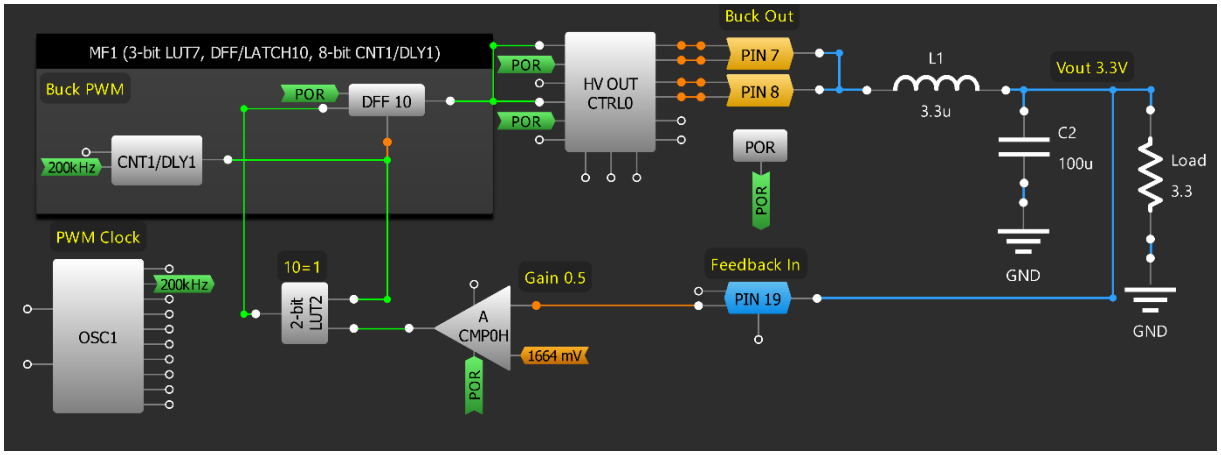
このアプリケーションでは、SLG47105 を降圧コンバータとして使用します。このデザイン例では、VDD2 電圧を 3.3V に降圧します。また、このデザインはアナログ PWM ブロックの構築方法も示しています。



必要なコンポーネント

- SLG47105
- インダクタ
- キャパシタ　3 個

GreenPAK の図

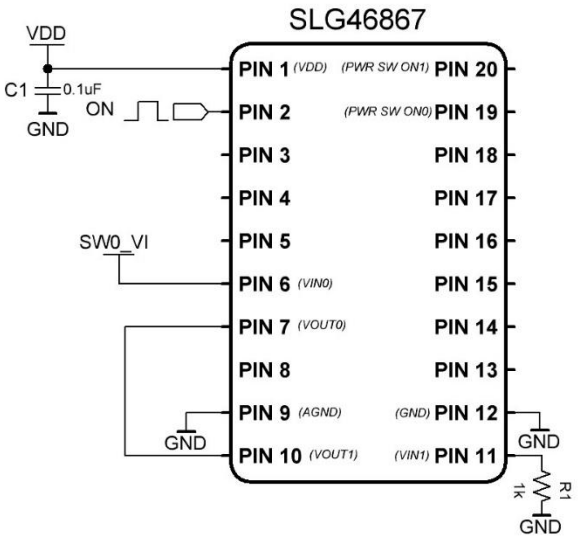


設計手順

1. GreenPAK 外部コンポーネントを含め、上記回路図に倣い回路接続をします。外部コンポ―ネントの値は property から設定ができます。L1 に PIN7,PIN8 を接続します。PIN19 は L1 と C2, C2 の他方は GND に、C2 と負荷は並列接続します。
2. HV OUT CTRL0 を設定： OE0/1 に POR を設定、Slew rate は“Fast for pre-drive” 、HV-OUTmode は “Half-Bridge”モードに設定します。
3. 端子設定：PIN7 と PIN8 の Output mode OE＝1 を“HIGH and LOW side on”に設定します。
4. ACMP0H の設定： POR と PWR up 端子に接続、有効にします。IN + gain は“x0.5”、 IN - source の値は “1664 mV” に設定し、出力を 3.3 V に調整できるようにします。出力は、2-bit LUT2 の IN0 に接続します。
5. CNT1/DLY1、OSC1 の設定： Delay モードに設定します。Counter data は 51、Edge detect mode は Falling、Output polarity は Inverted に設定します。Connect DLY IN には OSC1 の Flex-Div OUT を接続します。OSC1 の Flexible divider 値は 125 に設定します。
6. DFF10 の D 端子は POR に、CLK 端子は 2-bit LUT2 の出力に接続します。
7. DFF10 の出力先は HV OUT CTRL0 の IN0 と IN1 に接続します。

アプリケーション: Back-to-Back 接続 逆流防止回路

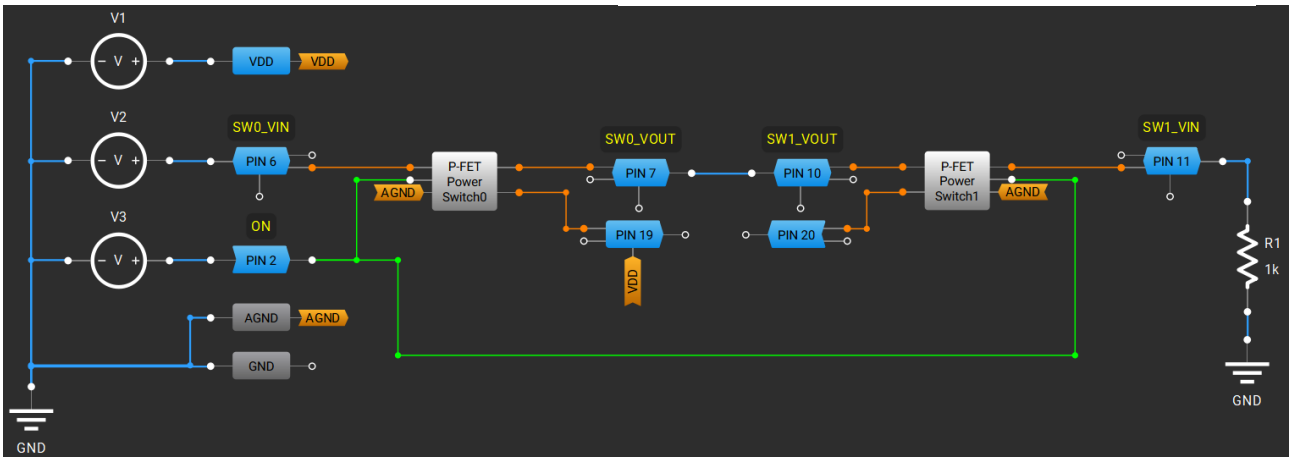
Back-to-back 接続逆流防止回路は電流の逆流を防ぐために使用されます。通常、2 つの MOSFET を逆向きに接続し、逆方向の電流を遮断する。この構成は、効率を高め、逆電流による損傷を防ぐため、電源、バッテリー駆動機器、DC-DC コンバーターなどで広く使用されています。

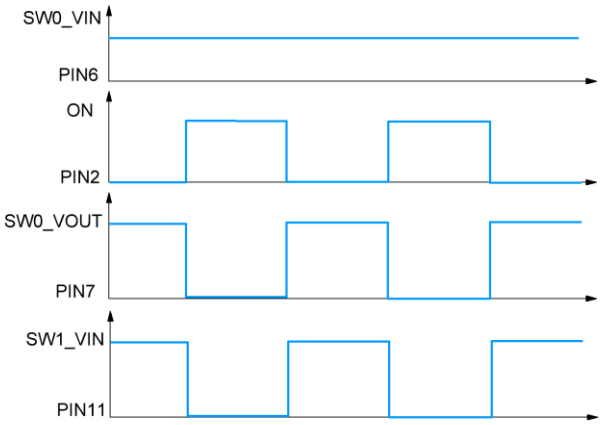


必要なコンポーネント

・ 2 つの PMOS を内蔵している GreenPAK

GreenPAK の回路図





設計手順

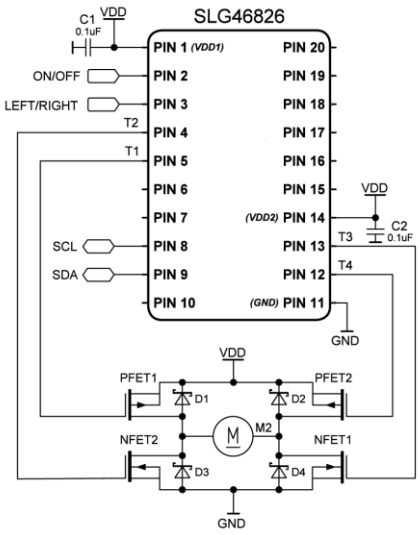
1. **P-FET Power Switch0** と **Switch1** の control mode を “Internal (ON input)”へ設定します。
2. **PIN 7** と **PIN 10 を接続します。**
3. **PIN 11** と抵抗を接続します。

第 8 章: Motor Control

この章では DC モータの制御について紹介します。主に、SLG47105 に内蔵された H-Bridge とそれに関わる電流、電圧制御のブロックについて説明します。

アプリケーション：H ブリッジコントロール

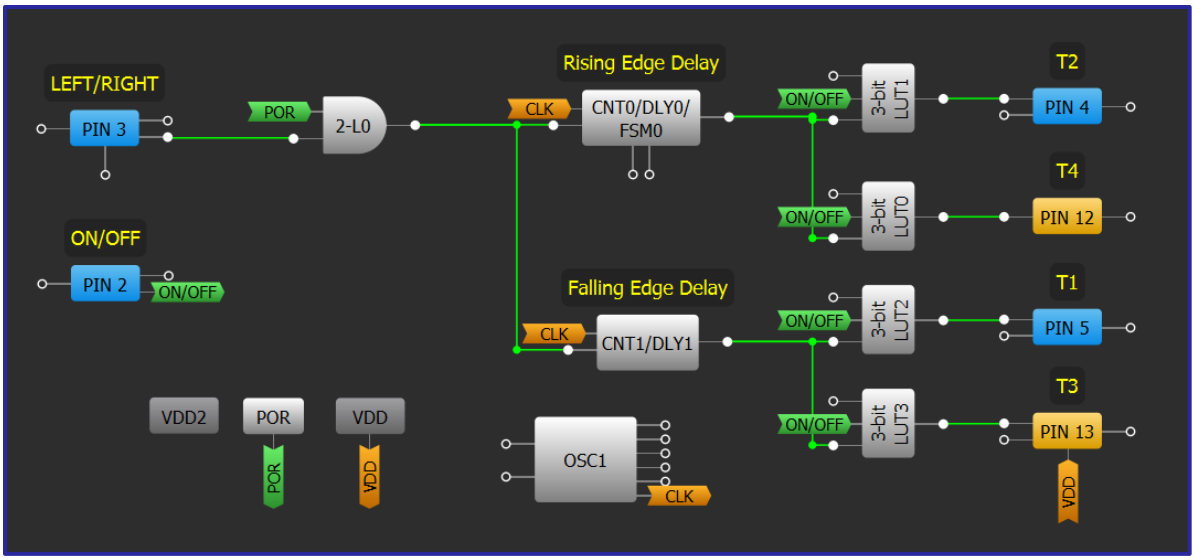
H ブリッジは 4 つのトランジスタで構成され、負荷にかかる電圧の極性を反転させる回路です。DC モータの制御によく使われます。



必要なコンポーネント

- 任意の GreenPAK
- 4 つのトランジスタ
- 4 つのダイオード

GreenPAK の図

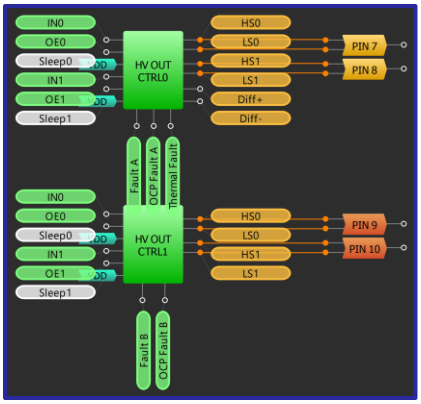


設計手順

1. 入力と出力端子を選んで、設定します。
2. 技法：CNT/DLY の精度の最適化に従って遅延ブロックを追加します。
3. . 技法：LUT マクロセルを使用した標準ロジックの構成に従って各出力用に LUT を追加し、構成します。

