AN90071 - CY8CMBR3xxx

CapSense®設計ガイド

文書番号: 001-91966 Rev. *B

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
http://www.cypress.com

© Cypress Semiconductor Corporation, 2014-2019. 本書面は, Cypress Semiconductor Corporation 及び Spansion LLC を含むその子会社 (以下「Cypress」という。) に帰属する財産である。本書面 (本書面に含まれ又は言及されているあらゆるソフトウェア若しくはファームウェア (以下「本ソフトウェア」という。) を含む) は, アメリカ合衆国及び世界のその他の国における知的財産法令及び条約に基づき Cypress が所有する。Cypress はこれらの法令及び条約に基づく全ての権利を留保し, 本段落で特に記載されているものを除き, その特許権, 著作権, 商標権又はその他の知的財産権のライセンスを一切許諾しない。本ソフトウェアにライセンス契約書が伴っておらず, かつ Cypress との間で別途本ソフトウェアの使用方法を定める書面による合意がない場合, Cypress は, (1) 本ソフトウェアの著作権に基づき, (a) ソースコード形式で提供されている本ソフトウェアについて, Cypress ハードウェア製品と共に用いるためにのみ, かつ組織内部でのみ, 本ソフトウェアの修正及び複製を行うこと, 並びに (b) Cypress のハードウェア製品ユニットに用いるためにのみ, (直接又は再販売者及び販売代理店を介して間接のいずれかで) 本ソフトウェアをバイナリーコード形式で外部エンドユーザーに配布すること, 並びに (2) 本ソフトウェア (Cypress により提供され, 修正がなされていないもの) が抵触する Cypress の特許権のクレームに基づき, Cypress ハードウェア製品と共に用いるためにのみ, 本ソフトウェアの作成, 利用, 配布及び輸入を行うことについての非独占的で譲渡不能な一身専属的ライセンス (サブライセンスの権利を除く) を付与する。本ソフトウェアのその他の使用, 複製, 修正, 変換又はコンパイルを禁止する。

適用される法律により許される範囲内で, Cypress は, 本書面又はいかなる本ソフトウェア若しくはこれに伴うハードウェアに関しても, 明示又は黙示をとわず, いかなる保証 (商品性及び特定の目的への適合性の黙示の保証を含むがこれらに限られない) も行わない。いかなるコンピューティングデバイスも絶対に安全ということはない。従って, Cypress のハードウェアまたはソフトウェア製品に講じられたセキュリティ対策にもかかわらず, Cypress は, Cypress 製品への権限のないアクセスまたは使用といったセキュリティ違反から生じる一切の責任を負わない。加えて, 本書面に記載された製品には, エラッタと呼ばれる設計上の欠陥またはエラーが含まれている可能性があり, 公表された仕様とは異なる動作をする場合がある。適用される法律により許される範囲内で, Cypress は, 別途通知することなく, 本書面を変更する権利を留保する。Cypress は, 本書面に記載のある, いかなる製品若しくは回路の適用又は使用から生じる一切の責任を負わない。本書面で提供されたあらゆる情報 (あらゆるサンプルデザイン情報又はプログラムコードを含む) は, 参照目的のためのみに提供されたものである。この情報で構成するあらゆるアプリケーション及びその結果としてのあらゆる製品の機能性及び安全性を適切に設計, プログラム, かつテストすることは, 本書面のユーザーの責任において行われるものとする。Cypress 製品は, 兵器, 兵器システム, 原子力施設, 生命維持装置若しくは生命維持システム, 蘇生用の設備及び外科的移植を含むその他の医療機器若しくは医療システム, 汚染管理若しくは有害物質管理の運用のために設計され若しくは意図されたシステムの重要な構成部分としての使用, 又は装置若しくはシステムの不具合が人身傷害, 死亡若しくは物的損害を生じさせるようなその他の使用 (以下「本目的外使用」という。) のためには設計, 意図又は承認されていない。重要な構成部分とは, それの不具合が装置若しくはシステムの不具合を生じさせるか又はその安全性若しくは実効性に影響すると合理的に予想できるような装置若しくはシステムのあらゆる構成部分をいう。Cypress 製品のあらゆる本目的外使用から生じ, 若しくは本目的外使用に関連するいかなる請求, 損害又はその他の責任についても, Cypress はその全部又は一部をとわず一切の責任を負わず, かつ Cypress はそれら一切から本書により免除される。Cypress は Cypress 製品の本目的外使用から生じ又は本目的外使用に関連するあらゆる請求, 費用, 損害及びその他の責任 (人身傷害又は死亡に基づく請求を含む) から免責補償される。

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

目次

1. はじめに 5

1.1 概要 5
1.2 はじめに 5
1.3 CY8CMBR3xxx ファミリの機能概要 6
1.4 CY8CMBR3xxx ファミリの機能比較 8
1.5 CY8CMBR3xxx CapSense システム概要 9
1.6 CapSense の設計フロー 9

2. CapSense の技術 12

2.1 CapSense の原理 12
2.2 CY8CMBR3xxx の静電容量式タッチ センシング技法 14
2.3 CapSense チューニング 15
2.4 SmartSense 自動チューニング 15
2.5 センサーの種類 16
2.6 耐水性 18

3. CapSense の回路設計 21

3.1 回路図設計上の考慮事項 21
3.2 回路図の設計チェックリスト 25
3.3 回路図の例 25

4. CapSense レイアウト ガイドライン 31

4.1 Design Toolbox (デザイン ツールボックス) 31
4.2 近接センサーのレイアウト ガイドライン 40
4.3 耐水性のレイアウト ガイドライン 43
4.4 サンプル レイアウト 45

5. CY8CMBR3xxx のコンフィギュレーション 49

5.1 CY8CMBR3xxx のコンフィギュレーション 49
5.2 CY8CMBR3xxx の設定 59

6. CapSense 性能のチューニング 78

6.1 一般的な注意事項 78
6.2 ボタン、スライダー、およびガード センサーのチューニング 80
6.3 近接センサーのチューニング 85

7. 低消費電力設計上の考慮事項 93
7.1 CY8CMBR3xxx の動作モード 93
7.2 消費電力に影響する要因 95
7.3 低消費電力向けのシステム設計推奨項目 96
8. リソース 97
8.1 ウェブサイト 97
8.2 データシート 97
8.3 レジスタ技術リファレンス マニュアル 97
8.4 Design Toolbox (デザイン ツールボックス) 97
8.5 EZ-Click™ 2.0 97
8.6 開発キット 97
8.7 デザイン サポート 98
9. 付録 99
9.1 高レベル API 99
9.2 低レベル API 105
用語集 108
改訂履歴 113

1. はじめに

1.1 概要

このガイド (CY8CMBR3xxx CapSense®設計ガイド) は CY8CMBR3xxx CapSense コントローラー ファミリを使う静電容量のタッチセンシング アプリケーションの設計方法について説明します。CY8CMBR3xxx CapSense コントローラー ファミリは前例のない信号対ノイズ比 (SNR)、クラス最高水準の耐水性、および多種多様なセンサー (ボタン、スライダーや近接センサーなど) を提供します。このガイドでは CY8CMBR3xxx CapSense の設計の流れ、設計ツール、設計上の考慮事項および性能調整について説明します。

1.2 はじめに

静電容量タッチ センサーは、人体の静電容量を使ってセンサー上またはセンサー近くの指の存在を検出するユーザー インターフェースのデバイスです。静電容量センサーは審美的に優れており、使い易く、長寿命を持っています。サイプレスのCapSenseソリューションでは、ユーザーの製品に優雅で、信頼性が高く、簡単に使用できる静電容量タッチ センシング機能をもたらしています。サイプレスのCapSenseソリューションは、40億個以上の機械式ボタンを置き換えてきました。

CY8CMBR3xxx CapSense Express™コントローラーは、高度な機能を持った実装が簡単な静電容量タッチセンシングを使用するユーザー インタフェース ソリューションを提供します。最大16の静電容量センシング入力をサポートするこのレジスタ設定可能なファミリは、設計サイクルから時間のかかるファームウェア開発および調整手順を必要としません。これらのコントローラーは、最小限の開発サイクル時間で静電容量ボタン、スライダー及び近接センシング ソリューションを実装するのに理想的なものです。CY8CMBR3xxxコントローラーは、霧、湿気、液滴または流水に起因する誤検知をなくすることにより、堅牢な耐水性のあるCapSenseアプリケーションを可能とします。

CY8CMBR3xxx CapSense の設計ガイドは、 CapSense コントローラーの CY8CMBR3xxx ファミリを使用して、静電容量式タッチセンシングの機能を実装する方法について説明します。このガイドでは、サイプレスの CapSense の技術に精通していることを前提としています。CapSense の技術を初めて使用する場合、その基礎を理解するために CapSense 入門のガイドを参照してください。

この設計ガイドでは、以下の項目について説明します。

- CY8CMBR3xxx CapSense コントローラー ファミリの機能
- CY8CMBR3xxx ファミリで利用される CapSense 技術
- 回路図およびレイアウト設計上の考慮事項
- CY8CMBR3xxx ファミリの設定方法
- 性能チューニング
- 低消費電力設計上の注意事項

1.3 CY8CMBR3xxx ファミリの機能概要

CY8CMBR3xxx ファミリは、高度な機能を持った実装が簡単な静電容量タッチセンシングを使用するユーザー インタフェース ソリューションを提供する高性能、低消費電力の CapSense Express™コントローラーです。

CY8CMBR3xxx ファミリの機能は以下の通りです。

- レジスタ設定可能な CapSense Express コントローラー
  - I²C インターフェースを通じて設定可能
  - ファームウェア開発またはデバイス プログラミングは不要
  - 最大 16 の静電容量センシング入力をサポート
  - 最大 8 つの汎用出力 (GPO) をサポート
  - GPO は CapSense のセンサーにリンクされるか、ホスト プロセッサによって制御
  - GPO は直接 LED を駆動可能
- SmartSense™ 自動チューニング
  - システム、製造及び環境の変化に応じて連続的に補正を行う CapSense アルゴリズム
  - 最適な性能を実現するために自動的に静電容量ボタンのパラメーターを設定
  - システムの手動調整が不要
  - 広範囲の寄生容量 (C_P) に対応 (5～45pF)
  - 必要に応じてユーザーが手動で閾値を設定することが可能
- 高度な機能
  - ウェイク オン アプローチ
    - 近接イベントで低電力モードからアクティブ モードへとシステムをウェークアップ
  - 耐水性
    - 静電容量ボタンを使用可能にして液滴、水流、または霧が存在する環境でも正常に動作
  - GPO ピンでの PWM 出力
    - センサーの ON／OFF 状態の間にユーザー設定可能な LED の輝度
  - 隣接センサー抑制 (FSS)
    - 複数の隣接したボタンからの接触を識別
  - SNR を改善さすファームウェア フィルタ
  - 音響タッチ フィードバック用のブザー信号出力
  - 外部抵抗ブリッジを使用したアナログ電圧出力
  - センサーの状態変化をホストに知らせる為の割り込み出力
  - I²C インターフェースを介してシステム診断データ
    - 生産ラインのテストやシステムのデバッグを簡素化
- ノイズ耐性
  - 電磁妨害 (EMI) を最小限にするための疑似ランダムシーケンス (PRS) のクロック ソース
  - 外部の放射ノイズおよび伝導性ノイズに対して優れたノイズ耐性をサポートする電磁環境適合性 (EMC)

- システム診断機能は以下のものを検出
  - VDD とグランドに短絡されるセンサー
  - センサー間の短絡
  - センサーがシールドと短絡
  - 調整用コンデンサ (C_{MOD}) の不適切な値
  - 寄生容量 (CP) の値が範囲外
- I^2C インターフェース
  - 400kHz までの速度をサポート
  - ハードウェアのアドレス一致のウェークアップ
  - クロック ストレッチなし
- 広い動作電圧範囲
  - 1.71～5.5V の動作電圧
- 低消費電力
  - デバイスの消費電力を低減するために異なる動作モードをサポート
  - 120ms のスキャン周期でセンサー当たりの平均電流消費量は 22µA
  - I^2C が ON 状態に備えるディープスリープ電流: 2.5µA
- 工業用途向け温度範囲: -40°C ～ +85°C
- パッケージのオプション
  - 8 ピン SOIC (150mil)
  - 16 ピン SOIC (150 mil)
  - 16 ピン QFN (3×3×0.6mm)
  - 24 ピン QFN (4×4×0.6mm)

1.4 CY8CMBR3xxx ファミリの機能比較

表 1-1 は CY8CMBR3xxx ファミリの異なるコントローラーにサポートされている機能を比較します。これらのコントローラーはサポートされるセンサーや機能の数に基づいて区別されます。設計要求を満たす適切なデバイスを選択するのにこの表を使用します。

表 1-1.CY8CMBR3xxx コントローラーの比較

#	機能	MBR3116	MBR3106S	MBR3110	MBR3108	MBR3102	MBR3002
1	ボタンの最大数	16	11	10	8	2	2
2	スライダー セグメントの最大数	×	10	×	×	×	×
3	近接センサーの最大数	2	2	2	2	2	×
4	シールド電極	✓	✓	✓	✓	✓	×
5	ガード センサー	✓	×	✓	✓	×	×
5	ウェイク オン アプローチ	✓	✓	✓	✓	✓	×
6	耐水性	✓	×	✓	✓	✓	×
7	GPOs／LED 駆動出力の最大数	8	0	5	4	1	2
8	LED 輝度調整	✓	×	✓	✓	✓	×
9	I^2C インターフェース	✓	✓	✓	✓	✓	×
10	ブザー駆動出力	✓	✓	✓	✓	×	×
11	ホスト割り込み出力	✓	✓	✓	✓	×	×
12	FSS	✓	✓	✓	✓	✓	×
13	メディアン及び IIR フィルタ	✓	✓	✓	✓	✓	✓
14	高度な LOW パス フィルタ	✓	×	✓	✓	✓	×
15	電磁環境適合性 (EMC)	✓	✓	✓	✓	✓	×
16	感度制御	✓	✓	✓	✓	✓	×
17	自動指閾値	✓	✓	✓	✓	✓	✓
18	柔軟な指閾値	✓	✓	✓	✓	✓	×
19	LED ON 時間	✓	×	✓	✓	✓	×
20	トグル モード	✓	×	✓	✓	✓	×
21	システム診断機能	✓	✓	✓	✓	✓	✓
22	センサー自動リセット	✓	✓	✓	✓	✓	✓

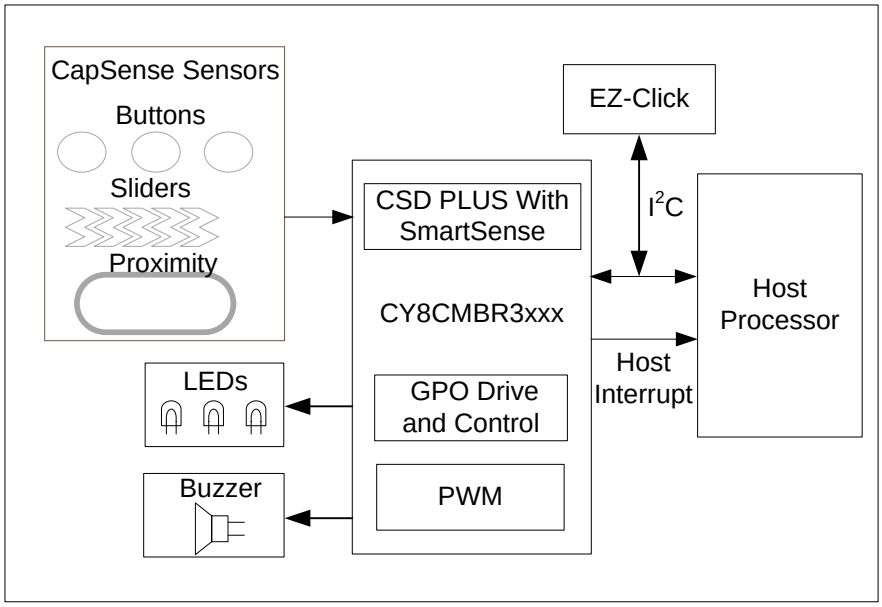
注:

- CY8CMBR3102コントローラーはガードセンサーをサポートしないため、液体流ではなく液滴のみに対して耐水性を提供します。
- CY8CMBR3002用のセンサー自動リセットは20秒に固定され、変更することができません。