技法: HV OUT CTRL ブロックの使用

*この技法では、SLG47105V での **HV OUT CTRL ブロック**の使い方について説明します。*



HV OUT CTRL0/1

SLG47105V は、2 つの HV OUT CTRL マクロセル、すなわち **HV OUT CTRL0** および **HV OUT CTRL1** で構成されています。 各マクロセルは、2 つの単方向 DC モータまたは 1 つの双方向 DC モータを駆動するために使用できます。 **また、HV OUT CTRL0/1** の両方によりステッピングモータを駆動できます。

HV OUT CTRL0/1 を有効にするには、**Sleep 0/1** をアクティブ LOW に接続します。各 Sleep ピンは個別にアクティブ化できます。単方向に駆動するには、**HV OUT mode** を「ハーフブリッジ」として選択し、双方向モータの場合は「フルブリッジ」を選択します。「フルブリッジ」モードでは、**Mode control** を「IN-IN」または「PH-EN」として選択します。表 2 に、「ハーフブリッジ」制御ロジックを示します。表 表 は、それぞれ「IN-IN」および「PH-EN」モード制御について記載しています。 **Pin7/8/9/10** (デフォルトでは Hi-Z)は、PWM を使用してモータ速度を制御するために使用されます。これらのピンは、外部でプルアップ/プルダウンできます。 「フルブリッジ」モードでは、 **Pin7/9** と **Pin8/10** を外部で並列接続できます。

PWM0/1 を使用して「IN-IN」**mode control** で DC モータを制御するには、**IN1** を高速減衰モードではアクティブ LOW 信号に接続し、低速減衰モードではアクティブ HIGH 信号に接続します。 **IN0** を **PWM0/1** 出力に接続します。高速減衰モードでは、誘導電流が瞬時に減少し、モータをゼロ速度に向かって惰走させます。低速減衰モードでは、誘導電流がゆっくりと減少し、急速に（モータが）減速します。

表 2 フルブリッジ モードの IN-IN モード ロジック

Sleep 0/1	IN0	IN1	Pin 7/9	Pin 8/10	動作
1	X	X	Hi-Z	Hi-Z	オフ
0	0	0	Hi-Z	Hi-Z	惰力走行
0	0	1	L	H	反転
0	1	0	H	L	順転
0	1	1	L	L	ブレーキ

表 3 ハーフブリッジ モードロジック

Sleep 0/1	OE	IN0/1	Pin 7/8	動作
1	X	X	Hi-Z	オフ
0	0	X	Hi-Z	オフ 惰力走行)
0	1	0	L	ブレーキ
0	1	1	H	順転

表 4 フルブリッジ モードの PH-EN モードロジック

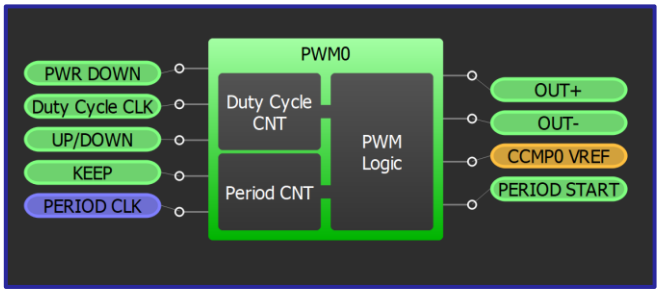
Sleep 0/1	Decay	EN/PWM	PH/Direct	Pin 7/9	Pin 8/10	動作
1	X	X	X	Hi-Z	Hi-Z	オフ (惰力走行)
0	0 (fast)	0	X	Hi-Z	Hi-Z	惰力走行
0	1 (slow)	0	X	L	L	ブレーキ
0	X	1	0	H	L	順転
0	X	1	1	L	H	反転

異常が発生すると、 **Fault A/B** ピンは HIGH になり、**HV 出力 CTRL0/1** はディセーブル状態になります。Fault ピンが LOW になると、通常の動作に戻ります。**Fault A** と **Fault B** は、VDD2_A と VDD2_B に関わる全ての障害信号をそれぞれ別々に構成しています。過電流状態が発生すると、**OCP Fault A / B** は HIGH になります。OCP デグリッチ タイムは、**Pin 7/8/9/10 で**有効にできます。**OCP デグリッチタイムイネーブル**は、**Pin7/8** と **Pin9/10** の間で共有されます。OCP の再試行時間はユーザーが選択でき、ピンごとに異なります。チップの温度が安全限界を超えると、 **Thermal Fault は HIGH** に変わります。VDD2_A と VDD2_B には、別々の UVLO(低電圧ロックアウト)イネーブルがあります。

技法: 通常モードでの SLG47105 PWM ブロックの使用

この技法は、SLG47105 で使用可能な PWM ブロック用です。

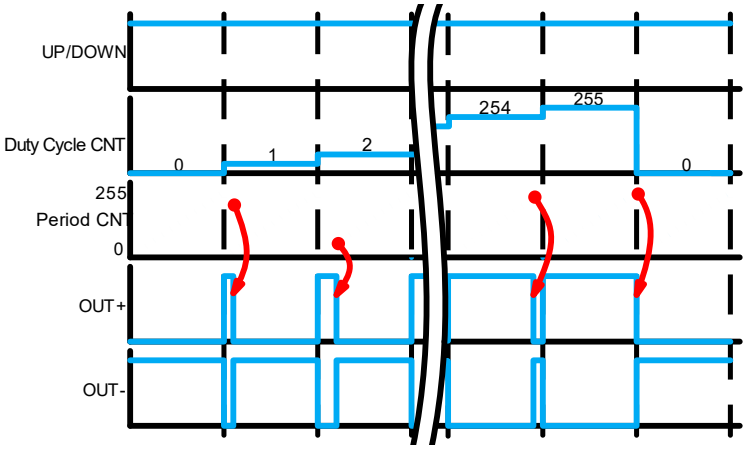
PWM は、DC モータ制御、LED 輝度制御、およびその他のアプリケーションで一般的に使用されます。SLG47105 には、より高い電圧レベルの専用 PIN を処理するための高度な PWM ブロックが装備されています。 PWM ブロックは、「技法:PWM モードで DCMP/PWM マクロセルを使用」で説明されているように、他の GreenPAK にすでに実装されています。しかし、SLG47105 の 2 つの PWM ブロックは、「通常モード」のときに動作に関わるカウンタを統合し、より高度な設定を含みます。



ブロックに含まれる最初の 8 ビットカウンタは**PWM 周期 CNT** で、PWM 信号の周波数を設定し、0～255 などをカウントします。 PWM 出力には「OUT+」と「OUT-」の 2 種類があります。 OUT+は周期の開始時にロジック HIGH であり、**PWM 周期カウンタ**が**デューティサイクル値**に達すると、下図に示すように PWM 期間が終了するまで出力はロジック LOW に変わります。**OUT+**は正の PWM 出力で、**OUT-**は OUT+を反転した負の PWM 出力で、**デッドバンド時間**が定義されている場合はその分シフトされます。どちらもレジスタ設定によって出力を反転させることができます。

通常モード(この技法で説明している)では、**デューティサイクルソース**は「デューティサイクル CNT」に設定されます。**デューティサイクル CNT** と呼ばれる 2 番目の 8 ビットカウンタは、**アップ/ダウン**入力に応じて次の PWM 期間の**デューティサイクル値**を増減します。**デューティサイクル CLK** は、デフォルトでマトリックスからの外部クロックです。立ち上がりエッジによって**デューティサイクル値が**変化します。 また、**周期カウンタ**オーバーフローまたはオーバーフローの 2 番目または 8 番目のパルスごとにクロックされるように設定することもできます。

PWM ブロックの分解能はデフォルトで 8 ビットですが、代わりに 7 ビット分解能を選択して、より高い PWM 周波数を可能にすることができます。 PWM デューティサイクルは、8 ビット分解能で 0.4%、7 ビット分解能で 0.8%のステップで変化します。デューティサイクルは、真の 0%から真の 100%まで変化することができます。PWM は**初期デューティサイクル値**から始まります。ブロックには、デューティサイクルの変化の方向を定義する**アップ/ダウン**の内部接続があります。HIGH に設定するとデューティサイクルは増加し、LOW に設定すると減少します。



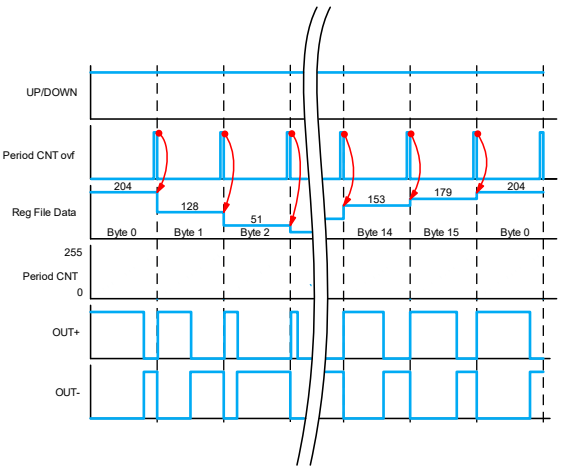
キープ/ストップ接続は、デューティサイクルを保持する(「キープ」設定)か、HIGH に設定されているときにデューティサイクルと OUT+および OUT-出力を一定に保持する(「ストップ」設定)ように選択できます。**連続/自動停止モード**は、PWM 出力デューティサイクルがフルレンジ値(デフォルト設定)に達するとオーバーフローする「連続」、またはデューティサイクルの 0%または 100%に達すると PWM 出力が停止する「自動停止」に設定します。「自動停止」を選択すると、「境界 OSC を無効にする」オプションを有効にすることができます。 これにより、PWM セルで使用される発振器を、デューティサイクルの 0%または 100%に達したときに自動的に停止することができます。

技法: プリセット レジスタ モードでの SLG47105 PWM ブロックの使用

この技法は、SLG47105 で使用可能な PWM ブロック用です。 ブロックがどのように機能するかの詳細については、 「技法:通常モードでの SLG47105 PWM ブロックの使用」を参照してください。

前の技法では、SLG47105 PWM ブロックを「通常モード」で使用し、デューティサイクルソースを「デューティサイクル CNT」に設定する方法について説明しました。この技法では、代わりに、デューティサイクルが 16 の定義済み値(Reg ファイル)を循環する「プリセットレジスタモード」でブロックを使用する方法について説明します。 「プリセットレジスタモード」を使用すると、モータ制御用の非線形 PWM パターン(正弦波または対数)が可能になります。

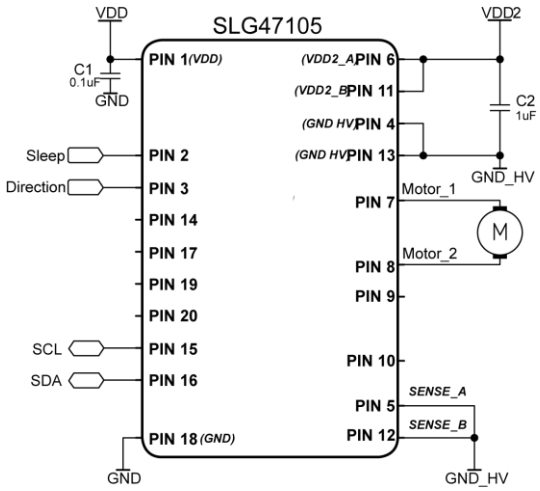
選択可能なプリセットレジスタは、16 種類の PWM デューティサイクル値を決定するために用意されています.デューティサイクル CLK は、マトリックスまたは PWM 周期 CNT ovf (オーバーフロー)からのクロックに選択することができます。**デューティ サイクル CNT CLK** 入力のクロックは、周期 CNT に対してどのレジスタの値を適用して比較するかを変更します。レジスタファイルは、2 つの PWM マクロセル間で共有されます。 16 バイトすべて、最下位 8 バイト、または最上位 8 バイトのいずれかを使用できます。最初のバイトは、各設定の一意の範囲によって制限されます。



内部接続は、通常モードと同様の機能を持ちますが、8 ビット カウンタではなく 16 バイト構造に適用されます。**アップ/ダウン**の極性によって、次のレジスタ (HIGH) が適用されるか、前のレジスタ (LOW) が適用されるかが決まります。**キープ/ストップ**は同じように動作しますが、レジスタのシーケンスを停止します。

アプリケーション: 定電圧ブラシ付きモータドライバー

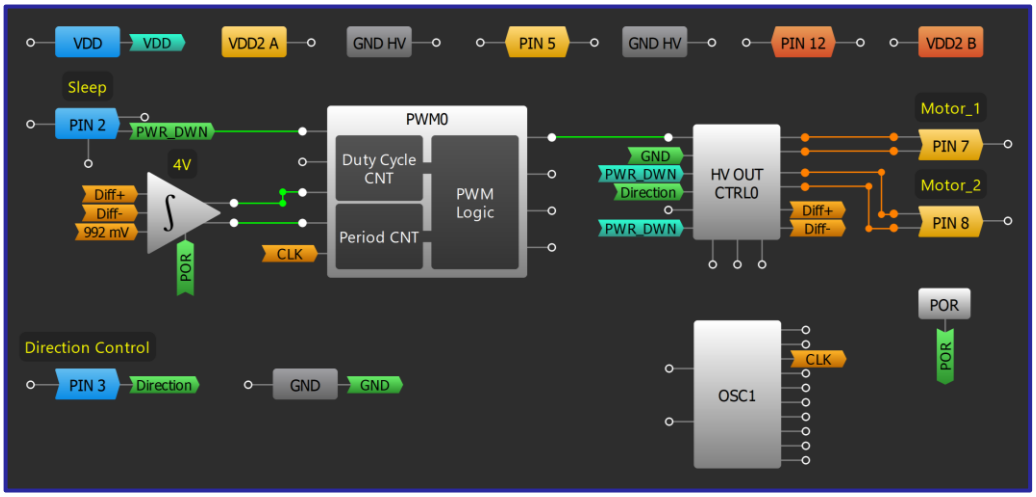
ブラシ付き DC モータに一定の電圧をかけると、一定の速度が維持されます。この設計では、積分器とコンパレータを備えた差動増幅器が PWM ブロックを制御して、負荷に印加する電圧を調整します。



必要なコンポーネント

- SLG47105V
- ブラシ付き DC モータ

GreenPAK の図

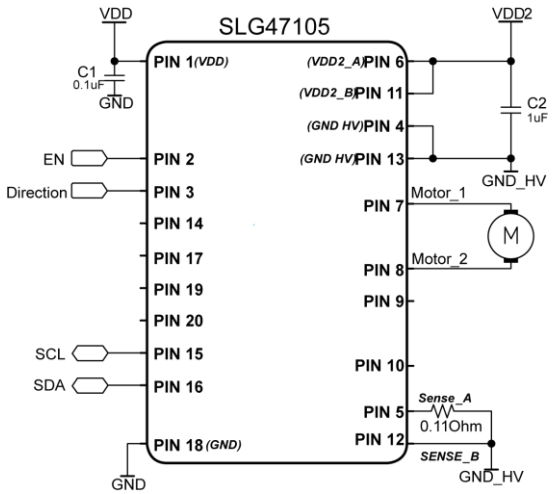


設計手順

1. PIN2 で HV OUT CTRL0 を有効にし、モードを「フルブリッジ」に、モード制御を「PH-EN」に設定します
2. モータの方向を変更するには、PIN3 を HV OUT CTRL0 の PH に接続します
3. 積分器と比較器を備えた差動増幅器を有効にし、積分器の基準電圧を所望のしきい値 (Threshold = V_REF x 4) に選択します。
4. 積分器と比較器を備えた差動増幅器を正しく動作させるには、PIN2 を介して PWM0 を有効にし、Duty Period CLK を「OSC1」に設定します。積分器が正しく動作するためには、PWM 周波数は 44 kHz 以上である必要があります。 UPWARD と Equal 出力をそれぞれ PWM0 の UP/DOWN と Keep に接続します
5. PWM0 分解能を「8 ビット」に、デューティ サイクルソースを「デューティ サイクル CNT」に、デューティ サイクル CLK を「周期 CNT ovf/8」に、初期デューティ サイクル値を「50%」に設定します
6. PWM0 OUT+ を HV OUT CTRL0 の EN に接続して、モータを一定速度で駆動します

アプリケーション: 定電流ブラシ付きモータドライバー

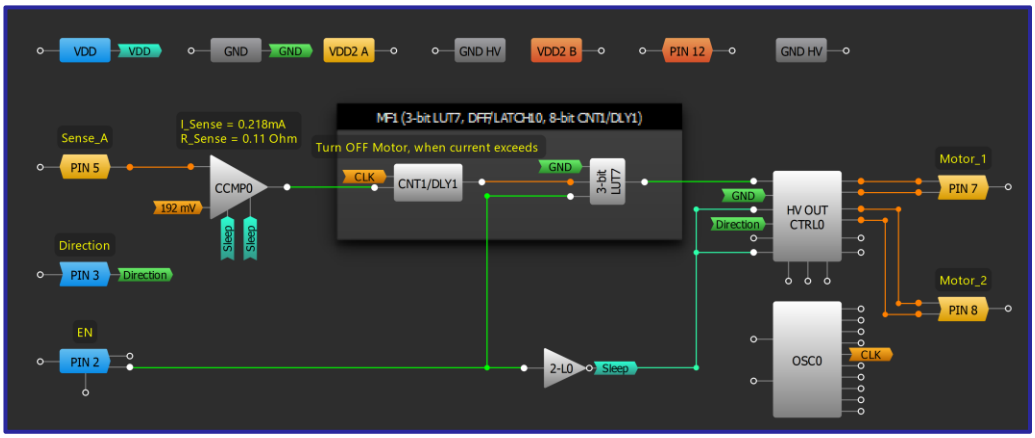
ブラシ付き DC モータに一定の電流を流すと、一定のトルクが維持されます。この設計では、電流検出コンパレータ (CCMP0) を使用して検出抵抗を流れる電流を制限し、電流が指定された制限値を超えたときにモータをオフにします。



必要なコンポーネント

- SLG47105V
- ブラシ付き DC モータ
- １つの抵抗

GreenPAK の図

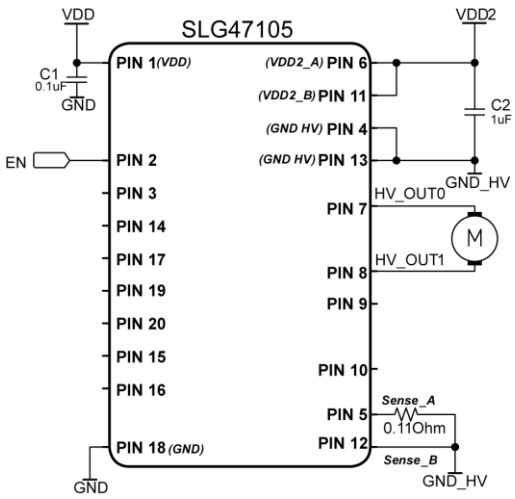


設計手順

1. PIN7 と PIN8 の間にブラシ付き DC モータを接続し、PIN5 と GND の間にセンス抵抗を接続します。
2. 反転した PIN2 信号をスリープ 0/1 に接続して、HV OUT CTRL0 を有効にします。
3. HV OUT CTRL0 のモードを「フルブリッジ」、モード制御を「PH-EN」に設定します。
4. モータの方向を変更するために、ピン 3 を HV OUT CTRL0 の PH に接続します。
5. Sleep CTRL を「Auto」に変更して、CCMP0 を有効にします。
6. CCMP0 の IN ソースを「192mV」に選択して、電流を 0.218mA に制限します。
7. CNT1/DLY1 を構成して、電流が指定の制限値を超えたときにモータを 100 ミリ秒間オフにします。
8. 3 ビット LUT7 を構成して、電流が範囲内にあり、PIN2 が HIGH の場合にのみモータをオンにします。
9. 3 ビット LUT7 の 出力を HV OUT CTRL0 の EN に接続します。

アプリケーション: PWM チョッパーを使用した定電流

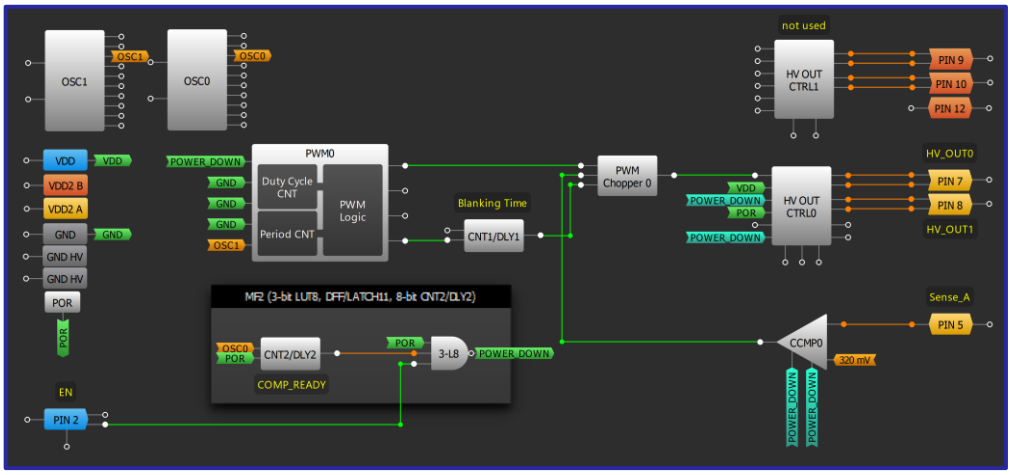
ブラシ付き DC モータに一定の電流を流すと、一定のトルクが維持されます。このアプリケーションでは、PWM チョッパー ブロックで電流を制限する方法を示します。



必要なコンポーネント

- SLG47105V
- ブラシ付き DC モータ
- 1 つの抵抗

GreenPAK の図

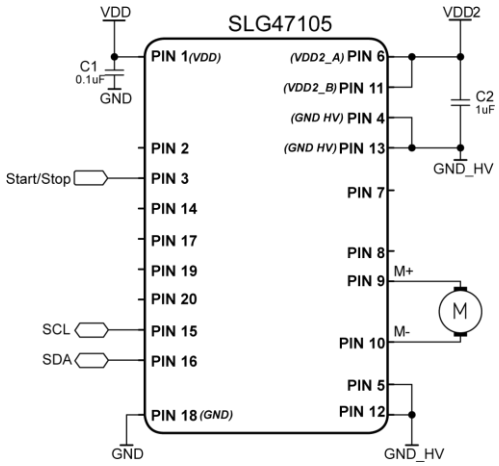


設計手順

1. PIN7 と PIN8 の間にブラシ付き DC モータを接続し、ピン 5 と GND の間 にセンス抵抗を接続します。
2. HV OUT CTRL0 のモードを「フルブリッジ」、モード制御を「PH-EN」に設定します。
3. Sleep CTRL を「Auto」に変更して、CCMP0 を有効にします。
4. CCMP0 の IN ソースを「320mV」に選択して、電流を 0.36mA に制限します。
5. PWM0 ブロックを追加し、初期デューティ サイクルを 230 に設定し、OSC1 分周器を調整します。
6. CNT1/DLY1 を立ち下がりエッジ遅延として構成し、ブランキング時間を設定します。
7. デューティ サイクル チョッパーを作成しモータ電流を制限するように PWM チョッパー 0 を追加し、PWM0、CNT1/DLY1、および CCMP0 と適切に接続します。
8. ピン 2 を有効化ボタンとして追加し、モータおよび内部モータ制御ブロックを開始/停止します。
9. 3 ビット LUT8 の出力を HV OUT CTRL0 の EN に接続します。