1.5 CY8CMBR3xxx CapSense システム概要

図 1-1 は、CY8CMBR3xxx CapSense コントローラーとの CapSense アプリケーションの一般的なシステム レベルのブロック図を示します。The CY8CMBR3xxx コントローラーは、ユーザー インターフェース パネル上の指タッチを検出し、タッチ状態をホスト プロセッサに報告するスレーブ デバイスとして作動します。CY8CMBR3xxx コントローラーは、ホストが I²C インタフェースを通じてセンサーの状態を読み込むために、センサーの状態が変化するたびにホスト プロセッサに割り込みパルスを送信します。CY8CMBR3xxx コントローラーは駆動 GPO および専用ブザー出力でタッチ フィードバックを提供します。EZ-Click™ ツールは I²C インタフェースを使用して CY8CMBR3xxx コントローラーをコンフィギュレーションし、デバッグするために使用されます。

図 1-1. CY8CMBR3xxx コントローラーを備えた CapSense システムのブロック図



1.6 CapSense の設計フロー

図 1-2 は CY8CMBR3xxx コントローラーを使用する静電容量センシングの製品設計サイクルの代表的なフローを示したものです。表 1-2 には CY8CMBR3xxx コントローラーを使用する CapSense アプリケーションを設計するためのサポート ドキュメンテーションおよびエコシステムが一覧表示されます。

図 1-2. 標準的な CapSense の製品設計フロー

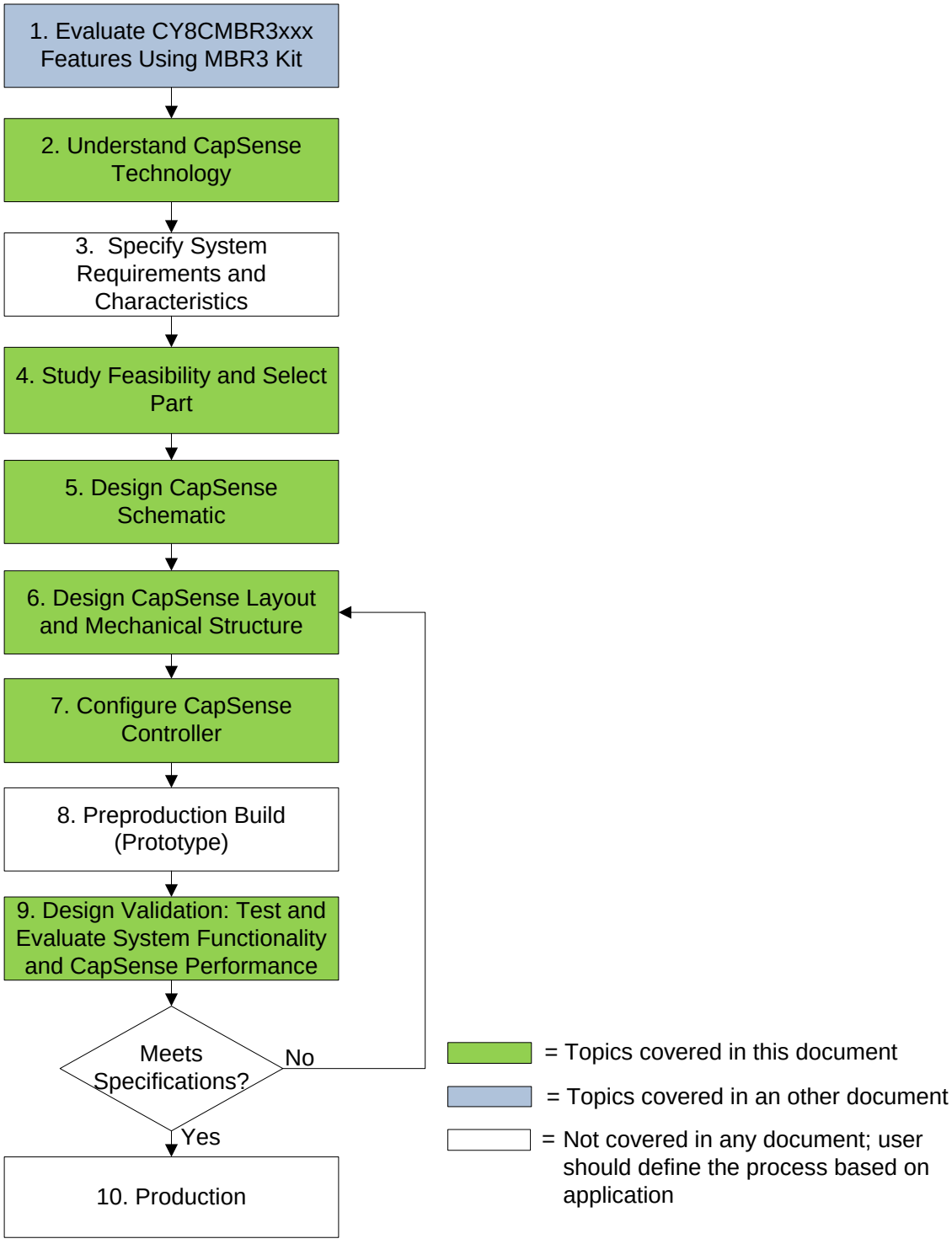


表 1-2. サポートするドキュメンテーション及びエコシステム

フローチャート内のステップ	サイプレスのサポートドキュメンテーション	
	ドキュメント名	章
1. MBR3 キットを使用して CY8CMBR3xxx の機能を評価	CY3280-MBR3 評価キットのユーザー ガイド	該当なし
2. CapSense 技術を理解する	CapSense 入門 CY8CMBR3xxx CapSense の設計ガイド (本書)	第 2 章
3. システム要求および特性を指定	いかなるドキュメントにも記載されていない; ユーザーはアプリケーションに基づいてプロセスを定義する必要がある	該当なし
4. 実行可能性の調査: パーツ選択	CY8CMBR3xxx CapSense の設計ガイド (本書) CY8CMBR3xxx データシート	第 1 章
5. CapSense 回路設計	CY8CMBR3xxx CapSense の設計ガイド (本書)	第 3 章
6. CapSense レイアウト及び機械構造を設計	CY8CMBR3xxx CapSense の設計ガイド (本書)	第 4 章
7. CapSense コントローラーを設定	CY8CMBR3xxx CapSense の設計ガイド (本書)	第 5 章
8. 試作ビルド	いかなるドキュメントにも記載されていない; ユーザーはアプリケーションに基づいてプロセスを定義する必要があります。	該当なし
9. 設計検証	CY8CMBR3xxx CapSense の設計ガイド (本書)	第 6 章
10. 製造	いかなるドキュメントにも記載されていない; ユーザーはアプリケーションに基づいてプロセスを定義する必要があります。	該当なし

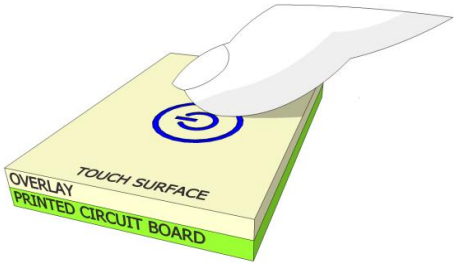
2. CapSense の技術

静電容量式タッチ センシング技術は、タッチ面上または近傍の指の存在を検出するために、面 (センサー) とその周辺環境との間で起きる静電容量の変化を測定します。

2.1 CapSense の原理

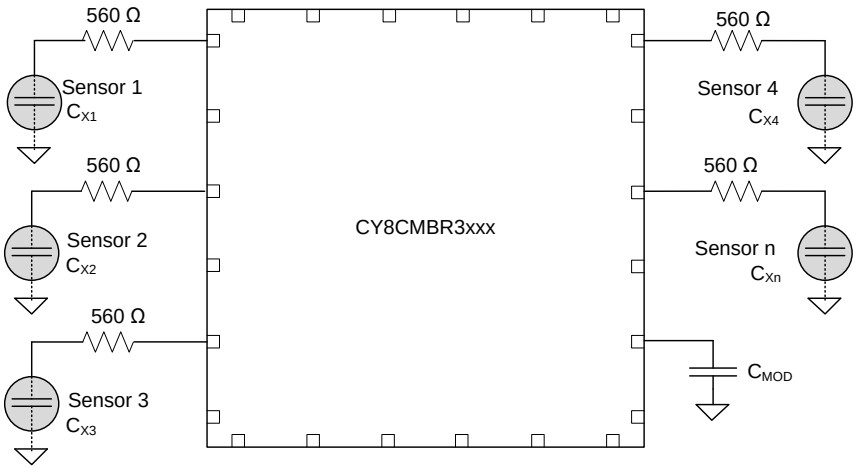
標準的な CapSense センサーは、プリント回路基板(PCB)の表面にエッチングされた銅パッドからコンフィギュレーションされています。図 2-1 に示したように、非導電性のオーバーレイ素材はセンサーのタッチ面として機能します。

図 2-1.静電容量タッチ センサー



PCB 基板の配線とビアは、センサー パッドを CY8CMBR3xxx コントローラーの CapSense 入力ピンに接続します。図 2-2 に示したように、各センサー ピンの合計静電容量は、C_{X1}、C_{X2}～C_{Xn} の値を持つ等価集中コンデンサとしてモデル化されます。CY8CMBR3xxx コントローラー内の CapSense 回路は、これらの静電容量の値を等価のデジタル カウントに変換します。そして、これらデジタル カウントは接触を検出するためにコントローラーで処理されます。

図 2-2.CY8CMBR3xxx デバイスへの CapSense の実装



注: CY8CMBR3xxx コントローラーは、CMOD ピンとグランドの間に接続される外部コンデンサ C_{MOD} を必要とします。

指を置かない場合のセンサーの静電容量は *寄生容量* C_P と呼ばれます。寄生容量は、センサー (センサー パッド、配線、ビアを含む) とシステム内の他の導電体 (グランド面、配線、製品のシャーシまたは封入物その他) の間で生じる電場により生じます。また、CY8CMBR3xxx コントローラーのセンサー ピン及び内部容量も寄生容量に寄与します。しかし、これらの内部容量はセンサー容量に比べ非常に小さいです。図 2-3 は、 CY8CMBR3xxx センサー ピンが配線とビアでセンサー パッドに接続する方法について説明しています。一般的には、グランドハッチは、他のセンサーと配線から分離させるためにセンサー パッドの周りを囲んでいます。この図はセンサー パッド周辺で幾つかの磁力線を示していますが、実際の電界分布は非常に複雑です。

図 2-3.寄生容量

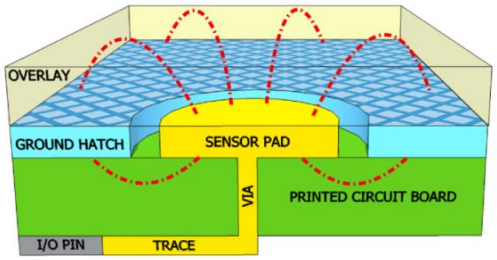
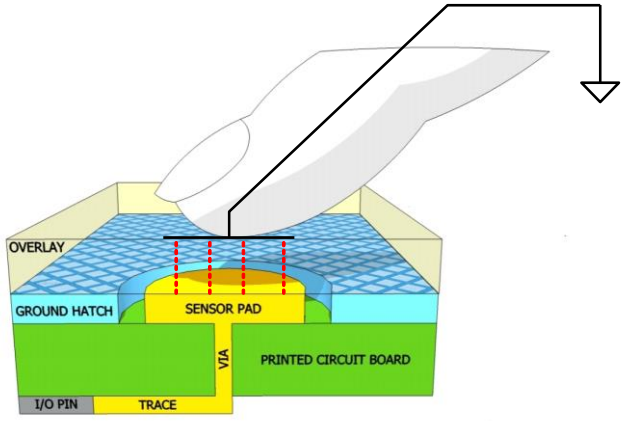


図 2-4 に示すように、指がオーバーレイに触れると、人体の導電性と大きな体積によりセンサー パッドに対し平行な接地された導体面と成ります。

図 2-4.センサーが指で活性化された時の一般的な CapSense を実装した PCB の断面図



ここでの構造は平行板のキャパシタとなります。センサー パッドと指の間の静電容量は次の通りです。

$$C_F = \frac{\varepsilon_0 \varepsilon_r A}{d}$$ 式 1

ε_0=真空の誘電率

ε_r = オーバレイの比誘電率

A = 指とセンサー パッドが重なっている面積

D = オーバーレイの厚さ

C_F は指の静電容量として知られています。寄生容量 C_P と指の静電容量 C_F は互いに並列です。これは、両者がセンサー ピンとグランド間で容量になるためです。そのため、指がセンサーに触れている時のセンサーの合計静電容量 C_S は、C_P と C_F の合計になります。

$$C_X = C_p + C_F$$ 式 2

接触していない場合は、C_X は C_P です。

CY8CMBR3xxx コントローラーは、*raw カウント*と呼ばれる等価デジタル カウントを容量 C_X に変換します。指の接触により、センサー ピンの合計静電容量が増加するので、raw カウント内の増加は指タッチを示します。寄生容量 C_P が増加すると、C_F の C_P に対する比率は減り、指タッチに対応する静電容量の単位当たりの変化も減ります。そのため、C_P が増加すると、タッチ検出がより困難になります。

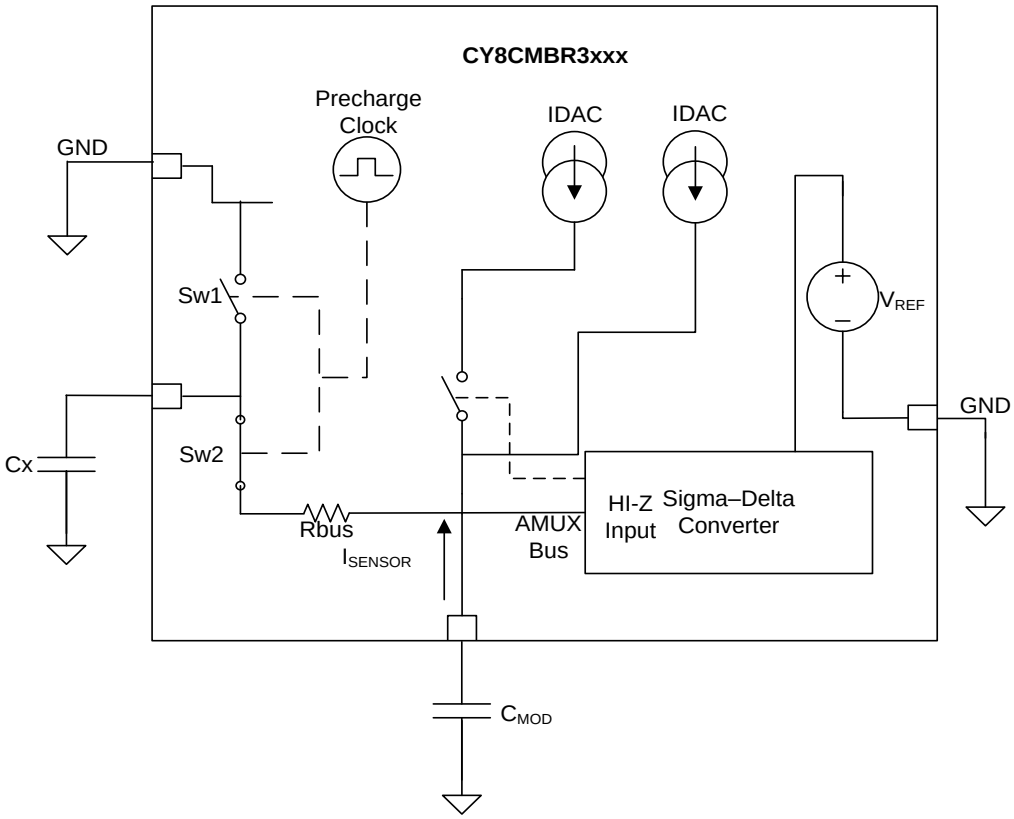
一般的に、 C_P は C_F より桁違いに大きい値です。C_P は通常 10pF～20pF の範囲ですが、極端な場合は 45pF まで大きくなることもあります。C_F は通常 100～400fF の範囲です。

2.2 CY8CMBR3xxx の静電容量式タッチ センシング技法

CY8CMBR3xxx コントローラーは、静電容量式タッチ センシング技法—CapSense シグマ デルタ PLUS (CSD PLUS) を使用して、静電容量の変化をデジタル カウントに変換します。CSD PLUS タッチ センシング技法では、極端にノイズの多い環境下でもタッチ精度を保証する信号対ノイズ比 (SNR) を提供します。

CY8CMBR3xxx デバイスの CSD PLUS CapSense センシング技法は、 図 2-5 に示すように、C_X をスイッチト キャパシタ回路に組み込まれます。C_X は、重複期間のないスイッチ Sw1 および Sw2 によって GND および AMUX バスに交互に接続されます。Sw1 および Sw2 はプリチャージ クロックによって駆動され、AMUX バスから電流 (I_{SENSOR}) を流出します。I_{SENSOR} は C_X の大きさに直接比例します。シグマ デルタ コンバータは、AMUX バス電圧をサンプリングして、定電流源 (IDAC) を制御する変調ビット ストリームを生成します。IDAC は、AMUX バスの平均電圧が V_{REF} に維持されるように AMUX を充電します。センサーは、変調コンデンサ (C_{MOD}) から電流 I_{SENSOR} を出力します。CMOD は、Rbus と一緒に、シグマ デルタ コンバータの入力部でプリチャージ スイッチングの過渡電流を減衰させる LOW パス フィルタを形成します。