アプリケーション: ソフト ON/OFF による単方向 DC モータ制御

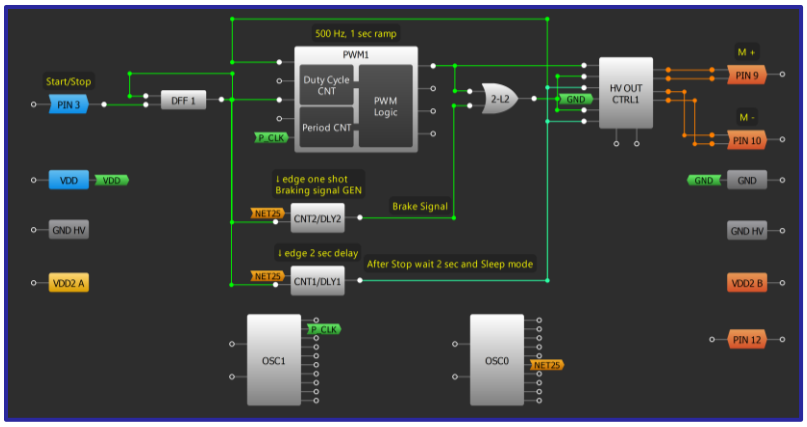
ソフト ON/OFF を使用して、ブラシ付き DC モータの始動電流と負荷トルクを減らすことができます。このアプリケーションは、機械的要素に単一方向を許可し、運転または停止する単方向 DC モータ用です。負荷トルクは、始動および停止時にモータシャフトのトルクを超えてはならないため、この調整は重要です。

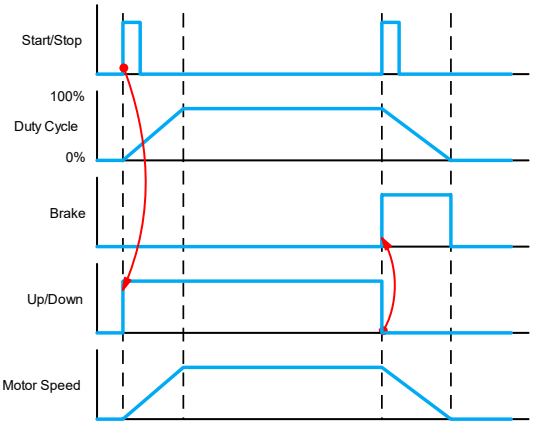


必要なコンポーネント

- SLG47105
- 単方向ブラシ付きモータ
- 2 つの容量

GreenPAK の図



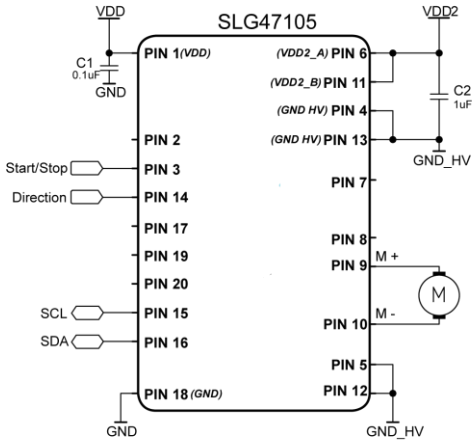


設計手順

1. PIN9 と PIN10 の出力モードを「ハイサイドとローサイド」に設定します。HV OUT CTRL1 を「ハーフブリッジ」HV OUT モードに設定します。
2. PWM1 ブロックの Duty Cycle CLK を「Period CNT ovf/2」に設定します。周期 CLK を「Ext.Clk」に設定します。 OSC1 の Flex-DIV OUT を PWM1 周期クロック入力に接続します。OSC1 プロパティで分周比の値を設定します。
3. DFF1 の初期値を High に設定し、Q の出力値を「Inverted (nQ)」に設定します。 DFF の出力を PWM1 の UP/DOWN 入力に接続します。
4. CNT1/DLY1 をデザインに追加します。それを反転出力の「立ち下がりエッジ遅延」に設定します。この遅延からの信号は、PWM1 ブロックを無効にし、停止信号が来た後に HV OUT CTRL1 のスリープモードを有効にします。
5. 2 ビット LUT2 は、HV 出力を Hi-Z にするとともに停止信号を形成するために使用されます。

アプリケーション:ソフト ON/OFF による双方向 DC モータ制御

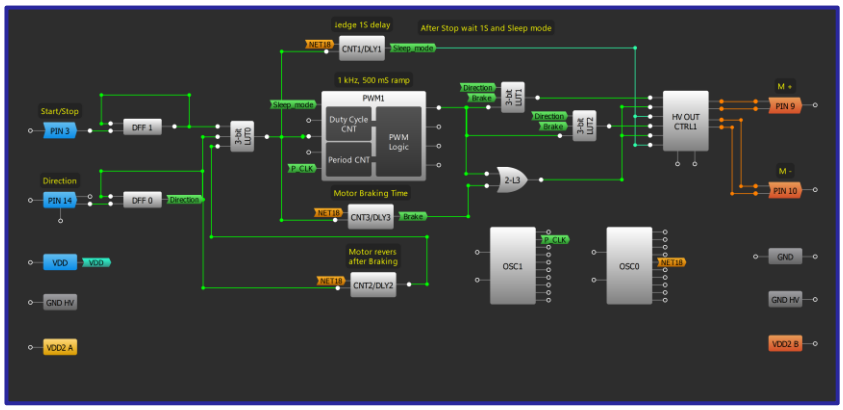
ソフト ON/OFF を使用して、ブラシ付き DC モータの始動電流と負荷トルクを減らすことができます。このアプリケーションは、機械的要素の逆方向の動きを可能にする双方向モータ用です。逆方法の運転になる前に、メカニズムを停止するためにモータのブレーキトルクが負荷トルクを超える必要があります。

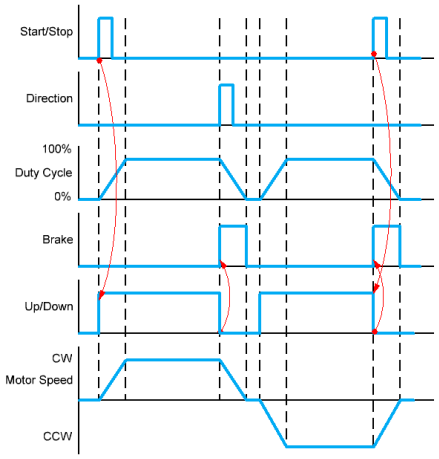


必要なコンポーネント

- SLG47105
- 双方向ブラシ付きモータ
- 1 つの容量

GreenPAK の図

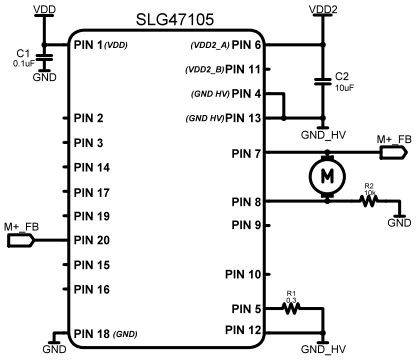




設計手順

1. PIN9 と PIN10 の出力モードを「ハイサイドとローサイド」に設定します。 HV OUT CTRL1 を「ハーフブリッジ」HV OUT モードに設定。
2. PWM1 ブロックの Duty Cycle CLK を「Period CNT ovf/2」に設定します。周期 CLK を「Ext.Clk」に設定します。 OSC1 の Flex-DIV OUT を PWM1 周期クロック入力に接続します。 OSC1 プロパティで分周の値を設定します。
3. DFF1 の初期値を High に設定し、Q の出力極性を「Inverted(nQ)」に設定します。 DFF0 も同じ設定です DFF の出力を 3 ビット LUT0 に接続します。
4. CNT1/DLY1 を反転出力の「立ち下がりエッジ遅延」として構成します。この遅延からの信号は、PWM1 ブロックを無効にし、停止信号が来た後に HV OUT CTRL1 のスリープモードを有効にします。
5. CNT2/DLY2 を「立ち上がり/立ち下がりの両エッジ」遅延に設定して、反転前にモータを停止する時間遅延を設定します。
6. CNT3/DLY3 を「立ち下がりエッジ ワンショット」に設定して、カウンターの出力を 3 ビット LUT1 と 3 ビット LUT2 に接続します。

アプリケーション: プッシュスタート/ホールドストップ

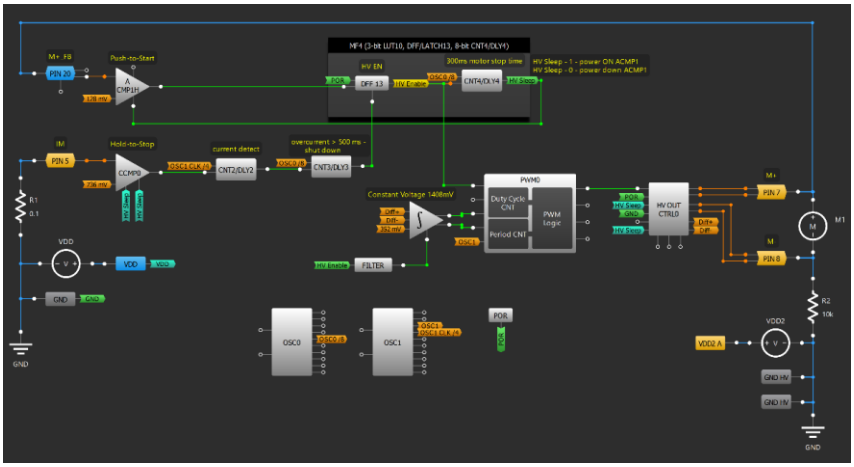


このアプリケーションでは、SLG47105 をプッシュスタート/ホールドストップ機能を備えたブラシ付き DC モータのドライバとして使用します。この機能は、おもちゃ、工具などに使用できます。

必要なコンポーネント

- SLG47105
- ブラシ付き DC モータ
- 抵抗 2 個

GreenPAK の図

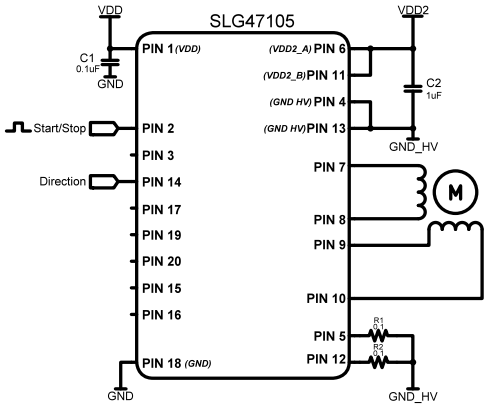


設計手順

1. ブラシ付き DC モータとの接続:PIN7 と PIN8 間にモータ、および 10kΩ のプルダウン抵抗を接続します。PIN7 と PIN8 の出力はそれぞれ High/Low サイドです。センス抵抗は PIN5 から GND の間に接続します。
2. HV OUT CTRL0 の Sleep0/1 をローアクティブ信号の CNT4/DLY4 の出力に接続します。HV OUT CTRLL0 の HV OUT モードは「Full Bridge、Mode control を「PN-EN」に設定します。POR を Decay に、GND を PH に接続します。
3. ホールドストップ機能のための電流センスコンパレータ CCMP0 の設定：CCMP0 の Sleep CTRL を Auto に設定し有効にします。IN-source の閾値はホールドストップ機能に適した値(ここでは 736 mV)に設定します。
4. CNT2/DLY2 を Delay モードとして設定し、その出力を CNT3/DLY3(Delayed edge detect モード)に接続します。CNT3/DLY3 の出力はこの後作成するプッシュスタート回路の DFF13 の nRerset に接続し、過電流状態(ホールドストップ状態)が 500ms 継続した後にシステムがシャットダウンするようにします。
5. プッシュスタート機能のための設定：PIN20 をアナログ入力/出力として構成し、ACMP1H の IN+source に設定、IN-source の閾値は 128 mV を選択し、ACMP1H の出力は DFF13(出力極性-反転(nQ)、初期極性-LOW)とします。
6. モータ停止期間の設定：CNT4/DLY4 を Delay モードに設定し、300ms のモータ停止時間(HVSleep 信号)を確保します。
7. 定電圧制御および PWM 信号生成のための構成：PWM 0 と差動アンプ Differential Amplifier(基準電圧 352 mV)、DFF13(Filter を介して差動アンプの PWR UP に接続)、HV OUT CTRL0 と接続することで、ブラシ付き DC モータの PWM 制御、定電圧制御ができるようにします（アプリケーション：定電圧ブラシ付き DC モータドライバを参照。

アプリケーション: バイポーラステッピングモータドライバ

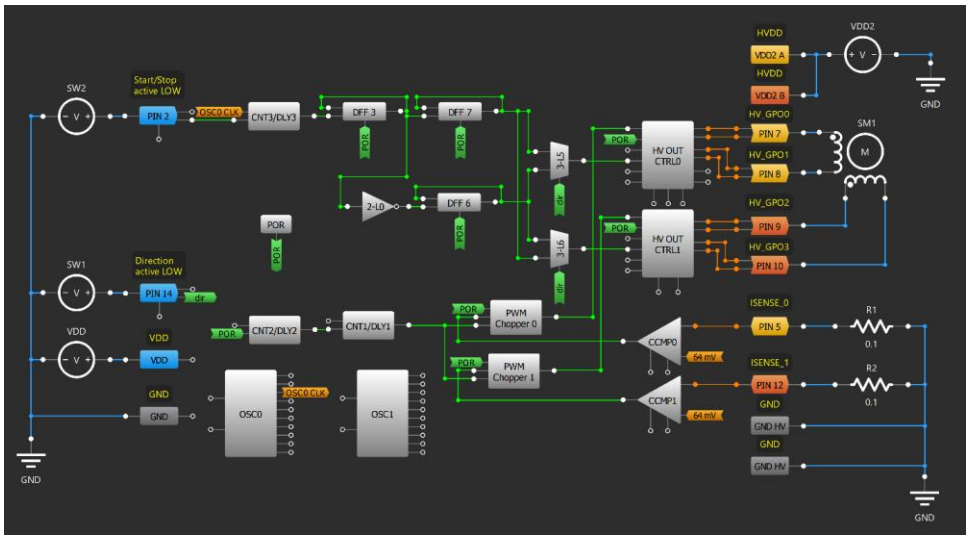
このアプリケーションのデザイン例では SLG47105 をステッピングモータドライバとして使用します。両方向のフルステップモードにて設計しており、PWM Chopper コンポーネントを使用した定電流制御の方法を示します。



必要なコンポーネント

- SLG47105
- ステッピングモータ
- 抵抗 2 個

GreenPAK の図



設計手順

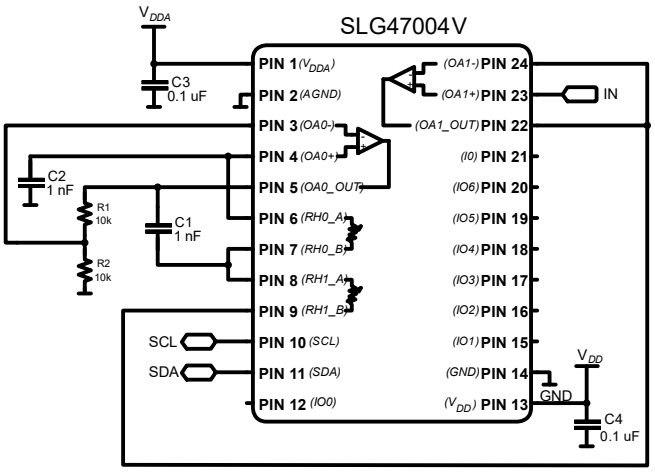
1. ステッピングモータを接続: 第 1 巻線は HV OUT CTRL0 の出力 PIN 7 と PIN8 の間、第 2 巻線は HV OUTCTRL1 の出力 PIN 9 と PIN10 の間に接続します。センス抵抗は PIN5 と GND の間、及び PIN12 と GND の間に接続します。
2. HVOUTCTRL0/1 の設定：Sleep0/1 は Low アクティブの信号線、Full Bridge mode かつ PN-EN mode に設定します。
3. CCMP0/1 の設定：Sleep CTRL を”Auto”し有効化、入力 IN-source の閾値を 64mV、Gain を x8 に設定することで、電流の閾値を 80mA とします。
4. CNT2/DLY2 を設定：Reset Counter モードに設定し、出力先は Delay モードに設定した CNT1/DLY1 の DLYIN に設定。これにより、Blanking 時間を構成できます。
5. PWM0/1 Chopper を追加し、入力端子に POR、CNT1/DLY1、CCMP0/1 を接続。これにより Duty Cycle Chopper とモータ電流制限ができます。出力は、HVOUTCTRL0/1 の EN 端子に接続します。
6. モータのスタート/ストップ信号となる PIN2、回転方向を決める PIN14 の設定：設定を Digital Input とします。
7. 遅延時間を作るための回路を構成：DFF3、DFF7、DFF6 を出力極性：nQ、初期極性は Low に設定、遅延時間をつかさどる CNT2/DLY2、2bitLUT(Invertor 設定)、LUT 5 と LUT6 は Mux とし、回路を図のように接続します。この回路の出力は HV OUT CTRL0/1 の PH に接続します。これによりモータ回転のスタート/ストップ、向きをコントロールする回路ができます。

第 9 章: Advanced Analog Features

この章では、AnalogPAK のコンポーネントを使用してアナログ機能を管理するアプリケーションについて説明します。
アプリケーションには、以下の要素を使用する最も一般的な回路トポロジが含まれます。
オペアンプ、デジタル可変抵抗器、チョッパーACMP など。

アプリケーション: オペアンプを使った可変アクティブフィルタ

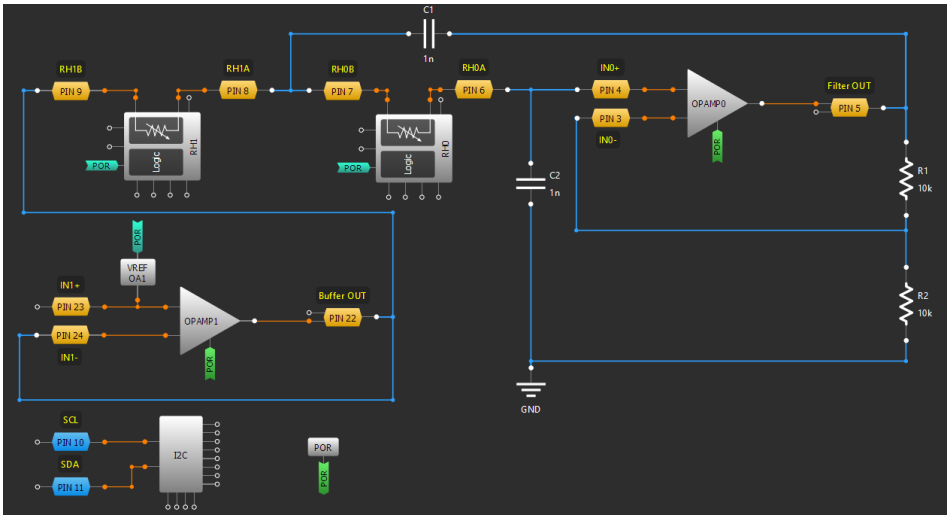
様々な信号源（センサなど）からの信号を 1 つの ADC で検知できるアプリケーションは数多くあります。このようなシステムでは各信号源が独自のフィルタ要件（カットオフ周波数など）を持つ可能性があるため、各チャネル用のアナログフィルタを備えたマルチプレクサが必要です。このデザインでは 1 つのフィルタが複数のアナログ入力に対応し、それぞれに適当なカットオフ周波数を提供します。I2C のマスタは可変抵抗のレジスタにデータを書き込み、フィルタのカットオフ周波数を調整できます。



必要なコンポーネント

- SLG47004V
- 2 つの抵抗
- 2 つのコンデンサ

GreenPAK の図

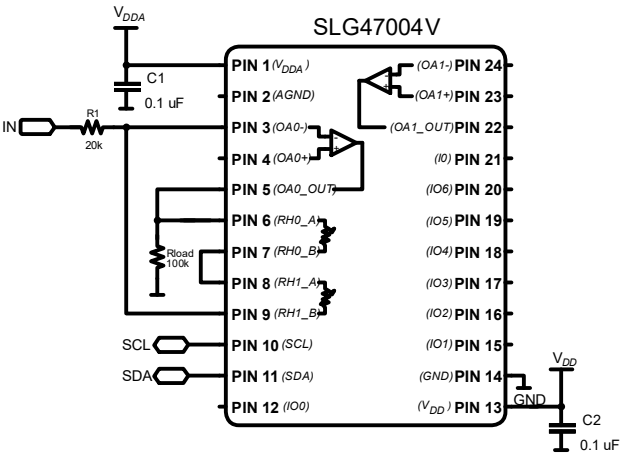


設計手順

1. OpAmp0 と OpAmp1 を有効にします。バンド幅を 8 MHz に設定し Charge Pump を有効にします。
2. Vref OA1 を有効にし、OpAmp1 の IN+と接続します。入力電圧を VDD に、出力電圧を VDD * (16/64)に設定します。
3. Digital Rheostat 1 を Rheostat モードに設定します。 Auto-Trim を無効にし、FIFO nRST 端子と POR を接続します。2 つの可変抵抗を所望の Resistance (initial data) の値に設定します。

アプリケーション：可変反転オペアンプ

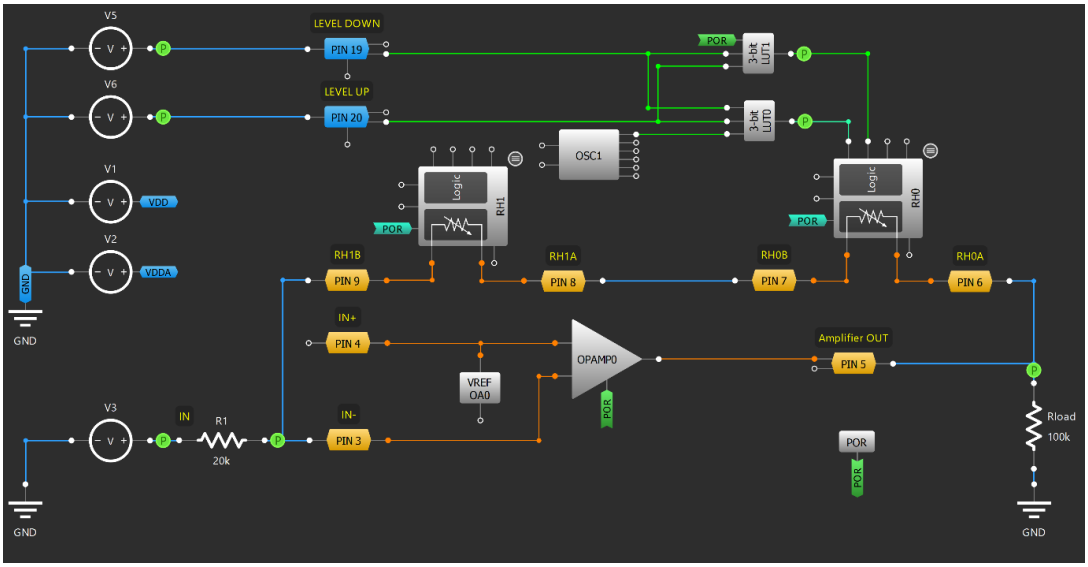
反転増幅器はオペアンプ回路の一種で、入力に対して 180° 位相のずれた出力を生成します。つまり、入力信号が正であれば出力信号は負になります。外部の制御信号により、ユーザーは反転増幅器のゲインを調整できます。