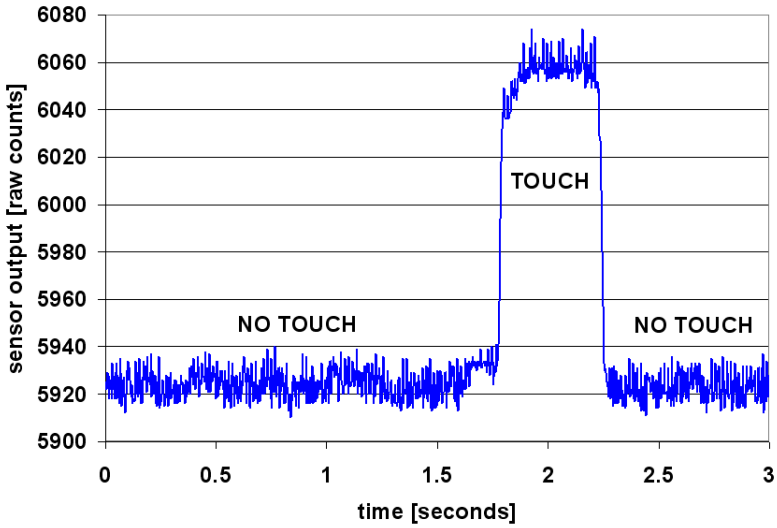
図 2-5.CSD PLUS ブロック図



AMUX バス電圧を V_{REF} 電圧に維持するために、シグマ デルタ コンバータは、ビット ストリームのデューティ サイクルを制御することで IDAC 出力電流を I_{SENSOR} に一致させます。シグマ デルタ コンバータはセンサー スキャンの期間に渡ってビット ストリームを格納し、蓄積された結果が C_X に正比例した、raw カウントであるデジタル出力となります。この raw カウントは、高レベルのアル

ゴリズムによって解釈され、センサーの状態を判断します。図 2-6 は、センサーを指で触れて離すまでの間の連続したスキャン数により CSD PLUS の raw カウントをプロットしたものです。CapSense の原理で説明したように、指の接触によって、C_X が C_F だけ増加し、その結果、raw カウントが比例して増加します。定常状態で raw カウント レベルのシフトをあらかじめ定められた閾値と比較することで、センサーが ON (接触) 状態であるか OFF (接触なし) 状態であるかを高レベルのアルゴリズムにより決定できます。raw カウント、指の閾値および信号対ノイズ比 (SNR) の詳細については、CapSense 入門のガイドを参照してください。

図 2-6.指を触れたときの CSD の raw カウント



2.3 CapSense チューニング

CapSense システムの最適な性能は、基板のレイアウト、センサーの寸法、オーバーレイの素材、およびアプリケーションの要件などによって決まります。これらの要素は CapSense レイアウト ガイドラインで説明します。これらの要素に加え、CSD PLUS センシング技法は堅牢な動作のために幾つのハードウェア及びソフトウェア パラメーターがセットされることを要求します。これらのパラメーターの最適値を設定するプロセスは手動チューニングと呼ばれています。手動チューニングは面倒で時間のかかるプロセスです。サイプレスの SmartSense 自動チューニング アルゴリズムは、すべてのハードウェア及びソフトウェアのパラメーターを自動的に設定することで手動チューニング プロセスを不要にします。

2.4 SmartSense 自動チューニング

タッチ センシングのユーザー インタフェースのチューニングは、適切なシステムの動作と快適なユーザーの使用体験のために重要です。標準的な設計フローには、初期設計段階、システム統合中、および生産立上げ前の最終製造の微細調整でのセンサー インターフェースのチューニングが含まれています。チューニングは繰り返しプロセスなので時間がかかることがあります。

SmartSense 自動チューニングは、最適な性能を実現するために静電容量ボタンのパラメーターを自動的に設定する CapSense アルゴリズムです。これは連続的にシステム、製造および環境変化のために補正し、ユーザーインターフェースの開発サイクルを簡略化します。それは使い易く、試作や製造段階で手動チューニングを無くすことで、設計のサイクル時間を短縮します。SmartSense 自動チューニングは、電源が入ったままで、各 CapSense ボタンを自動的にチューニングし、実行時の最適なボタン操作を維持します。SmartSense 自動チューニングは、PCBs 及びオーバレイの製造上のばらつきに適応します。LCD インバータ、AC ライン、スイッチ モード電源等、様々なソースからのノイズを自動的に消します。SmartSense アルゴリズムの動作方法を理解するために CapSense 入門のガイドを参照してください。

2.5 センサーの種類

CY8CMBR3xxx は、図 2-7 に示すように、ボタン、スライダー及び近接センサーの 3 種類の静電容量センサーをサポートします。

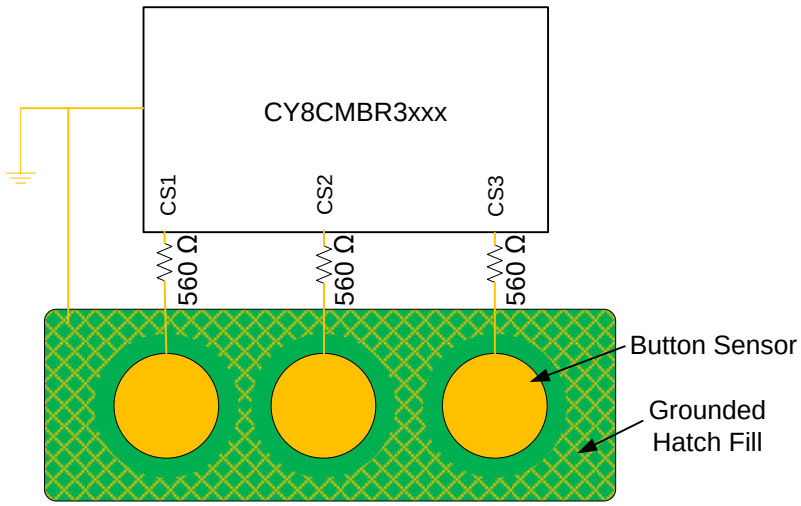
図 2-7.CY8CMBR3xxx にサポートされる静電容量センサーの種類



2.5.1 ボタン (0 次元)

CapSense のボタンは、多種多様なアプリケーション (家電、医療機器、照明管理および他の多くの製品) の中で使われている機械式ボタンに取って代わります。これは、単一センサーからなる CapSense センサーの最も簡単な種類です。CapSense のボタンは、2 つの出力状態の内の 1 つを出力します: アクティブ (指がボタンに接触) か非アクティブ (指がボタンに不接触) です。これら 2 状態はそれぞれに ON と OFF 状態とも呼ばれます。簡単な CapSense ボタンは円形の銅パッドでコンフィギュレーションされます。図 2-8 に示したように、CY8CMBR3xxx コントローラー上の CapSense ボタンの機能のためにマークされるピンに接続します。CY8CMBR3xxx では、CSx ピンはボタン センサーがそれに接続できることを示します。ボタンは他のボタンや配線から分離するためにグランドされた銅のハッチ部で囲まれています。円形のギャップは、ボタン パッドとグランド ハッチ部を分離しています。レイアウトの推奨については、全般的なレイアウト ガイドライン節を参照してください。

図 2-8.簡単な CapSense ボタン



2.5.2 スライダー (1 次元)

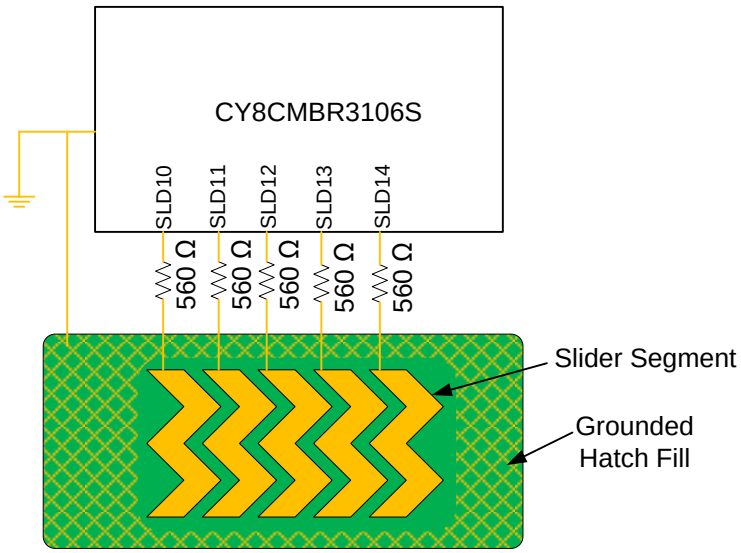
スライダーは、必要な出力が漸次インクリメントまたはデクリメントされる形態である場合に使用されます。例としては、照明管理 (調光装置) 、音量管理、グラフィック イコライザ、速度制御などが挙げられます。スライダーは、互いに隣接する静電容量センサーのアレイでコンフィギュレーションされます。これらのアレイはセグメントと呼ばれます。1 個のセグメントに触れると、隣接するセグメントを部分的に活性化します。ファームウェアは、**図心位置 (centroid position)** として知られる指タッチの幾何学的中心の位置を計算するために、触れたセグメントと近傍のセグメントからの raw カウントを処理します。

計算された図心位置の実際の分解能は、スライダーのセグメント数よりさらに高いです。例えば、5 個のセグメントを持つスライダーでは、少なくとも 100 ヶ所の物理的な指の位置を判別できます。この高精度の分解能は、スライダーを横切る指の滑りとして

図心位置のスムーズな移動を感知します。図 2-9 に示した通り、リニアスライダーにおいては、セグメントがインラインに配置されます。

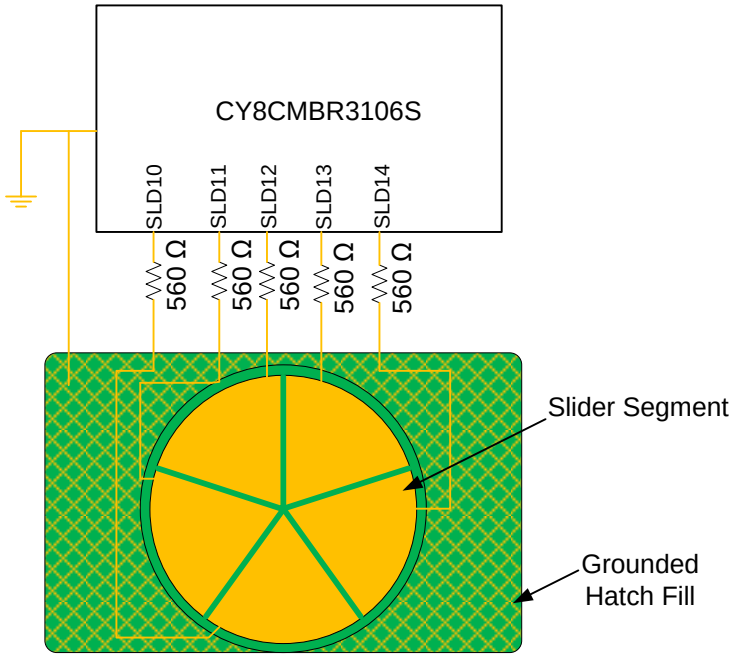
各スライダー セグメントは、CY8CMBR3106S コントローラー上のスライダーの機能用にマークされるピンに接続します。CY8CMBR3106S では、SLDxx ピンはスライダー セグメントがそれに接続できることを示します。スライダーのセグメント用には、ジグザグのパターン (二重の山形) が推奨されます。このレイアウトにより、セグメントに触れると、必ず隣接セグメントにも部分的に触れます。これは、図心位置を計算する助けになります。レイアウトの推奨については、全般的なレイアウト ガイドライン の節を参照してください。

図 2-9.リニアスライダー



ラジアル スライダーが連続していること以外は、ラジアル スライダーはリニア スライダーに似ています。図 2-10 は、一般的なラジアル スライダーを示します。

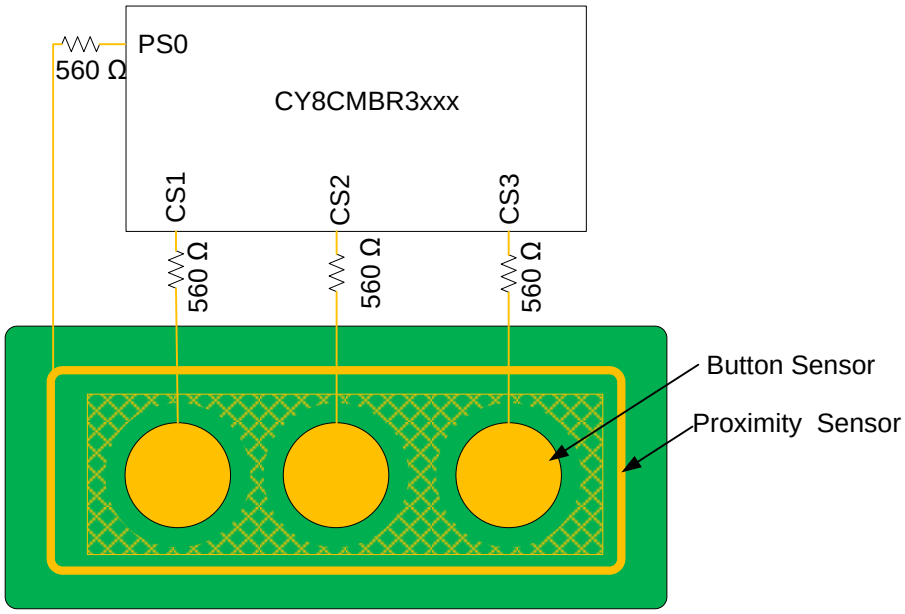
図 2-10.ラジアルスライダー



2.5.3 近接 (3 次元)

近接センサーは、センサー周囲の 3 次元空間における手の存在を検出します。しかし、近接センサーの実際の出力は、CapSense のボタン同様に ON／OFF 状態です。近接センシングは、センサーのサイズ及び構造により、数 cm～10cm 離れた手を検出できます。各近接センサーは、CY8CMBR3xxx コントローラー上の近接センシングの機能用にマークされるピンに接続します。CY8CMBR3xxx コントローラーでは、PSx ピンは近接センサーがそれに接続できることを示します。近接センシングには、ボタンやスライダーの使用時に比べてはるかに長い距離に放射される電界が必要であるために、大きなセンサー面積を要求します。近接センサーは、図 2-11 に示すように、PCB 上の配線であるか、シンプルなワイヤ ループです。詳細については、近接センサーのレイアウト ガイドライン節を参照してください。

図 2-11.近接センサー



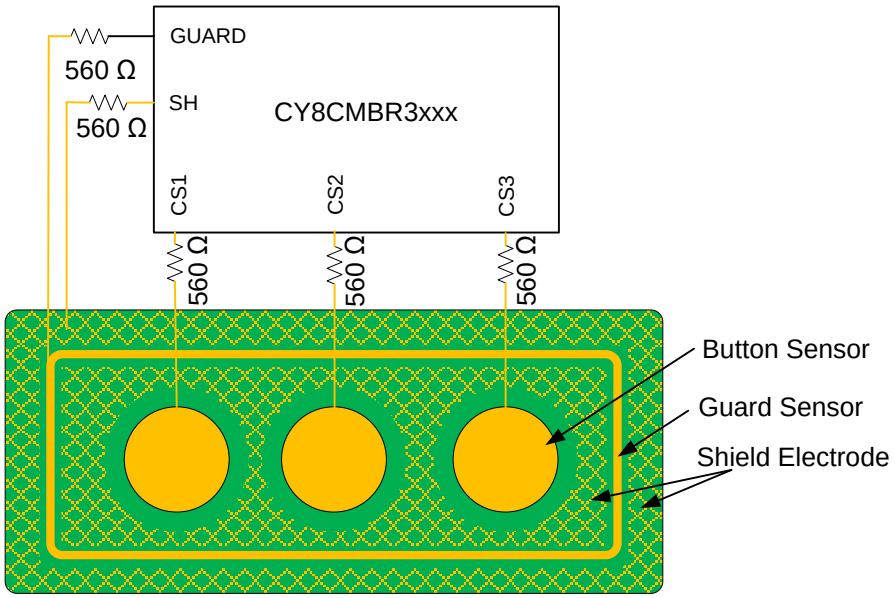
2.6 耐水性

CapSense の設計において、誤ったタッチ検知は、タッチ面上の液膜か液滴の存在に起因して起こることがあります。家電製品、車載アプリケーション、および産業用アプリケーションは、水、氷および湿度の変化を伴う環境で機能する必要があるシステムの例です。そのようなアプリケーションには、シールド電極およびガード センサーが堅牢なタッチスクリーンを提供します。