必要なコンポーネント

- SLG47004V
- １つの抵抗

GreenPAK の図



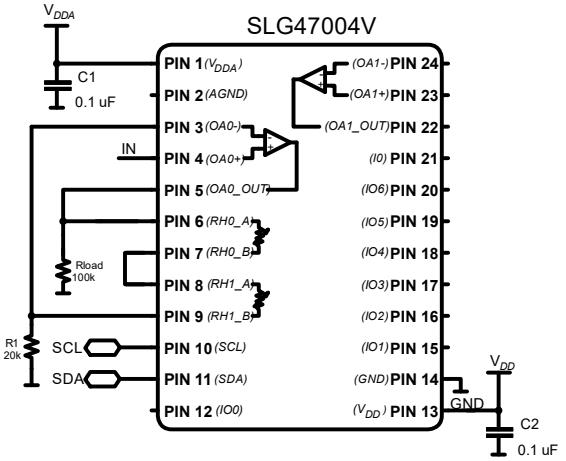
G=Vout/Vinp =-(RH0+RH1)/R1

設計手順

1. OpAmp0 を有効にします。バンド幅を 8 MHz に設定し、Charge Pump を有効にします。
2. Vref OA0 を有効にし、OpAmp1 の IN+と接続します。入力電圧を 2.048 V に、出力電圧を 512 mV に設定します。
3. Digital Rheostat 1 を Rheostat mode に設定します。Auto-Trim を無効にし、FIFO nRST 端子と POR を接続します。 ２つの可変抵抗をご希望の Resistance (initial data) の値に設定します。
4. LUT0 は LEVEL UP または LEVEL DOWN 信号が High のときに OSC1 のクロック信号を出力するように設定します。LUT1 は LEVEL UP 信号が High のときに High の信号を出力するように設定します。

アプリケーション：可変非反転オペアンプ

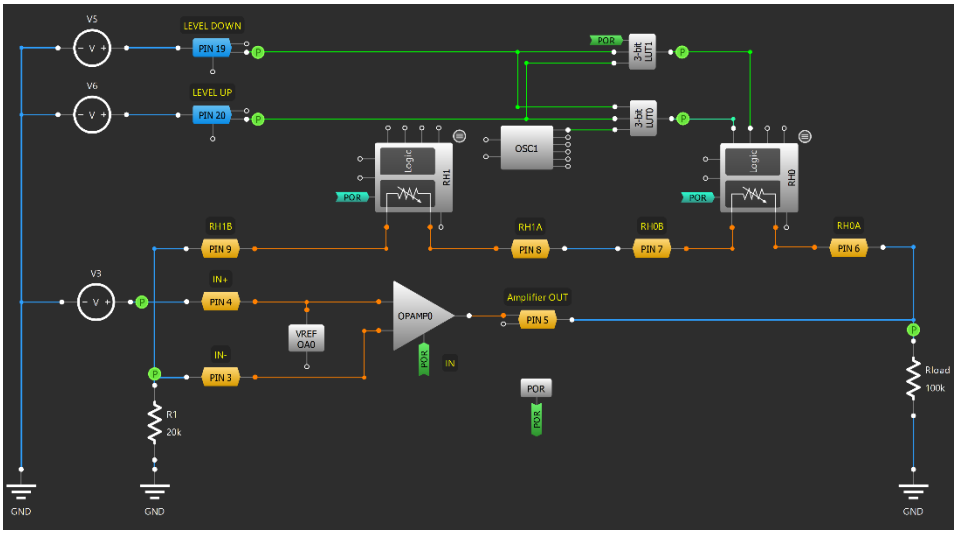
非反転増幅器はオペアンプを基にした増幅器で正の電圧利得を持ちます。入力端子にどんな信号を入力しても、出力端子で増幅される際に極性が変わることはありません。つまり、非反転増幅器のゲインは常に正になります。外部の制御信号により、ユーザーは非反転増幅器のゲインを調整できます。



必要なコンポーネント

- SLG47004V
- １つの抵抗

GreenPAK の図



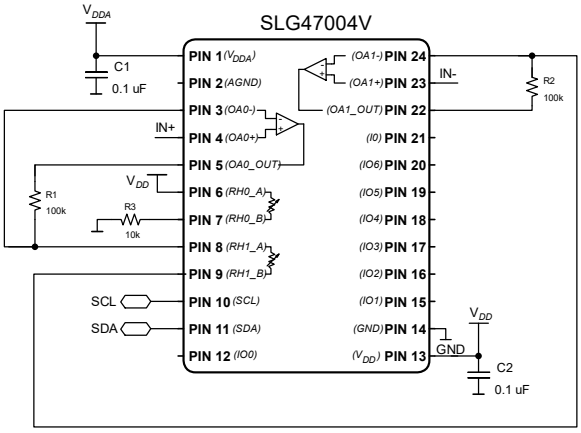
G=V_out/V_inp =(1+(RH0+RH1)/R1).

設計手順

1. OpAmp0 の PWR UP と POR を接続します。バンド幅を 8 MHz に設定し、Charge Pump を有効にします。
2. Vref OA0 を有効にし、OpAmp1 の IN+と接続します。入力電圧を 2.048 V に、出力電圧を 256 mV に設定します。
3. Digital Rheostat 1 を Rheostat mode に設定します。Auto-Trim を無効にし、FIFO nRST 端子と POR を接続します。 ２つの可変抵抗をご希望の Resistance (initial data) の値に設定します。
4. LUT0 は LEVEL UP または LEVEL DOWN 信号が High のときに OSC1 のクロック信号を出力するように設定します。LUT1 は LEVEL UP 信号が High のときに High の信号を出力するように設定します。

アプリケーション: 計装アンプ

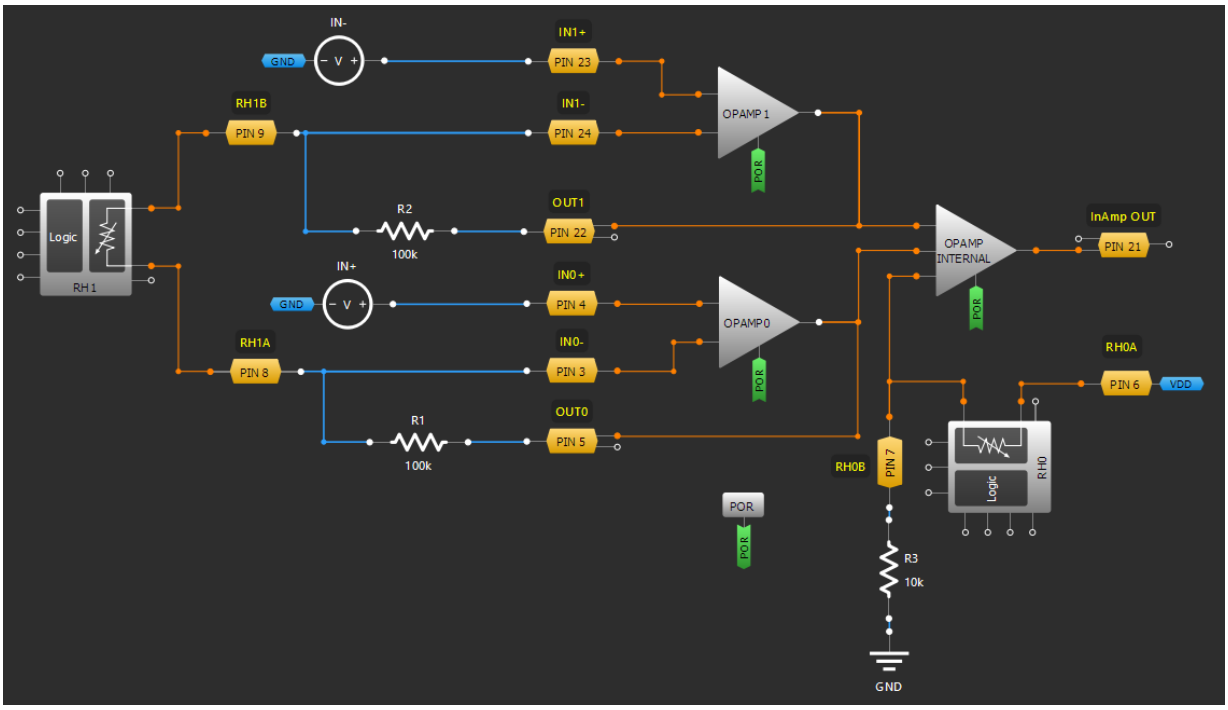
計装アンプは、入力バッファを備えた差動アンプの一種であり、入力インピーダンス整合が不要です。非常に低い DC オフセット、低ドリフト、低ノイズ、非常に高いオープンループゲイン、非常に高い同相除去比、および非常に高い入力インピーダンスを備えます。



必要なコンポーネント

- SLG47004V
- 抵抗 3 個

GreenPAK の図



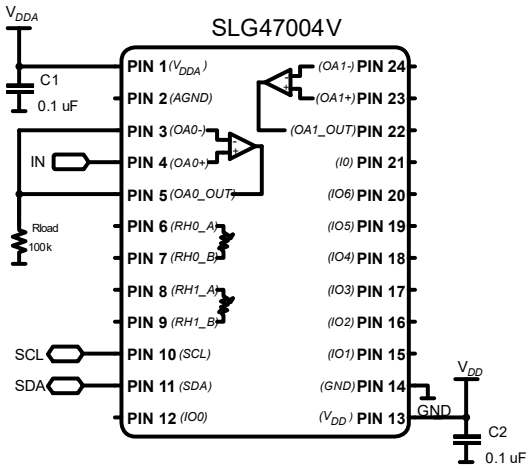
V_OUT=(1+(R_1+R_2)/R_H1)(V_(IN+)- V_(IN-)) + V_DD R_3/(R_H0+R_3).

設計手順

1. オペアンプの設定：POR を接続し OpAmp0、OpAmp、 OpAmp Internal を有効にします。帯域幅を 128kHz に設定し、チャージポンプを有効にします。 OpAmp Internal の Vref を RH0 PIN B に設定します。
2. Digital Rheostat を設定：Rheostat mode に設定します。Auto-trim は無効とし、必要な抵抗値(初期値)を両 Rheostat に設定します。 また、計装アンプを構成できるように、外部回路(抵抗・配線)を設定します。

アプリケーション: オペアンプを使ったボルテージフォロア

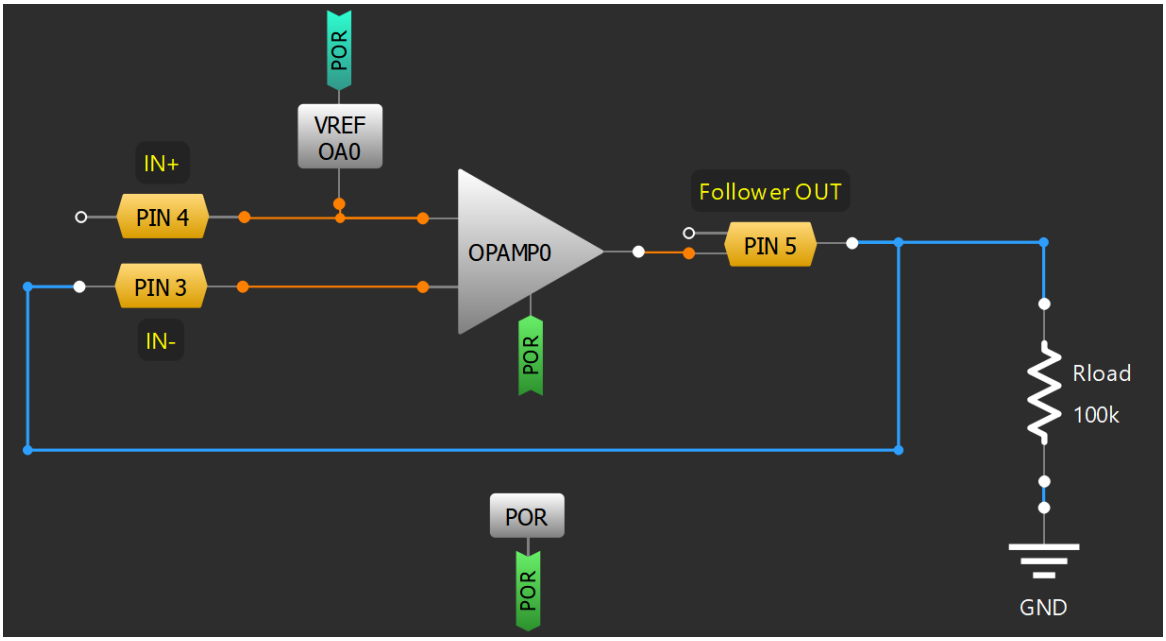
ボルテージフォロア（バッファアンプとも呼ばれます）は、出力電圧が入力電圧と等しいオペアンプ回路です。従って、ボルテージフォロアオペアンプは入力信号を増幅せず、電圧ゲインは 1 になります。ボルテージフォロア回路の入力インピーダンスは、非常に高くなります。この特性により、入力信号と出力信号の間の絶縁が必要なさまざまなタイプの回路でよく使われます。