液滴や水分への耐性が必要な用途の場合は、シールド電極を使用する必要があります。接触面の液体流に対する耐性も要求する用途の場合は、シールド電極と共にガード センサーを使用する必要があります。シールドが有効になっていても、非アクティブなセンサーはグランドに接続されています。そのため、シールドが有効になっていても、1 つのセンサーに存在する液滴が誤ってトリガーされる原因とならないように、ボタンセンサーを遠く離して配置することを推奨します。

注: 耐水性はスライダー センサーではサポートされません。

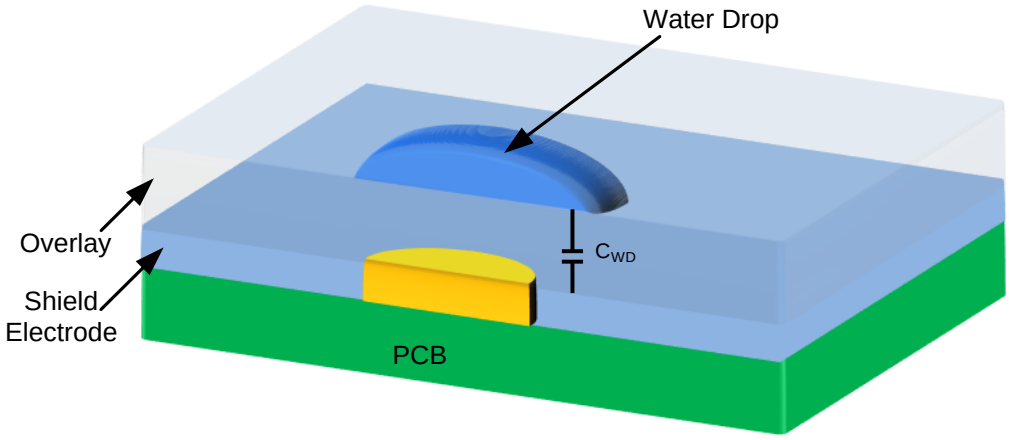
図 2-12.CY8CMBR3xxx に接続されるシールド電極(SH) 及びガード センサー(GUARD)



2.6.1 シールド電極

シールド電極は、CapSense ボタンセンサーが液滴による誤ったタッチを検出しないように保護します。オーバーレイの表面に液滴が存在する場合、図 2–13 に示すように、シールド電極とセンサー パッドの間のカップリングが C_{WD} の分だけ増加します。

図 2-13.液滴を伴う静電容量測定



■ C_{WD} – 液滴およびシールド電極間の静電容量

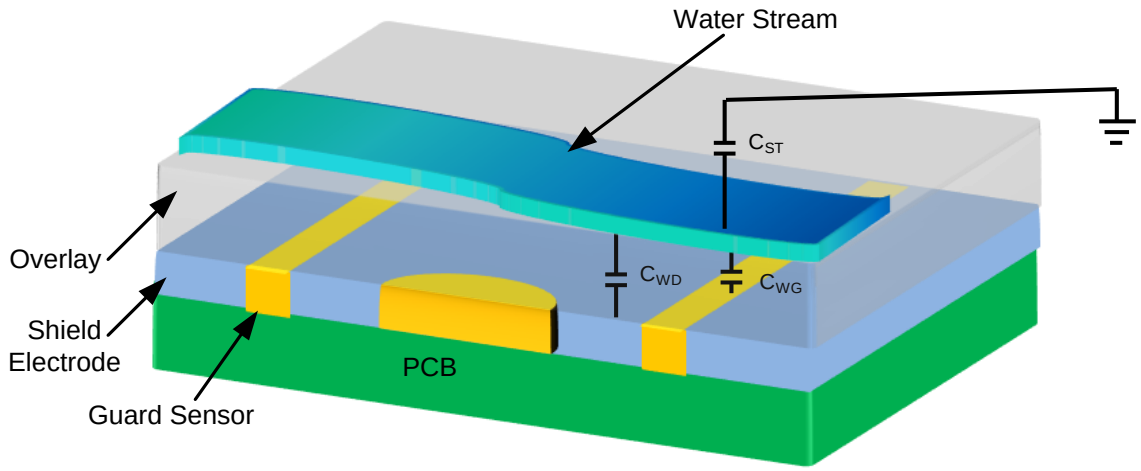
シールド電極の目的は、タッチ センサーの周囲に液体の影響の軽減に役立つ電場を設定することです。シールド電極は、タッチ センサーの電圧をミラーリングすることにより動作します。シールド電極およびセンサーが同じ信号で駆動されるため、それらの間の電位差はゼロであり、センサーとシールド電極との間のいかなる静電容量は電荷移動を引き起こすことができません。そのため、センサーとシールド領域の両方に部分的に存在している水膜または液滴が、センサーの容量を変化させず、CapSense は水膜または液滴があっても操作できるようになります。

2.6.2 ガード センサー

ガード センサーは、図 2-12 に示すように、PCB 上のすべてのセンサーを取り囲んでいる銅配線です。液体流の存在を検出するために使用されます。感知表面に液体流が存在する場合、図 2-14 に示すように、大きな静電容量 (C_{ST}) がシステムに追加されます。この静電容量は、C_{WD} の数倍大きくなる可能性があります。このため、シールド電極の効果は完全にマスクされており、センサーで測定される Raw カウントは、指のタッチと同じかそれより高くなります。この場合、ガードセンサーが液体流を検出すると、その他のセンサーがトリガーするのをブロックします。それで、液体を流す状況の場合はタッチを検出できません。

MBR3 では、ガードセンサー(が有効な場合)はアクティブ モードと Look-for-Touch モードでのみスキャンされます。そのため、デバイスが Look-for-Proximity モードのときは、液体の流れがあるときに誤った近接トリガーが発生する可能性があります。

図 2-14. 液体流を伴う静電容量測定



- C_{WD} - 液体流およびシールド電極間の静電容量
- C_{ST} - 液体流およびシステム グランド間の静電容量
- C_{WG} - 液体流およびガード センサー間の静電容量

設計に耐水性を持たせるために、以下の手順に従ってください。

1. アプリケーションは液滴や湿気に対する耐性が必要な用途の場合は、シールド機能を持つ CY8CMBR3xxx コントローラーを選択します。液体流に対しても耐性が必要な用途の場合は、シールドとガード機能の両方を持つ CY8CMBR3xxx コントローラーを選択します。コントローラーを選択するために、表 1-1 を参照してください。

2. シールド電極及びガード センサーをコンフィギュレーションするために、「耐水性のレイアウト ガイドライン」節で説明した回路図とレイアウト ガイドラインに従ってください。

3. シールドおよびガード センサーコンフィギュレーションの節で説明した通り、EZ-Click のシールドとガード センサーの機能を有効にします。

4. ボタン、スライダー、およびガード センサーのチューニング節で説明した通りにガード センサーを調整します。

3. CapSense の回路設計

CY8CMBR3xxx コントローラーは専用ピンのボタン、スライダー及び近接センサーをサポートします。サポートされているボタン、スライダー及び近接センサーの数はまたこのコントローラーに固定されます。センサー ピンの選択は CapSense アプリケーションを設計するための最初のステップです。この章では、CapSense の回路図を設計するためのセンサー ピンを選択するガイドラインを提供します。

3.1 回路図設計上の考慮事項

CY8CMBR3xxx で回路図を設計する手順は以下の通りです。

1. 特定の出力ピン (SPO) の選択
2. センサー ピンの選択
3. GPO ピンの選択
4. 電源及びグランド接続
5. I^2C の配線接続

この章は特定の出力、センサー及び GPO ピンの選択する方法について説明します。電源、グランド及び I^2C の配線接続については、CY8CMBR3xxx データシートを参照してください。

3.1.1 特定出力ピンの選択

シールド電極、ガード センサー、ブザー及びホスト割り込みピンは特定出力ピン (SPO) と呼ばれます。これらの機能はセンサー ピンと多重化され、専用のピンでのみサポートされます。そのため、センサー ピンを接続する前に、これらの特定出力ピンを選択するのは重要です。

例えば、7 ボタン、ブザー出力及び耐水機能を必要とする CapSense 設計を検討してください。表 1-1 の一覧のように、CY8CMBR3116 及び CY8CMBR3110 コントローラーは設計要求に適合します。例示の目的のために、CY8CMBR3110 が選択されます。ブザー (BUZ) 、シールド (SH) 及びガード (GUARD) ピンは図 3-1 に示すように、センサーと GPO ピンに多重化されるため、 回路図設計の最初のステップはこれらのピンを選択することです。同じピンでサポートされる複数の機能については、CY8CMBR3xxx データシートのピン配置の節を参照してください。特定の出力ピンが選択された後、図 3-2 に示すように、ボタン センサーCS0、CS1、CS3、CS5、CS6、CS7 及び CS8 を選択してください。

図 3-1. 単一のピンでサポートされる複数の機能

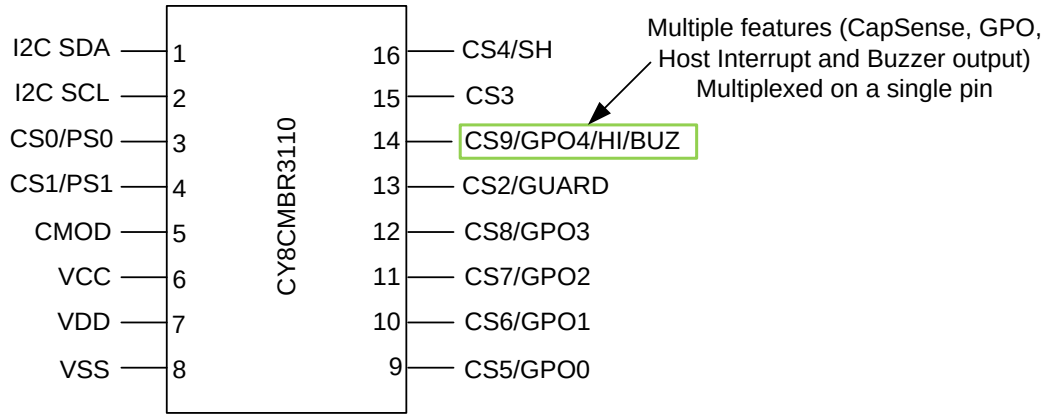
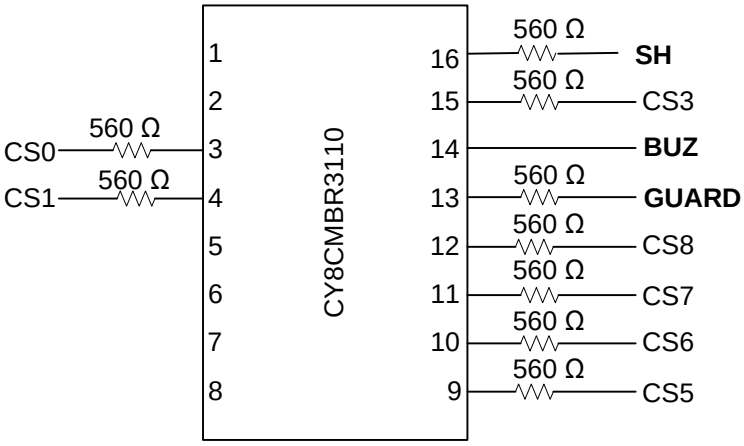


図 3-2.特定の出力ピン (SPO) の選択



3.1.2 センサー ピンを選択

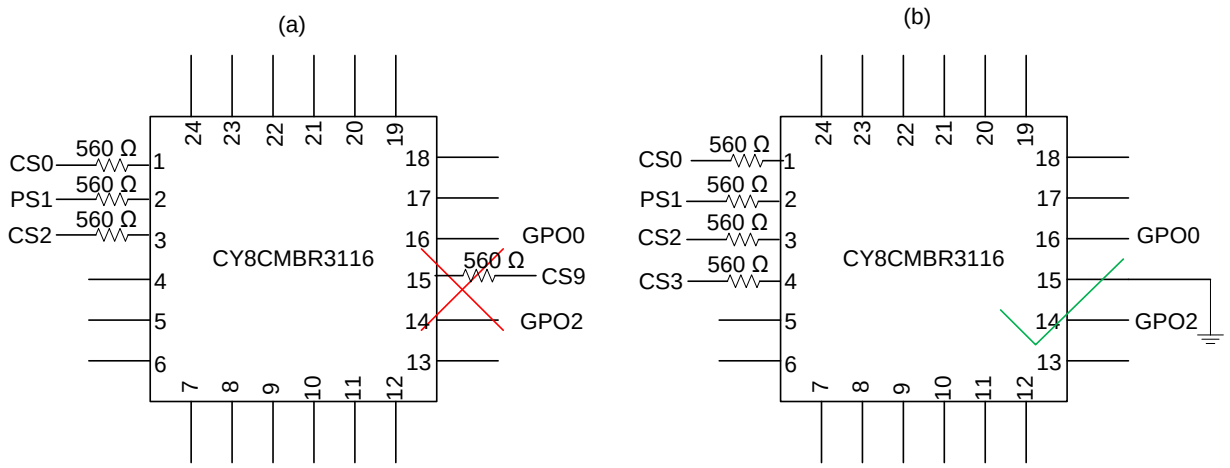
センサー ピンを選択する場合、GPO ピンの次、または GPO ピンの間のセンサー ピン (CSx、SLDx、PSx、または GUARD) を選択しないでください。GPO ピンの次にあるセンサー ピンを選択するとクロストークが発生する場合があります。センサー配線及び GPO 配線が並列に動作する場合は、GPO スイッチングは GPO ピンの次にあるセンサーを謝ったトリガーに引き起こすことがあります。クロストークを回避するために、CapSense 入門ガイドの「PCB レイアウト ガイドライン」節を参照してください。

例えば、図 3-3 (a) はセンサーCS9 が GPO0 と GPO2 ピンの間にある回路図です。この場合では、CS9 配線が GPO0 または GPO2 配線と並列に動作すると、GPO0 または GPO2 のスイッチングはセンサーCS9 を誤ったトリガーに引き起こします。この誤ったトリガーを回避するために、図 3-3 (b) に示したように、GPO ピンの次ではないセンサー ピン CS3 を選択するか、またはセンサー配線及び GPO 配線が並列に動作していないことを確保します。

ユーザーの設計は近接センサーを必要とする場合、近接センサーが 2 つのピンでのみサポートされるのでボタンピン (CSx) を選択する前に近接センサー ピン (PSx) を選択してください。

ノイズ耐性を高めるために、560Ωの直列抵抗をセンサー ピンに接続します。CapSense のいかなるピンも使用されていない場合、センサー ピンとして設定し、グランドに接続することを推奨します。CapSense 入力としてピンを設定する方法については、センサー コンフィギュレーション節を参照してください。

図 3-3. センサー ピンの選択ガイドライン



3.1.3 スライダー ピンを選択

スライダー ピンを選択するために、表 3-1.を参照してください。

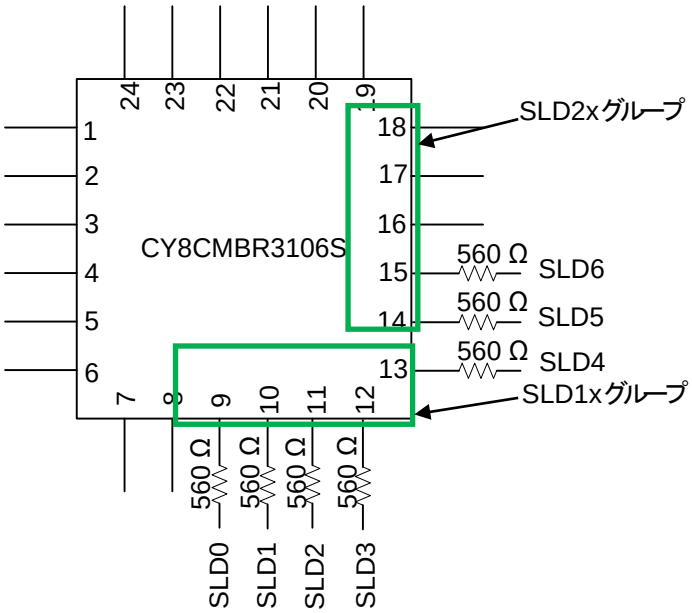
表 3-1.スライダー ピンの選択ガイドライン

セグメント数	推奨値
5 以下	SLD1x グループからピンを選択します。SLD10 は最初のスライダー セグメントであり、SLD14 は第 5 セグメントです。
5 以上	スライダーの最初の 5 セグメントのために SLD1x グループからのすべての 5 ピンを選択し、残りのセグメントのために SLD2x グループからのピンを選択します。SLD10 はスライダーの最初のセグメント、SLD20 は第 6 セグメントであり、SLD24 は第 10 セグメントです。