必要な回路ブロック

- SLG47004V

GreenPAK の図

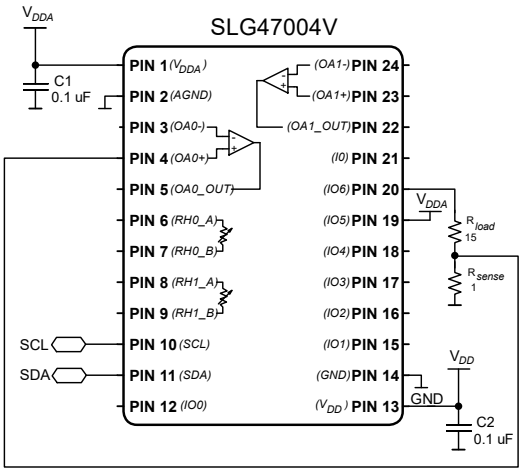


設計手順

1. OpAmp0 をイネーブルします。帯域幅を 8 MHz に設定します。チャージポンプをイネーブルして Vref を IN+に接続します。
2. Vref OA0 をイネーブルします。入力電圧を VDDA 、出力セレクタを VDDA*(8/64)に設定します。

アプリケーション：オペアンプとＰチャネル FET を使用した電流源

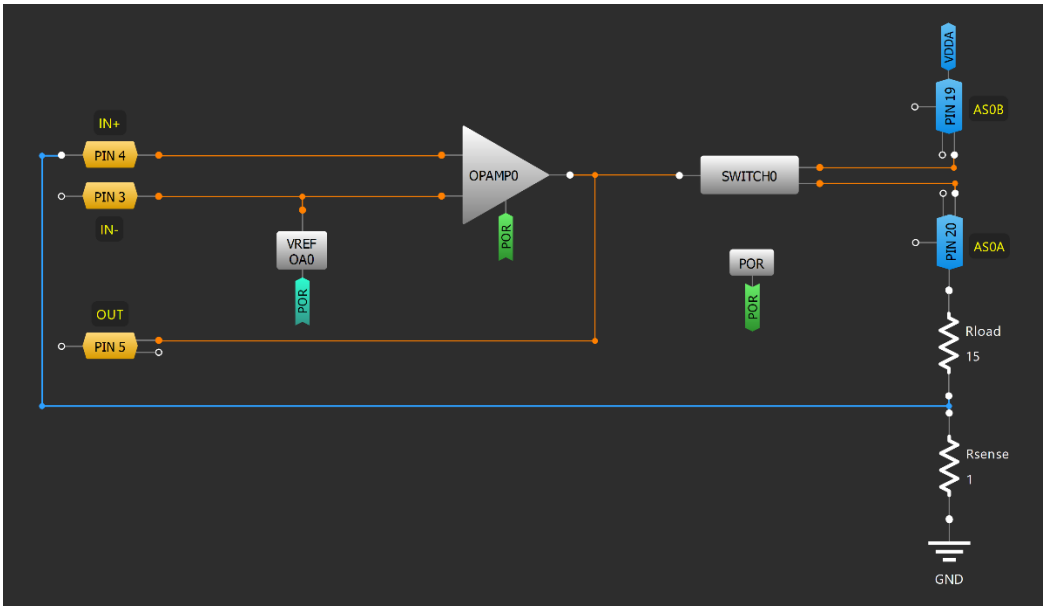
電流源として動作するオペアンプの使い方をご紹介します。この電流源は検出抵抗器にかかる電圧を一定に保つことで、負荷抵抗に関係なく負荷を流れる電流を一定にします。



必要なコンポーネント

- SLG47004V
- 抵抗１つ

GreenPAK の図



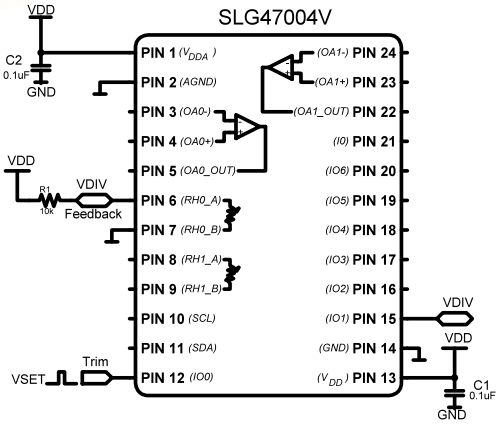
I_load=V_ref/R_sense

設計手順

1. OpAmp0 の PWR UP と POR を接続します。バンド幅を 128 kHz に、Charge Pump を Enable CP に、Vref connection を IN-に設定します。
2. Vref OA0 の PWR UP と POR を接続します。Input voltage を 2.048 V に、output selection を 96 mV に設定します。
3. Switch0 の Mode を Analog Switch に、Big PMOS control を OpAmp に設定します。

アプリケーション：オートトリム

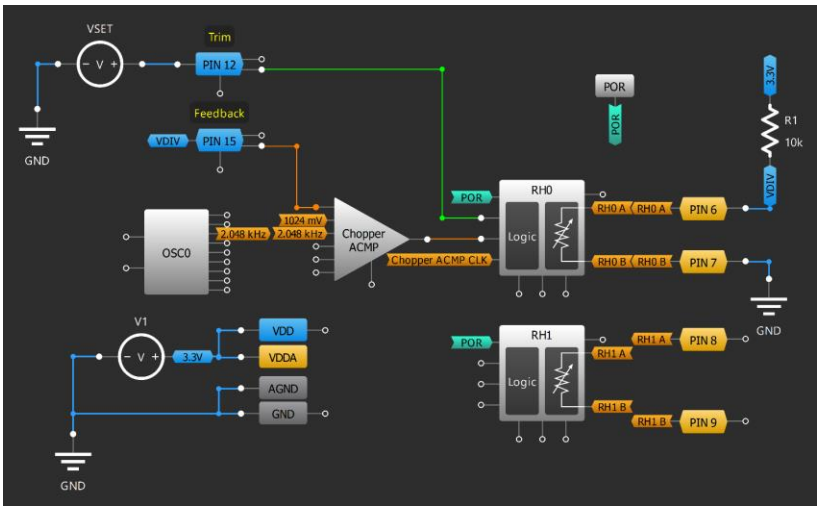
オートトリム機能を実現するための回路をご紹介します。

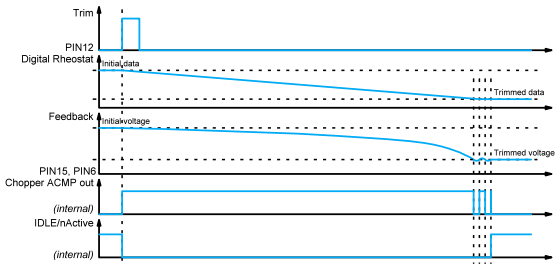


必要なコンポーネント

- SLG47004
- 抵抗

GreenPAK の図



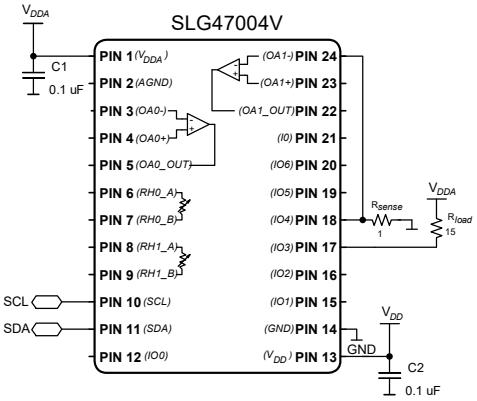


設計手順

1. VSET 信号を PIN 12 の“Trim”を通して RH0 の “SET” に接続します。
2. PIN 7 を GND に接続します。抵抗 R1 と PIN 6、 VDD を接続します。 PIN 6 から PIN 15 へのフィードバックの配線を行います。
3. Chopper ACMP の IN- CH0 source を 1024 mV に設定します。CH0 clock を OSC0 に、IN+ CH0 source を Ext. Vref PIN 15 に設定します。Chopper ACMP の出力を RH0 の nUP/Down に接続します。
4. RH0 の Auto-Trim を Enable に、Active level for UP/DOWN を Up when LOW に、Resistance (initial data) を 511 (~50.7096 kΩ)に設定します。

アプリケーション：オペアンプと N チャネル FET を使用した電流シンク

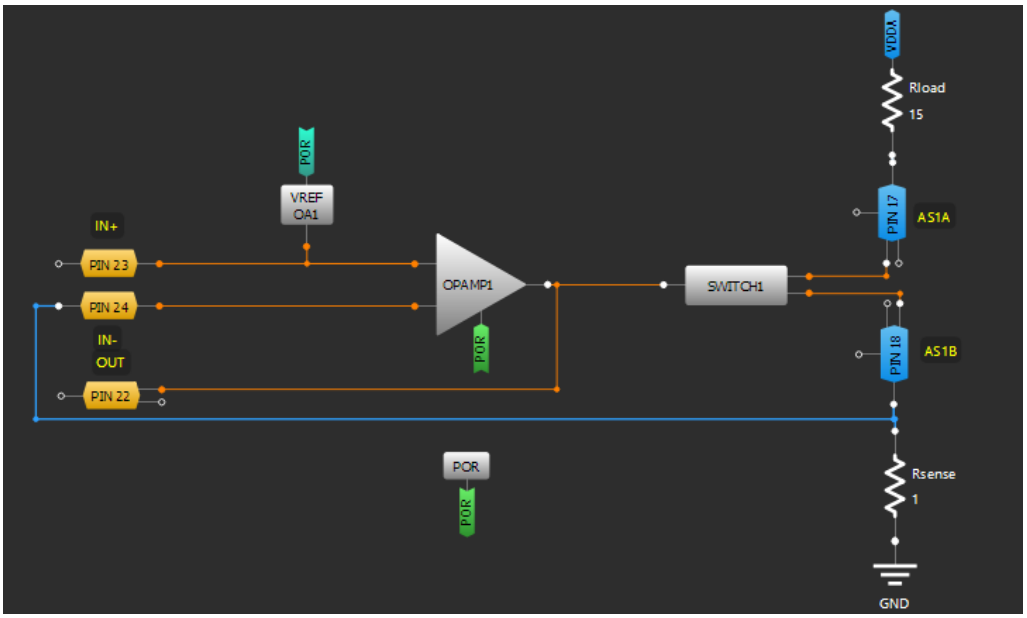
電流シンクとして動作するオペアンプの使い方をご紹介します。このシンクは一定の検出抵抗に一定 電圧をかけ続けることで、負荷の抵抗に関係なく一定の電流を出力することができます。