図 3-4 は 7 セグメントでスライダー用に設計された回路図の例を示します。この回路図では、スライダーのセグメント SLD0 ～ SLD4 はコントローラーの SLD1x グループからのピンを使用し、 SLD5 及び SLD6 セグメントは SLD2x グループのピンを使用します。SLD2x グループの残りのピン (SLD22、 SLD23、 and SLD24) はボタンとして設定されます。未使用の場合、ボタンとして設定され、グランドに接続されます[1]。

1 18 番目のピン(CY8CMBR3106S の場合は CS15/SLD24) は未使用時開放にする必要があります。この I/O は起動中にリセット (AXRES) ピンとして機能します。デバイスの正常なブートアップために、電源投入中にこのピンをグランドに接続しないでください。ブートアップ後、この I/O はピン名が示すように機能します。

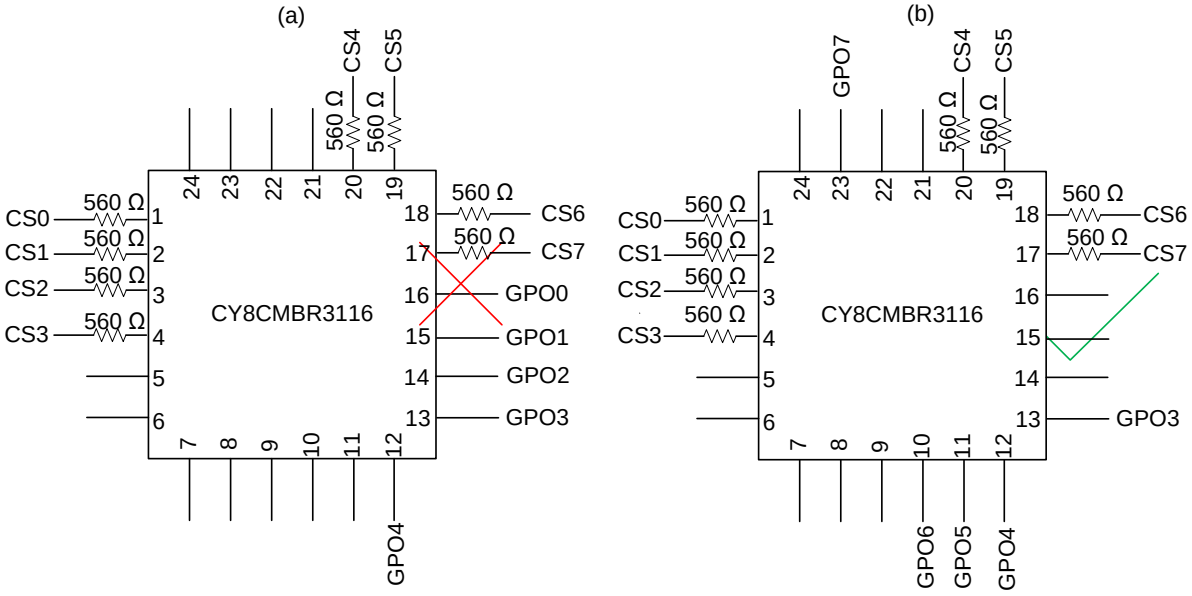
図 3-4.スライダー ピン選択



3.1.4 GPO を選択

GPO はホスト コントローラーによって制御される場合、任意の使用可能な GPO がタッチ フィードバックを提供するために使用されます。GPO7～GPO0 の順で GPOs を選択することを推奨します。例えば、ユーザーの設計は 8 ボタン及び 5 つの GPO を必要とする場合、ボタン用に CS0～CS7 を、 GPO 用に GPO7～GPO3 を選択します。これは、図 3-5 に示すように、GPO ピンの次にあるセンサー ピンの選択を回避するためです。

図 3-5. ホスト制御が有効にされた時の GPO ピンの選択



GPO ホスト制御が無効な場合、 CS0～CS7 からの各センサーはそれぞれに GPO0～GPO7 にマッピングされます。この場合、選択されたセンサーに対応する GPO を選択します。例えば、GPO ホスト制御は無効で、CS0、 CS1 及び CS2 はボタンとして選択された場合、LEDs を駆動するために GPO0、GPO1 及び GPO2 を GPO ピンとして選択してください。

CY8CMBR3xxx コントローラーは、GPO ピンの LED を駆動するためにソースおよびシンクコンフィギュレーションの両方をサポートします。GPO のシンク機能はソース機能よりはるかに高いので、シンクコンフィギュレーションで LED を接続することを推奨します。ソース及びシンクの最大電流制限については、CY8CMBR3xxx データシートの DC I/O ポートの仕様を参照してください。

電流を制限するには、供給源の電圧によって適当な値の直列抵抗を追加します。ユーザーの設計は最大制限値より大きなシンク電流が必要な場合、EZ-Click を使用して、外部トランジスタを GPO ピンに接続し、駆動モードで GPO を設定して下さい。

3.2 回路図の設計チェックリスト

回路図の設計を検証するために、表 3-2 のチェックリストを使用してください。

表 3-2. 回路図の設計チェックリスト

No.	カテゴリ	推奨／注釈
1.	VDD[2]ピンのデカップリング コンデンサ	1µF と 0.1µF のコンデンサを並列に接続する。
2.	VCC[3]ピンのデカップリング コンデンサ	VDD > 1.8Vの場合、0.1µFのデカップリング コンデンサに接続する。 VDDが1.71V～1.89Vの場合、このピンをVDDと短絡させる。
3.	VDDIO[4]ピンのデカップリング コンデンサ	1µF と 0.1µF のコンデンサを並列に接続する。
3.	C_{MOD}	2.2nF、5V、X7R または NPO コンデンサを接続する。
4.	CSx、PSx、SLDx、GUARD と SH ピンの直列抵抗	560Ω の抵抗をピンと直列に接続する。
5.	I^2C の配線上の直列抵抗	330Ω の抵抗を SDA[5]と SCL[6]線に直列に接続する。
6.	I^2C の配線のプルアップ	I^2C インターフェース電圧に応じて、適切な値を選択する。
7.	HI 線のプルアップ	I2C インターフェース電圧に応じて、適切な値を選択する。
8.	未使用の CapSense ピン[7] (CSx、PSx、SLDx と GUARD)	使用しない場合、ピン接続の詳細については CY8CMBR3xxx データシートのピン配置の節を参照してください。

3.3 回路図の例

3.3.1 携帯電話のタッチ ボタン

タッチ ボタンは、多くのアプリケーションに迅速にアクセスできるように携帯電話で使用されます。標準的携帯電話は最大 4 つのタッチ ボタンがあり、LED バックライトを必要とします。

図 3-6 に示す回路図はタッチ ボタンを携帯電話に実装する方法の例です。

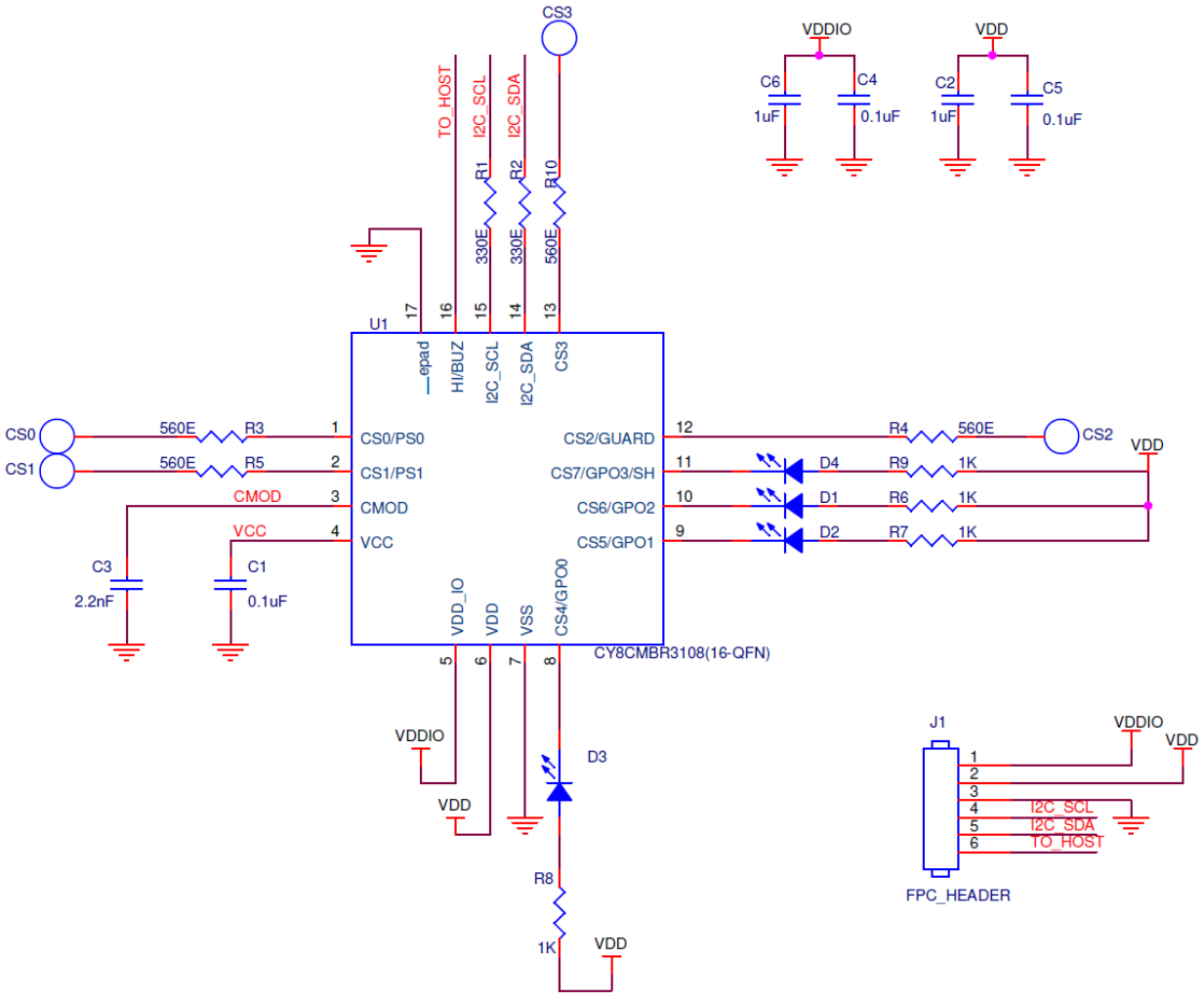
図 3-6 では、CY8CMBR3108 コントローラーは以下の方法で設定されます。

- CS0～CS3: CapSense ボタン
  - All ノイズ耐性を向上するためにすべての CapSense ピンは 560Ωの直列抵抗 (チップ近くに配置) を持つ必要があります。
- GPO0～GPO3: 外部 LED へ接続
  - CY8MBR3xxx コントローラーは HIGH 吸いこみ電流能力を持つため、各 LED がシンキング モードで接続される。
  - 直列抵抗によって、I_{IL} の限度内であるように GPO 電流を制限して接続します。
- CMOD ピン: 2.2nF のコンデンサを介してグランドに接続

[2] VDD: これはチップへの主電源であり、1.8V±5%か、または1.8～5.5Vで電源印加できます。
[3] VCC: これはコアと静電容量センシング回路に電源印加する内部レギュレータ出力です。
[4] VDDIO: これは、CY8CMBR3108のI^2C SDA、I^2C SCLとHI 線用の電源入力です。これらのI／Oの信号レベルはVDDIOを基準にしている。電源供給の詳細な情報については、CY8CMBR3xxx Datasheetをご参照ください。
[5] SDA: シリアル データ (SDA) 線、I^2CマスタとI^2Cスレーブとの間にデータを送信するために使用されます。
[6] SCL: シリアル クロック (SCL) 線は、I^2CスレーブをI^2Cマスタと同期化するために使用されます。
[7] 使用しない場合、ピン接続の詳細については CY8CMBR3xxx データシートのピン配置の節を参照してください。

- VCC ピン: 0.1µF のコンデンサを介してグランドに接続
  - 電源電圧は 3.3 V であるため、VCC ピンは VDD ピンと短絡されない。
- VDD ピン: 外部電源電圧へ接続
  - 1µFと0.1µFのデカップリング コンデンサはVDDへ接続
- VDDIOピン: 電源電圧 (≤ VDD) へ接続
  - VDDIOはI²CとHI配線に電源印加する。
- I²C_SCLとI²C_SDAピン: 330Ωの抵抗を介してI²Cのヘッダーに接続
- HIピン: ホストへ
  - 変化したセンサー状態を読み出すためにI²Cのトランザクションを初期化するようにホストに求めるため。

図 3-6. 携帯電話のタッチ ボタン用の回路図



注:

- コンデンサ C1、C2、C3、C4、C5、C6 及び 560Ω の直列抵抗をできる限り CapSense IC の近傍に配置してください。
- I²Cの配線 (SDAとSDL) 及びHIはホスト側の外部プルアップ抵抗を必要とします。

表 3-3. 部品表

No.	数量	リファレンス	説明	メーカー	メーカーの部品番号
1	2	C1、C4、C5	セラミック コンデンサ 0.1µF 10 V 10% X5R 0402	太陽誘電	LMK105BJ104KV-F
2	1	C2、C6	セラミック コンデンサ 1µF 10 V 10% X5R 0402	TDK 社	C1005X5R1A105K050BB
3	1	C3	セラミック コンデンサ 2200pF 50 V 5% X7R 0402	Murata 社	GRM155R71H222JA01J
4	2	R1、R2	RES 330Ω 1/10 W 5% 0402 SMD	Panasonic Electronic Components	ERJ-2GEJ331X
5	4	R3、R4、R5、R10	RES 560Ω 1/16 W 5% 0402 SMD	Yageo 社	RC0402JR-07560RL
6	4	R6、R7、R8、R9	RES 1kΩ 1/10 W 5% 0402 SMD	Panasonic Electronic Components	ERJ-2GEJ102X
7	1	U1	CY8CMBR3108-LQXI-16QFN	Cypress Semiconductor	CY8CMBR3108-LQXI - 16QFN
8	4	D3、D2、D1、D4	LED CHIP ANGAN HER RA 0603	Avago Technologies US Inc.	516-2277-1-ND

3.3.2 家電製品のタッチ ボタン

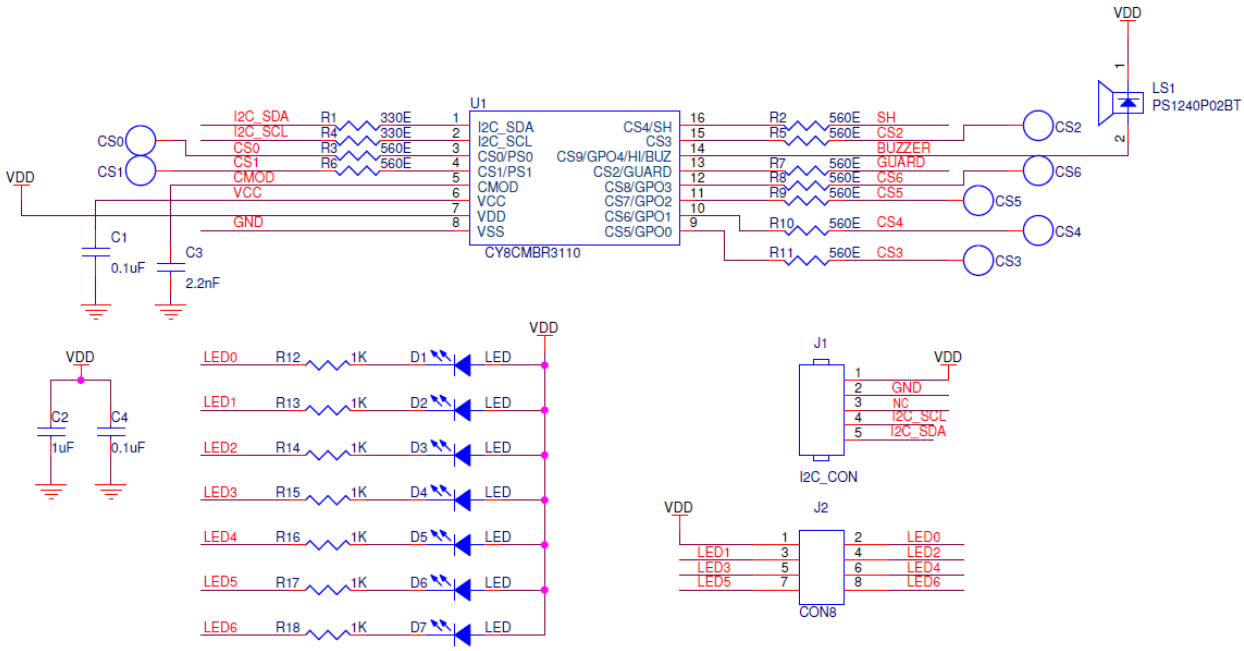
タッチ ボタンは、メカニカル ボタンに代わって電化製品に使用されます。これらのアプリケーションの重要な仕様は耐水性のある CapSense ボタンやオーディオ フィードバックです。