必要なコンポーネント

- SLG47004V
- 抵抗 1 つ

GreenPAK の図



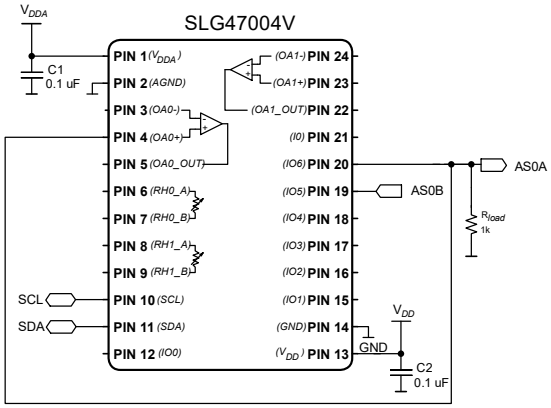
I_load=V_ref/R_sense

設計手順

1. OpAmp1 の PWR UP と POR を接続します。バンド幅を 128 kHz に、Charge Pump を Enable CP に、Vref connection を IN+に設定します。
2. Vref OA1 の PWR UP と POR を接続します。Input voltage を 2.048 V に、output selection を 96 mV に設定します。 基準電圧は負荷を通る電流を決定します。
3. Switch1 の Mode を Analog Switch に、Big NMOS control を OpAmp に設定します。

アプリケーション: OpAmp を使った電圧レギュレータ

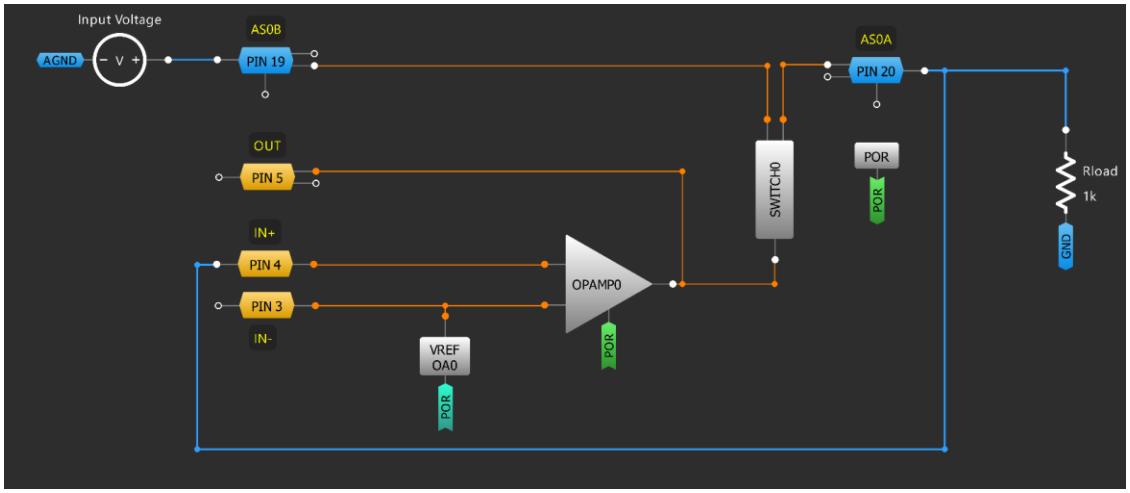
オペアンプベースの電圧レギュレータは、オペアンプを使用して出力電圧を調整および安定化する回路です。出力電圧を基準電圧と比較することにより、オペアンプは出力を調整して一定の電圧レベルを維持します。このフィードバックメカニズムにより、電圧レギュレータは入力電圧の変動を補償し、一貫して信頼性の高い出力電圧を提供できます。



必要な回路ブロック

- SLG47004V
- 他に必要なコンポーネントはありません。

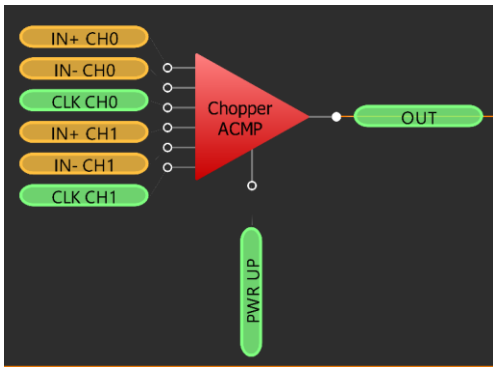
GreenPAK の図



設計手順

1. OpAmp0 をイネーブルします。帯域幅を 128 kHz に、チャージポンプをイネーブルして基準電圧を IN- に接続します。
2. Vref OA0 をイネーブルします。入力電圧を 2.048 V に、出力として 1792 mV を選択します。
3. Switch0 をアナログスイッチに設定し、Big PMOS として OpAmp により制御します。
4. 入力電圧を印加すると、出力電圧は入力電圧の変動に関わらず安定して VOUT = VREF を維持します。

技法: チョッパー ACMP をデジタル可変抵抗と一緒に使う



SLG47004 には、レイル to レイルのチョッパーアナログコンパレータ (ACMP) のマクロセルがあります。チョッパーACMP を使用して、自動トリミングモードで可変抵抗の値を変えることにより、オンボードでのトリミングが可能です。 (アプリケーション: 自動トリム機能を参照).

起動

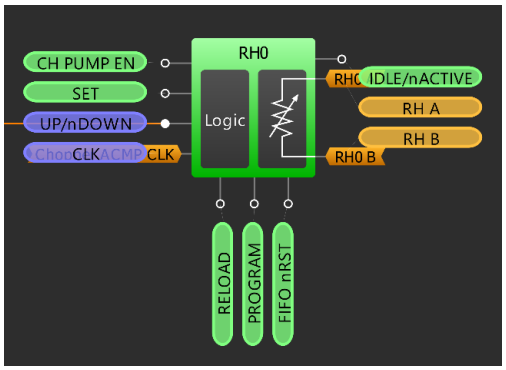
自動トリミングモードで使用すると、可変抵抗の校正中にチョッパーACMP が自動トリミング回路によって自動的に動作状態になります。

自動トリミングモードを使用する際には、次のような事前のステップが必要です：

- Digital Rheostat0/1 のプロパティで、自動トリミングをイネーブルします。

- チョッパーACMP の IN+ CH0/1 を選択、これはユーザーシステムの電圧フィードバックも可能です。２つの可変抵抗に自動トリミングを使用する場合には、両方の可変抵抗に対して IN+ CH0/1 を設定する必要があります。(SET0 がラッチされる場合と SET1 がラッチされる場合)

- IN- CH0/1 を設定します。これは、ユーザーが希望する設定値を閾値にできます。自動トリミング機能が２つの可変抵抗に使用される場合、梁上の可変抵抗に対して IN- CH0/1 を設定する必要があります。 (SET0 がラッチされる場合と SET1 がラッチされる場合)



- Chopper ACMP を使用するために、チョッパーACMP のプロパティでチャネルを選択します。

- チョッパー ACMP の出力を正転または反転に設定します。

- クロックソース（内部プリディバイダまたは接続マトリクスからの内部クロック）を選択します。自動トリミングモードではクロックソースの周波数はチョッパーコンパレータの応答時間によって制限されることに注意して下さい。従って、クロックソース周波数は <fChACMP> kHz を超えてはなりません。

- RH0/1 ブロックの SET0/1 入力を HIGH レベルに設定して、自動トリミングプロセスを開始します。トリミングプロセスは、セット入力の立上りエッジで始まります。このセット信号は、自動トリミングプロセスが終了するまでラッチされます。セット信号は、チョッパーACMP と Vref がそれ以前に有効になっていなかった場合、それらを有効にします。カウンタは、RH0/1 (チョッパー ACMP 出力) の UP/nDOWN 入力のレベルに応じてカウントアップまたはカウントダウンを開始します。ユーザーがクロック入力に「内部クロック」オプションを選択した場合、これらのクロックパルスはトリム時間中に自動的に生成されます。クロックパルスの各立上りエッジによってカウンタの値が変化し、その結果可変抵抗の値が変わります。

自動トリミングのプロセスは、３つの停止条件のうちのひとつが起きると停止します。

1) クロック入力の立上りエッジの瞬間における Up/Down 入力のその後の変化。

2) 可変抵抗の値が最大値(1023)に到達したとき

3) 可変抵抗の値が最小値(0)に到達したとき

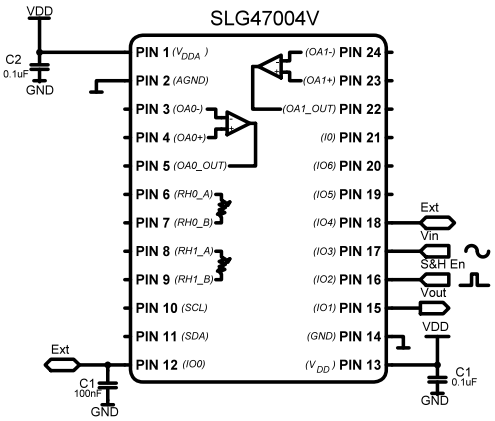
停止条件により IDLE/nACTIVE 信号が変化し、内部の自動トリミングロジックがリセットされます。

セット入力はエッジに敏感ですが、設定点に到達した後もユーザーがこの入力で HIGH ロジックレベルを維持すると、プログラマブルトリムブロックは動作を継続し、設定点の周辺で可変抵抗器を切り替え続けることに注意して下さい。

- 新たに自動トリミングプロセスを開始するには、ユーザーはセット入力に HIGH レベルを印加して下さい。

アプリケーション：サンプル・ホールド回路

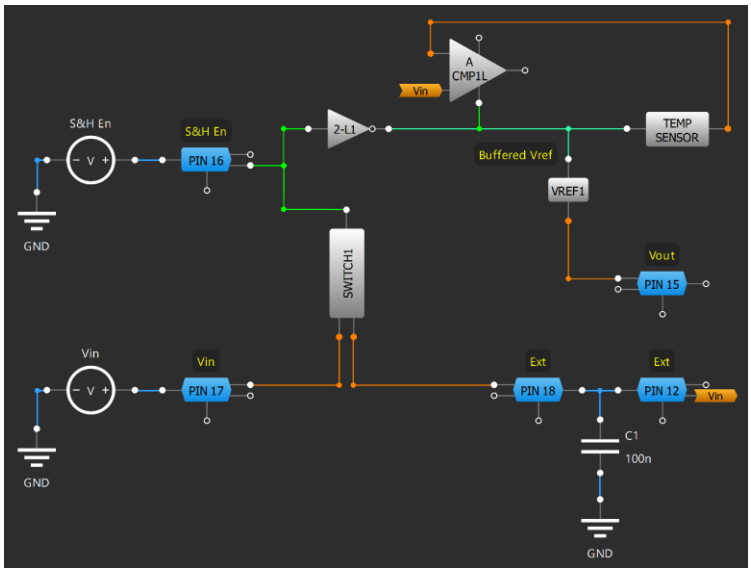
アナログスイッチ、コンデンサ、バッファから構成される、サンプル・ホールド回路をご紹介します。



必要なコンポーネント

- SLG47004
- コンデンサ

GreenPAK の図



設計手順

1. SWITCH1 の Small PMOS enable を Enable by Matrix に設定します。
2. S&H En 信号を設定し、PIN16 に接続します。 (端子の設定は Digital Input, Pull Down 1M).
3. SWITCH1 と PIN16 を接続します。
4. PIN17 (端子の設定は Analog In/Out, Floating) と信号源 Vin を接続します。PIN18 と PIN 12 (端子の設定は両方 Analog In/Out, Floating)を接続し、100 nF のコンデンサをつなぎます。
5. 2-L1 をインバータに設定し、IN0 と PIN16 を、OUT と ACMP1L/TEMP SENSOR の PWR UP を接続します。
6. ACMP1L の Vref source selection を VDDA に、IN+ source を TEMP SENSOR output に、IN- Low to High/High to Low source を Ext. Vref PIN12 に設定します。
7. TEMP SENSOR の power down source を“From matrix”に設定します。
8. PIN15 をこのサンプル・ホールド回路の Vout とします。