図 3-7 に示す回路図はタッチ ボタンを IH 調理器に実装する例です。

図 3-7 では、CY8CMBR3110 コントローラーは以下の方法で設定されます。

- CS0、CS1、CS3、CS5、CS6、CS7、CS8: CapSense のボタン
  - ノイズ耐性を向上するため、すべての CapSense ピンは (チップの近くに配置される) 560 Ω の直列抵抗を持つ必要がある。
- SH: シールド ピン
  - ノイズ耐性を向上するため、シールド ピンは (チップの近くに配置される) 560 Ω の直列抵抗を持つ必要がある。
- GUARD: ガード センサー
  - ノイズ耐性を向上するため、ガード センサーは (チップの近くに配置される) 560 Ω の直列抵抗を持つ必要がある。
- BUZZER: 1 ピンのブザーに接続。
- CMOD ピン: 2.2nF のコンデンサを通して接地
- VCC ピン: 0.1µF のコンデンサを通して接地 (デバイス)
  - 電源電圧は 5V であるため、VCC ピンは VDD ピンと短絡されない。
- VDD ピン: 外部電源電圧へ
  - 1 µF と 0.1のデカップリング コンデンサはVDDへ接続される。
- I^2C_SCLとI^2C_SDAピン: 330Ωの抵抗を通じて I^2Cヘッダーに接続

図 3-7. IH 調理器のタッチ ボタンの回路図



注:

- コンデンサ C1、C2、C3、C4 及び 560 Ω の直列抵抗をできる限り CapSense IC の近くに配置してください。
- この回路図に示されるLEDは、センサー状態に基づいてホスト コントローラーにより駆動さます。
- I²Cの配線 (SDAとSDL) はホスト側の外部プルアップ抵抗を必要とします。

表 3-4. 部品表

No.	数量	リファレンス	説明	メーカー	メーカーの型番
1	2	R1、R4	RES 330Ω 1/10W 5% 0603 SMD	Yageo 社	RC0603JR-07330RL
2	9	R3、R6、R2、R5、R7、R8、R9、R10、R11	RES 560Ω 1/10W 5% 0603 SMD	Yageo 社	RC0603JR-07560RL
3	1	C1、C4	セラミック コンデンサ 0.1µF 10 V 10% X5R 0603	Axios 社	0603ZD104KAT2A
4	1	C2	セラミック コンデンサ 1µF 10 V 10% X5R 0603	TDK 社	C1608X5R1A105K080AC
5	1	C3	セラミック コンデンサ 2200pF 50 V 5% NP0 0603	Murata 社	GRM1885C1H222JA01D
6	1	U1	CY8CMBR3110-SX2I-16SOIC	Cypress Semiconductor	CY8CMBR3110-SX2I-16SOIC
7	1	LS1	ブザー PIEZO 4KHZ 12.2MM PC MNT	TDK CORPORATION (VA)	PS1240P02BT
8	7	D1、D2、D3、D4、D5、D6、D7、D8	赤色 LED クリア 1206 背面実装タイプ SMD	スタンレー電気	BR1111R-TR
9	7	R12, R13, R14, R15, R16, R17, R18	RES 1.0kΩ 1/10 W 5% 0603 SMD	Yageo 社	RC0603JR-071KL
10	1	J1	CONN HDR 100" SNGL PCB 5POS	Sullins Connector Solutions	SWR25X-NRTC-S05-ST-BA

No.	数量	リファレンス	説明	メーカー	メーカーの型番
11	1	J2	CONN HEADER .100" DUAL STR 80POS	Sullins Connector Solutions	PRPC040DAAN-RC

3.3.3 近接センシング

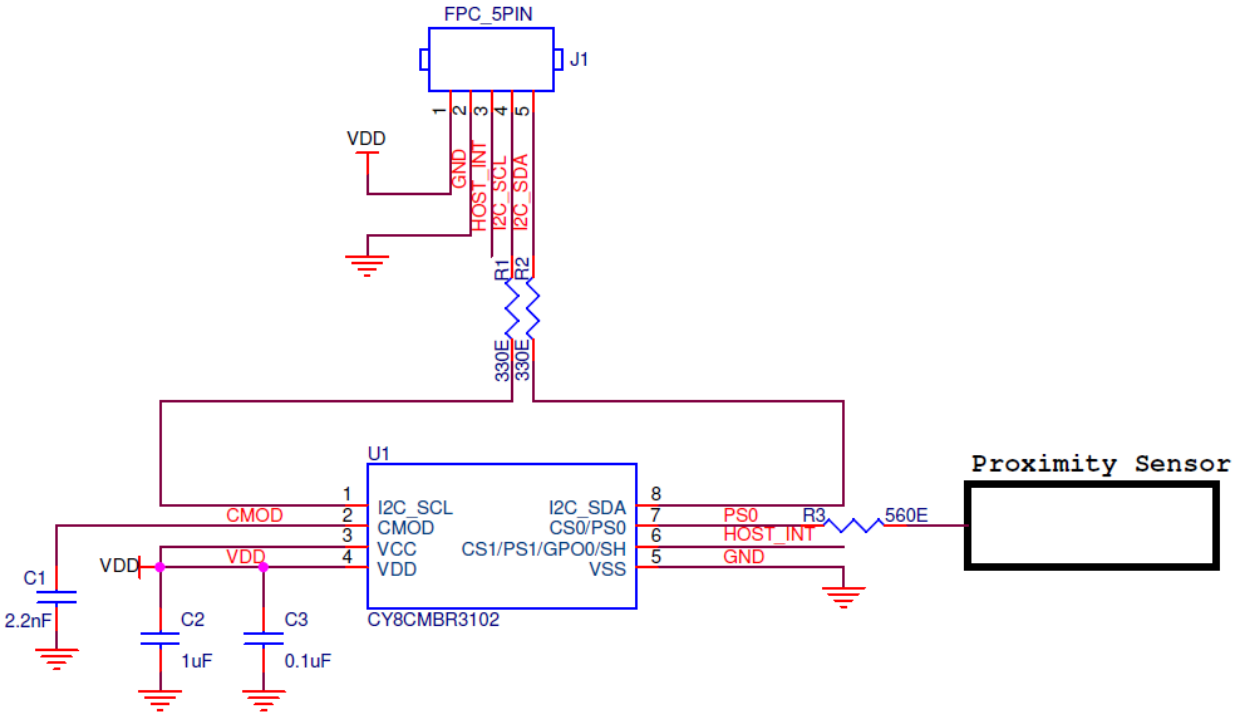
近接センシングは、ユーザーが電話中で、電話が耳の近い位置にある時、ディスプレーを OFF にし、キーパッドをディセーブルにするために、携帯電話・タブレットに使用されます。近接センシングは、携帯電話・タブレットの比吸収率 (SAR) 規制用にも使用されます。

図 3-8 に示される回路図は携帯電話・タブレットに近接センシングを実装する例です。

図 3-8 では、CY8CMBR3102 コントローラーは以下の方法で設定されます。

- PS0: 近接センサー
  - ノイズ耐性を向上するため、すべての近接センサー ピンは (チップの近くに配置される) 560 Ω の直列抵抗を持つ必要がある。
- GPO0: ホストへ
  - GPO は、ホスト コントローラーに近接センサー状態を示すために使用される。
- CMOD ピン: 2.2nF のコンデンサを介してグランドに接続
- VCC ピン: 1.71V～1.89V 動作の場合、VDD ピンと短絡
- VDD ピン: 外部電源電圧に接続
  - 1µFと0.1µFのデカップリング コンデンサはVDDへ接続
- I^2C_SCLとI^2C_SDAピン: 330Ωの抵抗を通じて I^2Cに接続

図 3-8.近接センサーアプリケーションの回路図



注:

- コンデンサ C1、C2、C3 及び 560Ω の直列抵抗をできる限り CapSense IC の近くに配置してください。
- I^2C 線 (SDA と SDL) と Host_INT はホスト側のプルアップ抵抗を必要とします。

表 3-5. 部品表

No.	数量	リファレンス	説明	メーカー	メーカーの部品番号
1	1	C1	セラミック コンデンサ 2200pF 50V 5% X7R 0402	Murata	GRM155R71H222JA01J
2	1	C2	セラミック コンデンサ 1µF 10V 10% X5R 0402	TDK Corporation	C1005X5R1A105K050BB
3	1	C3	セラミック コンデンサ 0.1µF 10V 10% X5R 0402	太陽誘電	LMK105BJ104KV-F
4	1	R3	RES 560Ω 1/16 W 5% 0402 SMD	Yageo 社	RC0402JR-07560RL
5	2	R1、R2	RES 330Ω 1/10 W 5% 0402 SMD	Panasonic Electronic Components	ERJ-2GEJ331X
6	1	U1	CY8CMBR3102-SX1I-8SOIC	Cypress Semiconductor	CY8CMBR3102-SX1I-8SOIC

4. CapSense レイアウト ガイドライン

標準的な CapSense アプリケーションでは、容量センサーは FR4/FR2 またはフレキシブル プリント基板 (FPC) 上の配線でコンフィギュレーションされています。CapSense レイアウト設計は、設計段階における重要なステップです。CapSense レイアウトの最善の実践方法に従って、設計が高いノイズ耐性、低い寄生容量 (C_P) 、および高い SNR を実現できます。

以下は、CapSense レイアウトを設計する際に考慮すべき要因です。

- 寄生容量 (C_P) : センサーの高い C_P は、センサーの静電容量の小さな変化を検知することをさらに難しくして、感度を低下させます。CapSense レイアウトは、センサーの C_P を最小限にするように設計する必要があります。
- 配線長: 長い配線のため、センサーの C_P が増加し、センサーの感度が低下します。また、長い配線がアンテナとして機能し、センサーのノイズ耐性を低下させます。
- センサー寸法: センサー寸法はオーバーレイの厚さに依存します。厚いオーバーレイは、大きなセンサー寸法を必要とします。
- 消費電力: センサーの消費電力は、センサーの C_P に依存します。センサーの高い C_P のため、センサーのスキャン時間が増加して、全体的な消費電力が多くなります。消費電力を削減するために、センサーの C_P を低下させます。

C_P の低下、SNR の増加、および消費電力の削減を実現するには、「レイアウト ベスト プラクティス」に従ってください。これらを実現する方法については、「CY8CMBR3xxx Design Toolbox」を参照してください。

4.1 Design Toolbox (デザイン ツールボックス)

CY8CMBR3xxx デザイン ツールボックスは、一般的なレイアウト ガイドライン、センサー寸法と寄生容量の見積もり、センサーの平均消費電力と応答時間を提供するスプレッドシートです。デザイン ツールボックスは、EZ-Click からアクセスすることができます。デザイン ツールボックスの最新バージョンは、「CY8CMBR3xxx Design Toolbox」からダウンロードすることができます。

このツールボックスには、4 つのセクションがあります。

1. General Layout Guidelines (全般的なレイアウト ガイドライン)
2. Layout Estimator (レイアウト エスティメータ)
3. C_P, Power Consumption, and Response Time Calculator (C_P、消費電力、応答時間計算器)
4. Design Validation (設計検証)

4.1.1 全般的なレイアウト ガイドライン

このシートには、センサーの形状、最大と最小寸法、配置、および PCB または FPC 上の配線ルーティングに関する推奨事項を記載してます。CapSense レイアウトを設計する際、これらのガイドラインに従ってください。

表 4-1 では、CapSense アプリケーションの設計のレイアウト ガイドラインをまとめています。ガイドラインの詳細ついては、「Getting Started with CapSense」ガイドを参照してください。

表 4-1.レイアウトの推奨事項

#	カテゴリ	詳細	Min	Max	推奨／注釈
1	ボタン	ボタン寄生容量 (C_P)	5pF	45pF	ボタンの C_P が低いほど、感度は上がり、消費電力は減少する
2		ボタンの形状	該当なし	該当なし	塗りつぶし円形パターン、LED 穴のある円形、または角が丸い長方形
3		ボタンのサイズ	5mm	15mm	「レイアウト エスティメータ」シートを参照
4		ボタン間の間隔	ボタンとグランドの間隔に等しい	該当なし	8mm
5		ボタンとグランドの間隔	0.5mm	2mm	オーバーレイの厚さに等しいが、最少／最大限度以内
6	スライダー	スライダー セグメントの寄生容量 (C_P)	5pF	45pF	スライダー セグメントの C_P が低いほど、感度は上がり、消費電力は減少する
7		スライダー セグメントの形状 (リニア スライダー)	該当なし	該当なし	ジグザグ パターン (二重の山形)
8		スライダー セグメントの幅	2mm[8]	8mm	8mm[9]
9		スライダー セグメントの高さ	7mm	15mm	12mm
10		セグメント間の間隔	0.5mm	2mm	0.5mm
11		ハッチとスライダー間のエアギャップ	0.5mm	2mm	オーバーレイの厚さに等しいが、エアギャップの最少／最大限以内
12	近接センサー	近接センサーの寄生容量 (C_P)	8pF	45pF	近接センサーの C_P が低いほど、近接検知範囲が広がる
13		近接センサーの形状	該当なし	該当なし	• PCB 上の広い面積を占める円形または角が丸い長方形ループ • 狭い面積を占める塗りつぶし円形または長方形 • 近接センサーをグランド ループで取り囲むことを推奨
14		近接センサーの配線幅	1.5mm		1.5mm
15		近接ループとグランド ループの間隔	1mm		1mm
16		グランド ループの配線幅	1.5mm		1.5mm

[8] 推奨の最小スライダー セグメント幅は使用するオーバーレイの厚さとオーバーレイ素材に依存します。所定のオーバーレイの厚さとオーバーレイ素材に応じる最小のスライダー セグメントの幅については、CapSense 入門の設計ガイドの「スライダーの設計」節を参照してください。

[9] 推奨されるスライダー セグメントの幅は人間の指の平均直径 9mm に基づいて計算します。詳細については、CapSense 入門の設計ガイドの「スライダーの設計」節を参照してください。

#	カテゴリ	詳細	Min	Max	推奨／注釈
17		近接センサー ループの直径	「推奨」列での説明の通り	「推奨」列での説明の通り	ALP フィルタが無効な場合、ループの直径は少なくとも必要な近接距離に等しい必要があります。ALP フィルタが有効な場合、ループの直径は少なくとも必要な近接距離の半分に等しい必要があります。近接ループ センサーの近くにあるグランド配線／グランド面が近接距離を減少させます。
18	**ガード**	ガード センサーの寄生容量 (C_P)	5pF	45pF	ガード センサーの C_P が低いほど、感度は上がり、消費電力は減少する
19		ガード センサーの形状	該当なし	該当なし	すべてのボタン センサーを取り囲む角が丸い長方形の PCB 配線
20		ガード センサーの厚さ	2mm		2mm
21	**グランド／シールド**	グランド／シールド領域 – 最上層	「推奨」列での説明の通り	「推奨」列での説明の通り	ハッチ パターン、0.17mm (7mil) の配線、1.143mm (45mil) のグリッド (25% フィル)
22		グランド／シールド領域 – 最下層	「推奨」列での説明の通り	「推奨」列での説明の通り	ハッチ パターン、0.17mm (7mil)の配線、1.778mm (70mil) のグリッド (17%フィル)。FPC における CapSense 設計の場合は、最下層でグランド フィルを使用しないでください。
23	**ルーティング**	2 層 PCB における部品の配置	該当なし	該当なし	• 最上層: センサー • 最下層: デバイス、他の部品、配線。
24		4 層 PCB における部品の配置	該当なし	該当なし	• 最上層: センサー • 第 2 層: CapSense 配線と V_{DD}。センサー下の V_{DD} 配線のルーティングを避けてください。 • 際 3 層: ハッチド グランド • 最下層: デバイス、他の部品、非 CapSense 配線。
25		センサー パッドとデバイス ピン間の配線長		520mm	520mm は、ボタン直径 5mm の FR4 PCB の場合の値です。異なる設計では、「レイアウト エスティメータ」シートを参照してください。配線が短いほど、センサーの C_P が小さくなります。
26		配線幅 (FR4/FR2/FPC)	0.17mm	0.20mm	0.17mm (7mil)
27		配線のルーティング	該当なし	該当なし	配線ルーティングは、センサーを配置しない側で実行する必要があります。任意の非 CapSense 配線が CapSense 配線に渡る場合、交差部分が直交であることを確保してください。
28		センサーのビア位置	該当なし	該当なし	ビアは、配線長を短くして、感度を高めるために、ボタンの端の近くに配置してください。
29		センサーの配線のビア ホール サイズ			10mil

#	カテゴリ	詳細	Min	Max	推奨／注釈
30		センサーの配線のビア数	1	2	1
31		CapSense 直列抵抗の配置		10mm	ノイズを抑制するために、CapSense 直列抵抗をデバイスの近くに配置してください。CapSense 抵抗は、LED 抵抗より高い優先順位を持っています。それらを先に配置してください。
32		CapSense 配線とグランド領域の距離	0.254mm (10mil)	0.508mm (20mil)	0.508mm (20mil)
33		デバイスの配置	該当なし	該当なし	センサーの逆側の層にデバイスを実装してください。デバイスとセンサー間の CapSense 配線長を最小限にする必要があります (上記の配線長を参照) 。
34		LED バックライト	該当なし	該当なし	センサー パッドに穴を開けて、背面取り付け可能な LED を使用します。
35		基板の厚さ	「推奨」列での説明の通り	「推奨」列での説明の通り	CapSense FR4 ベースの設計の標準的な基板の厚さは 1.6mm です。
36	**オーバーレイ**	オーバーレイの厚さ	設計要件の通り	5mm	「レイアウト エスティメータ」シートを使用してオーバーレイを決めてください。最大値は、プラスチック オーバーレイの場合の値です。
37		オーバーレイの素材	該当なし	該当なし	非導電性の素材 (ガラス、ABS プラスチック、フォーマイカ、木材など) を使用してください。プリント基板とオーバーレイ間にエア ギャップがないこと。接着剤を使ってプリント基板とオーバーレイを貼ってください。
38		オーバーレイ接着剤	該当なし	該当なし	接着剤は、非導電性で、誘電的に均質である必要があります。3M 製の 467MP と 468MP 接着剤を推奨します。

4.1.2 Layout Estimator (レイアウト エスティメータ)

図 4-1 に示すように、「レイアウト エスティメータ」シートは、最終システムの要件および工業設計に基づいて、最小ボタン サイズや最大配線長の推奨値を提供します。

このシートの入力は以下のパラメーターです。

- オーバーレイの厚さ: 設計で使用するオーバーレイの厚さを入力します。CY8CMBR3xxx は、ボタン センサーの場合に最大厚さ 5mm のプラスチック オーバーレイ、およびスライダー センサーの場合に最大厚さ 2mm のプラスチック オーバーレイをサポートしています。
- オーバーレイの比誘電率: オーバーレイの比誘電率を入力します。幾つかの素材の比誘電率については、同シートの表 C を参照してください。素材の比誘電率が高いほど、センサーの感度が高くなります。
- 1 インチ当たりの配線容量: 同シートの表 C を参照して、1 インチ当たりの配線容量を入力してください。FPC に比べると、FR4 PCB は小さい C_P を追加します。
- ボタンの感度: このパラメーターは、ボタン直径とオーバーレイ最大厚さを決めます。400fF を選択した場合、システムが最低感度に調整され、100fF を選択した場合、システムが最高感度に調整されます。
- スライダーの感度: 設計にスライダーがある場合、スライダーの感度を入力します。厚いオーバレイは、感度の値を小さくする必要があります。

- ノイズ状態: ノイズ状態は、センサー寸法を決める一つの要因です。ノイズが多い環境で設計を実行する必要がある場合、「High」を選択します。ノイズ状態が未定の場合、「Medium」を選択します。
- 近接センサー タイプ: 設計に近接センサーがある場合、近接センサー タイプを選択します。
- 必要な近接距離: 必要な近接距離を選択します。有効範囲は 1～30cm です。
- ALP フィルタ: 設計では、小さいループ サイズで長い近接距離を必要とする場合、「有効」を選択します。そうでない場合、「無効」を選択します。高級 LOW パス (ALP) フィルタが有効になった場合、近接センサーの応答時間が長くなります。

入力に応じて、ツールボックスは以下のものを見積もります。

- SNR 5:1 を達成するのに最小限必要なボタン直径: ツールボックスは、SNR 5:1 を達成するのに最小限必要なボタン直径が 15mm より大きい場合、オーバーレイの厚さを減らすか、またはセンサーの感度を高めるよう警告を表示します。CY8CMBR3002 CapSense コントローラーの場合、感度 パラメーターは 100fF に固定されます。センサーが過敏にならないように、推奨の最小ボタン直径から始める必要があります。タッチが検知されない場合、タッチ応答が最適になるまでボタン直径を大きくします。ボタン直径が大きすぎると、ホバー効果が発生し、指がセンサーに接触する前にオンになる可能性があります。
- SNR 5:1 を達成するのに最小限必要なスライダー寸法: ツールボックスは、スライダーの必要な寸法が範囲外の場合、警告を表示します。この場合、オーバーレイの厚さを減らすか、またはセンサーの感度を低めます。
- 設計におけるボタンとスライダー センサーの最大許容配線長。
- スライダー セグメント間の間隔、およびボタンとグランドの間隔。
- SNR 5:1 で必要な近接距離を達成するのに最小限必要な近接ループ直径。

図 4-1.レイアウト エスティメータ

Layout Estimator

Table A: Layout Estimator

Input Parameters	Value	Units	Comments
Overlay Thickness	2	mm	
Overlay - Dielectric constant	2.8	farad/m	
Capacitance of trace per inch	2	pF	
Button Sensitivity	200	fF	
Slider Sensitivity	100	fF	
Noise Condition	Medium		
Minimum Recommended Button Diameter	8	mm	
Minimum Recommended Slider Width	2	mm	
Minimum Recommended Slider Height	14	mm	
Maximum Trace Length for Button	508	mm	
Maximum Trace Length for Slider Segments	520	mm	
Clearance Between Slider Segments	2	mm	
Button to Ground clearance	2	mm	

Table B: Proximity Sensor Layout Estimator

Input Parameters	Value	Units	Comments
Proximity Sensor Type	PCB TRACE		
Required Proximity Distance	5	cm	
ALP Filter	Disabled		
Minimum Recommended Proximity Sensor Loop Diameter	5	cm	

Table C: Industry Standard Reference Values

Overlay Material	Dielectric constant
Plastic	2.8
Plexi glass	8
Formica	4.6 - 4.9
Glass (Standard)	7.6 - 8.0
Glass (Ceramic)	6
Mylar	3.2
ABS Plastic	3.8 - 4.5
Wood	1.2 - 2.5
Trace and board type	**Capacitance per inch in pF**
Copper trace , PCB, 2 layer, 64m	2
Copper trace, flex PCB, 2 layer	8

Note: Search online for the dielectric constant of the materials which are not listed in Table C.

	input cells, edit with actuals
	output cells, based on inputs

4.1.3 C_P, Power Consumption, and Response Time Calculator (C_P、消費電力、応答時間計算器)

基板レイアウトを完了した後、「C_P,、消費電力、応答時間計算器」シートを使用して、設計の C_P、平均消費電力、および応答時間を見積もることができます。

4.1.3.1 C_P 値の見積もり

センサーの C_P を見積もる手順は以下の通りです。

1. 表 A では、ドロップダウン リストから型番を選択します。
2. 選択した部品番号が「CY8C3106S」であり、スライダーが設計に必要な場合、スライダー数および各スライダー グループ内のセグメント数を入力します。
3. 表 B では、「Sensor Type」列でセンサー タイプを選択します。どのセンサーにも接続されないか、または選択した部品番号に適用できないピンの場合は、NC (接続なし) を選択します。ピンの詳細については、CY8CMBR3xxx データシートを参照してください。
4. 選択したセンサーの寸法を入力します。「Slider Length」列は、スライダー セグメントにのみ適用できます。選択したセンサーの寸法が最大範囲を超えた場合、ツールボックスは、センサー寸法を減少させるよう警告を表示します。センサー寸法の最小値と最大値については、「全般的なレイアウト ガイドライン」を参照してください。
5. 各センサーの配線長を「Trace Length」列に入力します。
6. すべてのパラメーターが入力された後、ツールボックスはセンサーの C_P 値を見積もって、「Parasitic capacitance (C_P) of sensors」列に表示します。積もりの C_P 値が 45pF より大きい場合、C_P 値が 45pF より小さくなるまで配線長を減少させます。その値がが最大範囲以内の場合、センサー寸法を減少させることは推奨しません。

図 4-2.センサーC_P の見積もり

Table B: Cp Calculator

Sensor	Sensor Dimension				Trace Length		Sensitivity		Parasitic capacitance (Cp) of sensors (Approx)		Comments
	Slider Length		Button Diameter/Slider Width/Proximity Loop								
PS0			10	mm	50	mm	100	fF	7	pF	
PS1			10	mm	55	mm	100	fF	7	pF	
CS2			10	mm	58	mm	100	fF	10	pF	
CS3			10	mm	56	mm	100	fF	10	pF	
CS4			10	mm	54	mm	100	fF	10	pF	
CS5			10	mm	57	mm	100	fF	10	pF	
SLD10	12	mm	2	mm	62	mm	100	fF	9	pF	
SLD11	12	mm	2	mm	64	mm	100	fF	9	pF	
SLD12	12	mm	2	mm	67	mm	100	fF	10	pF	
SLD13	12	mm	2	mm	69	mm	100	fF	10	pF	
SLD14	12	mm	2	mm	71	mm	100	fF	10	pF	
SLD20	10	mm	2	mm	45	mm	100	fF	7	pF	
SLD21	10	mm	2	mm	47	mm	100	fF	7	pF	
SLD22	10	mm	2	mm	49	mm	100	fF	8	pF	
SLD23	10	mm	2	mm	52	mm	100	fF	8	pF	
SLD24	10	mm	2	mm	54	mm	100	fF	8	pF	

4.1.3.2 平均電流の見積もり

CP 値の見積もり節で説明されたようにセンサー寸法を入力した後、コントローラーの平均消費電流を見積もる手順は以下の通りです。

1. 表 B では、「Sensitivity」列から各センサーの感度を選択します。感度値が大きいほど、消費電流が少なくなります。
2. 表 C では、スキャン期間を選択します。スキャン期間値が大きいほど、平均消費電流が少なくなります。
3. 状態タイムアウト期間を選択します。タイムアウト期間値が小さいほど、消費電流が少なくなります。この値を、設計の要件に応じて選択してください。
4. EMC が有効な場合、「Enabled」を選択します。無効な場合、「Disabled」を選択します。
5. ブザーが有効な場合、ブザー オン時間を選択します。無効な場合、0 を選択します。ブザーが有効になった場合は、ブザー オン時間値が大きいほど、平均消費電流が多くなります。
6. PWM が GPO で有効な場合、「Enabled」を選択します。無効な場合、「Disabled」を選択します。
7. 1 時間当たりの平均接触回数および各センサー接触の期間を入力します。この値は、ユーザーがセンサーを接触する頻度に依存します。

8. すべてのパラメーターが入力された後、ツールボックスは図 4-3 に示されるように、見積もりの 1 時間当たりの平均消費電流を表示します。

図 4-3.平均消費電流の見積もり

Table C: Power Calculator

Scan Period	200	ms
Time Out Period	8	s
EMC	Enabled	
Buzzer ON Time	0	ms
PWM on GPOs	Disabled	
Wake On Proximity	Disabled	
Average Number of Button Touch per Hour	10	
Average Button Touch Time	100	ms
Estimated Average Current Consumption	1.08	mA

4.1.3.3 応答時間の見積もり

センサーの応答時間は、表 B と表 C に入力された値に依存します。ボタンとスライダーの応答時間を見積もるために、4.1.3.1 節と 4.1.3.2 節それぞれで説明されたように、表 B と表 C のパラメーターを入力します。

ボタンとスライダーの応答時間を見積もる手順は以下の通りです。

1. 表 D では、デバウンス パラメーター値を選択します。
2. 無限インパルス応答 (IIR) とメジアン フィルタが有効な場合、それらに対応するセルで「Enabled」を選択します。無効な場合、「Disabled」を選択します。
3. これらのパラメーターが入力された後、図 4-4 に示されたように、ツールボックスは、ボタンとスライダーそれぞれの最初の接触応答時間と連続した接触応答時間を表示します。

近接センサーの応答時間とデバイスのタイミング パラメーターを見積もる手順は以下の通りです。

1. 近接センサーの応答時間を異なる動作モードで見積もるために、コントローラーの動作状態を選択します。各動作モードの詳細については、「CY8CMBR3xxx の動作モード」節を参照してください。
2. 高級 LOW パス (ALP) フィルタが有効な場合、「Enabled」を選択します。無効な場合、「Disabled」を選択します。
3. デバウンス パラメーター値を選択します。
4. これらのパラメーターが入力された後、ツールボックスは、図 4-4 に示されたように、近接検知と近接リリースの応答時間を表示します。
5. LED 継続時間の最小値と最大値を見積もるためには、LED の持続時間値を選択してください。

図 4-4.応答時間の見積もり

Table D: Response Time Calculator

Button and Slider Touch and Release Response Time		
Debounce	3	
IIR Filter	Enabled	
Median Filter	Enabled	
Response time for first button touch	754	ms
Response time for consecutive button touch	268	ms
Response time for first slider touch	665	ms
Response time for consecutive slider touch	178	ms
Response time for button/slider release	178	ms

Proximity Sensor Response Time				
Operating State	Look-for-Touch			
ALP Filter	Disabled			
Debounce	3			
	Minimum	Typical	Maximum	
Response time for proximity detection	354	354	354	ms
Response time for proximity release	400	400	400	ms

Device Timing Parameters		
LED Duration	500	ms
LED Duration (Minimum)	500	ms
LED Duration (Maximum)	594	ms
GPO Control Status Delay	200	ms
Device Wakeup Time	50	ms

4.1.4 設計検証

試作基板が作られて試験された後、「設計検証」シートを使用して設計を検証します。「設計検証」シートを使用するには、パラメーターを「CP, Power Consumption, and Response Time Calculator」シートの表 B に入力する必要があります。

「設計検証」シートは、ボタンとスライダー センサーの設計のみを検証します。近接センサーの設計が様々な外部要因に依存するため、このシートは近接センサーを検証せず、推奨を提供しません。

設計を検証する手順が以下の通りです。

1. EZ-Click を使用して、ボタンとスライダーの raw カウント、ノイズ、および C_P 値を測定します。
2. 図 4-5 に示されたように、各センサーの raw カウント、ノイズ、および C_P 値を表 C に入力します。
3. ツールボックスはこれらのパラメーターに基づいて、推奨のセンサー寸法を提供します。「CP, Power Consumption, and Response Time Calculator」シートの表 B に入力されたセンサー寸法が推奨のセンサー寸法に等しいかそれ以上である場合、設計が「Pass」として示されます。そうでない場合、「Fail」として示されます。

設計が合格した場合、試作段階が終了して、生産段階の準備ができています。

注: 設計に近接センサーがある場合、ツールボックスは近接センサーの推奨を提供せず、全体の設計を検証しません。

設計が不合格の場合、以下の手順を行ってください。

1. 表 B では、感度パラメーター値を下げます。
2. 設計がまだ不合格する場合、図 4-6 のように、表 A で、オーバーレイの厚さを減少させます。

3. 感度値およびオーバーレイの厚さを減少させても設計がまだ不合格の場合、図 4-7 のように、表 C の推奨値に基づいて表 B に新しいセンサー寸法を入力します。

注: レイアウトは、表 C の推奨値に基づいて再設計する必要があります。

図 4-5.設計検証—センサー寸法の推奨値表

Table C: Sensor Dimension Recommendation

Sensor	Values taken from I2C						Improvement Recommendations						
	Noise		Rawcount		Cp		Recommended Sensor Dimension				Comments	Maximum Trace Length	
PS0	20	counts	45000	counts	12	pF				mm			mm
PS1	25	counts	45400	counts	13	pF				mm			mm
CS2	10	counts	3000	counts	12	pF			6	mm		520	mm
CS3	10	counts	3000	counts	12	pF			6	mm		520	mm
CS4	10	counts	3000	counts	13	pF			6	mm		520	mm
CS5	10	counts	3000	counts	15	pF			7	mm		514	mm
SLD10	12	counts	3100	counts	9	pF	14	mm	2	mm		520	mm
SLD11	13	counts	3125	counts	10	pF	14	mm	2	mm		520	mm
SLD12	15	counts	3150	counts	10	pF	14	mm	2	mm		520	mm
SLD13	14	counts	3140	counts	9	pF	14	mm	2	mm		520	mm
SLD14	12	counts	3170	counts	10	pF	14	mm	2	mm		520	mm
SLD20	16	counts	3158	counts	8	pF	14	mm	2	mm		520	mm
SLD21	12	counts	3100	counts	7	pF	14	mm	2	mm		520	mm
SLD22	14	counts	3140	counts	8	pF	14	mm	2	mm		520	mm
SLD23	13	counts	3125	counts	9	pF	14	mm	2	mm		520	mm
SLD24	16	counts	3158	counts	7	pF	14	mm	2	mm		520	mm

図 4-6. 設計検証—実際の設計値表

新しいオーバーレイの厚さ

Table A: Actual Design Values

Input Parameters	Initial Value	New Value	Units
Overlay Thickness	2	1	mm
Overlay - Dielectric constant	2.8		farad/m
Capacitance of trace per inch	2		pF
Scan Period	200	200	ms
Time Out Period	8	8	s
EMC	Enabled	Enabled	
Buzzer ON Time	0	0	ms
Wake On Proximity	Disabled	Disabled	
Average Number of Button Touch per	10	10	
Average Button Touch Time	100	100	ms

図 4-7. 設計検証—実際のセンサー寸法値表

Table B: Actual Sensor Dimensions

Sensor	Sensor Dimension								Sensitivity			
	Slider Length		Button Diameter/Slider Width/Proximity Loop Diameter		Slider Length		Button Diameter/Slider Width/Proximity Loop Diameter					
	Initial Value				New Value				Initial Value		New Value	
CS0			10	mm				mm	100	fF		fF
CS1			10	mm								fF
CS2			10	mm								fF
CS3			10	mm								fF
CS4			10	mm								fF
CS5			10	mm				mm	100	fF		fF
SLD10	12	mm	2	mm	14	mm	2	mm	100	fF		fF
SLD11	12	mm	2	mm	14	mm	2	mm	100	fF		fF
SLD12	12	mm	2	mm	14	mm	2	mm	100	fF		fF
SLD13	12	mm	2	mm	14	mm	2	mm	100	fF		fF
SLD14	12	mm	2	mm	14	mm	2	mm	100	fF		fF
SLD20	10	mm	2	mm	14	mm	2	mm	100	fF		fF
SLD21	10	mm	2	mm	14	mm	2	mm	100	fF		fF
SLD22	10	mm	2	mm	14	mm	2	mm	100	fF		fF
SLD23	10	mm	2	mm	14	mm	2	mm	100	fF		fF
SLD24	10	mm	2	mm	14	mm	2	mm	100	fF		fF

新しいセンサー寸法値

4.2 近接センサーのレイアウト ガイドライン

静電容量近接センサーは、物理的接触なしに隣接する導電性物体を検知します。近接センサーは、センサー構造とセンサー感度の2つの観点でボタン センサーとは異なります。近接センサーは、もっと長い近接距離を得るために、電界線が最大距離まで発生させるようにコンフィギュレーションする必要があります。近接距離は、センサーの電界伝播 (電界の強度) に依存します。伝播距離が長いほど、検知範囲が長くなります。

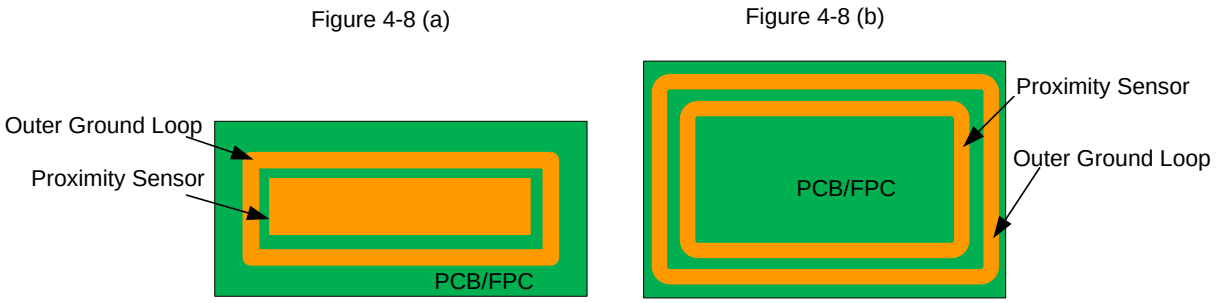
4.2.1 センサーの設計ガイドライン

近接センサーの構造は、所望のアプリケーションおよび所要の近接範囲に依存します。携帯電話の顔近接検出やタブレットの比吸収率 (SAR) の調整などの応用は、最大3cmの近接距離を必要とします。この近接距離を得るには、狭い面積を占める塗りつぶし形状のセンサーは十分です。フォト フレーム、モニタ、キーボードなどの長い近接距離を必要とするアプリケーションに対しては、デバイスの周囲を取り囲む大きいなセンサー ループを使用します。

以下のものを使用して、近接センサーをコンフィギュレーションすることができます。

- PCBまたはFPC上の銅配線: 長いPCB配線は近接センサーとして使用することができます。 配線は、直線であるか、または図4-8のようにシステムのユーザー インターフェースの周囲を囲むことがあります。PCB配線による近接センサーのコンフィギュレーション方法は、大量生産には適していますが、ワイヤ センサーほど感度は高くありません。 ノイズを低減させ、ESD イベントに対する耐性を提供するために、近接センサーをグランド ループで取り囲むことを推奨しています。近接範囲とノイズ耐性およびESD性能間にトレードオフがあります。近接センサーおよびグランド ループ レイアウトの推奨については、設計ツールボックスの「全般的なレイアウト ガイドライン」シートを参照してください。

図 4-8. PCB/FPC 上の近接センサーのコンフィギュレーション

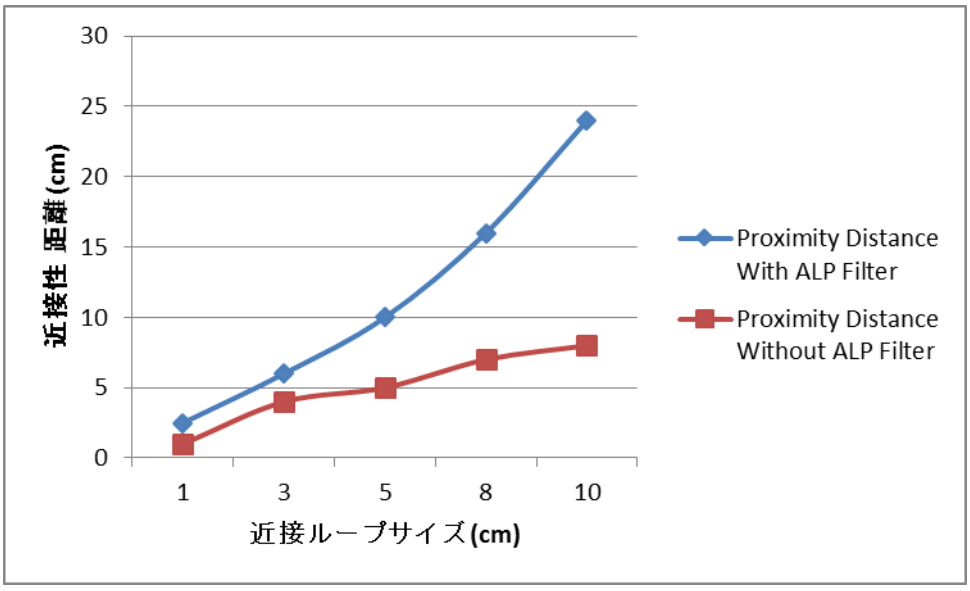


- ワイヤ: 単一長さのワイヤは、近接センサーとして正常に機能します。手の検出は電界の変化による静電容量の変化に依存しているため、ワイヤの周囲の電界に影響を与える浮遊静電容量または物体は近接センサーの範囲に影響を与えます。ワイヤセンサーの使用は、製造コストや複雑さから、大量生産に関する最適なソリューションではありません。

近接範囲を増加するために、以下のガイドラインに従います。

- 近接センサー ループ サイズを大きくします: 図4-9に示されたように、近接センサー範囲が大きいほど、近接距離が長くなります。
- 近接分解能を最大値に設定します: 近接センサー分解能が高いほど、近接距離が長くなります。
- 高級LOWパス (ALP) フィルタを有効にします: 図4-9に示されたように、ALPフィルタは近接センサーのrawカウントのノイズを減衰させ、近接距離を長くします。
- 近接センサーをすべての隣接する金属物体から保護します: 近接センサーの近くに置かれた金属物体は、感度を低くし、近接距離を短くします。金属物体の影響を無くし、近接距離を長くするために、シールド電極を使用します。

図 4-9.近接ループ サイズと近接距離の関係



注:

- 図 4-9 に示された近接距離は理想的な条件で測定されます。実際の適用では、この距離は動作環境によって減少することがあります。
- 図 4-9 に示された近接距離は、近接センサーがグランド ループによって取り囲まれている条件で測定されます。近接ループとグランド ループの間隔は 1mm に維持されます。

4.2.2 近接距離への金属物体の影響

センサーの近くに金属面があると、近接検知範囲は大幅に減少します。多くの家庭電化製品と車載用アプリケーションに金属フレームまたはケースがあるため、これらのアプリケーションでは大きい近接範囲を達成するのは困難です。

導電性物体がセンサーの近くに置かれた時、以下の因数により近接検知範囲は大幅に減少します。

- センサーの寄生容量が増加します。大きい浮遊容量は、センサーの動作周波数を減少させて、近接距離を減少させることが多いです。
- 接地された金属面がセンサーの電界の一部をキャッチして、手のひらで追加された容量を低下します。

図4-12のように、近接センサーと金属物体の間にシールド電極を配置することで、センサーに対する隣接する金属面の影響を減少させることができます。

図 4-10.金属物体がない場合の単一センサーコンフィギュレーションの電界伝播

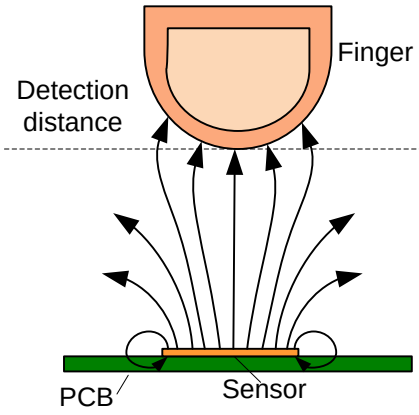


図 4-11. 固体金属物体がある場合の単一センサーコンフィギュレーションの電界伝播

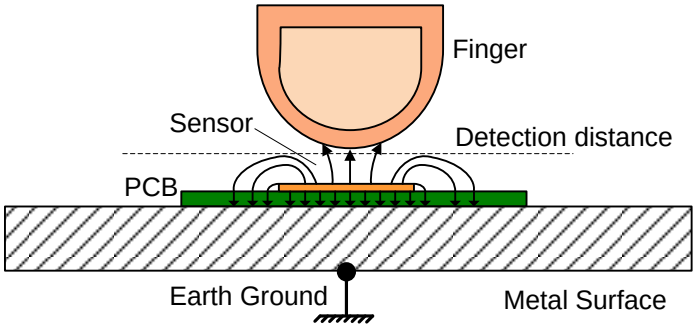
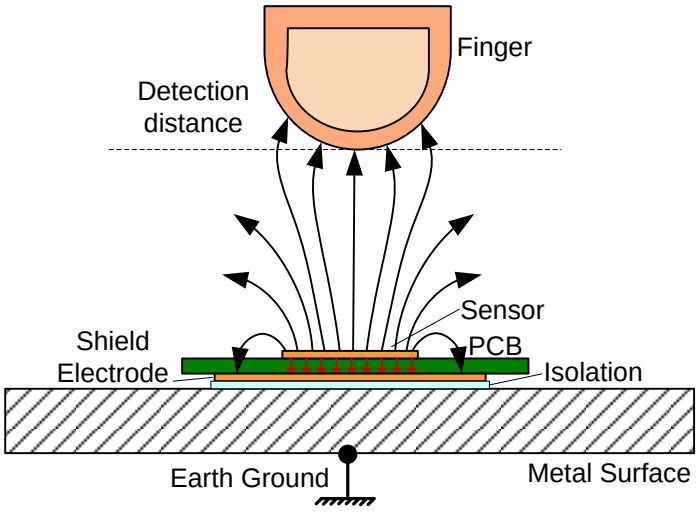


図 4-12. シールド電極による金属物体の影響の制限



4.3 耐水性のレイアウト ガイドライン

オーバーレイ上に液滴または連続液体流があると、容量性接触センサーが誤ってトリガーする可能性があります。液滴に起因したセンサーへの誤った接触を防ぐために、シールド電極を使用します。同様に、連続液体流に起因した誤った接触を防ぐために、ガード センサーを使用します。シールドとガード センサーの動作については、「耐水性」節を参照してください。

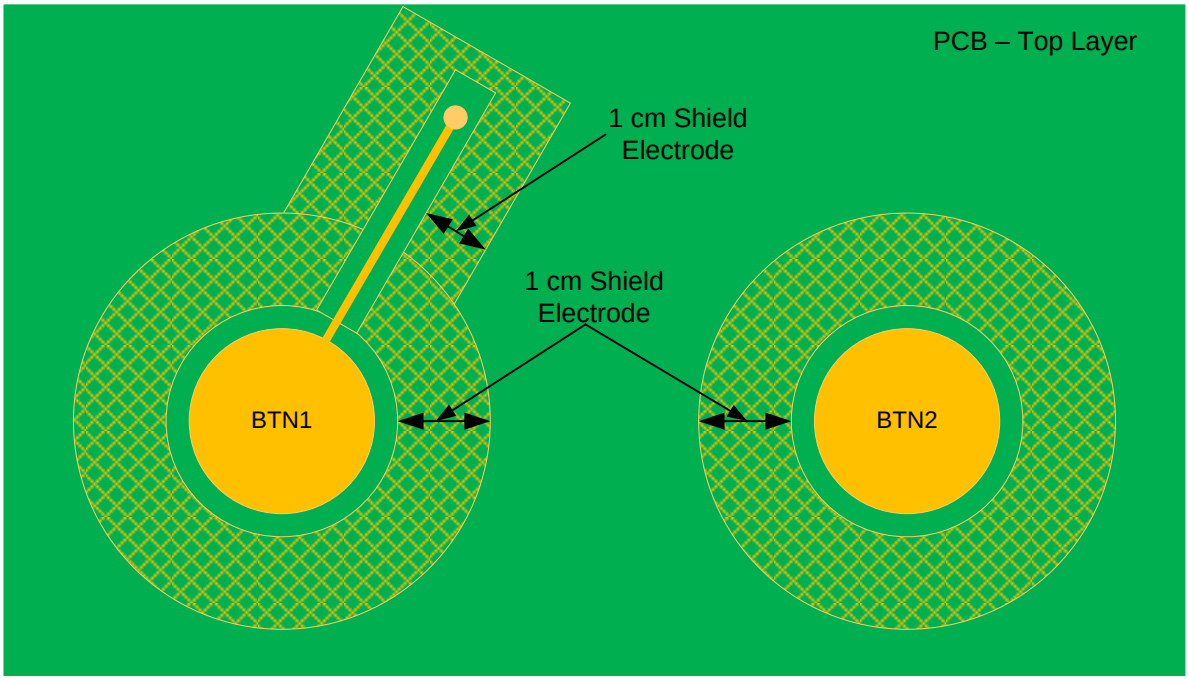
4.3.1 シールド電極の構造

耐水性容量センシング アプリケーションを設計する際は、以下のガイドラインに留意しておいてください。

- 0.17mm (7 mil) の配線と 1.143mm (45 mil) のグリッドのハッチ グランドを PCB の最上層で使用し、このハッチ パターンをシールド信号 (SH ピン) で駆動します。
- センサー パッドおよび曝される配線 (1 層プリント基板の場合) をシールド電極で取り囲みます。

  シールド電極をセンサーから 1 cm 以内に広げます。シールド電極を 1 cm を越えて広げると、システム性能に少し影響し、放射エミッションを増加させる可能性があります。基板面積が非常に広い場合は、図 4-13 に示すように、1cm のシールド電極の外側の領域は、空のままにする必要があります。アプリケーションによっては、プリント基板上の電極シールドの実装に十分な領域がないことがあります。このような場合、シールド電極を 1 cm 以内で広げることができます; センサー実装後の基板上の残りの領域を最小限のシールド エリアとすることができます。

図 4-13. シールド電極パターン



2 層と 4 層プリント基板にシールド電極を実装するためには以下のガイドラインに従ってください。

2 層プリント基板

- 最上層: 7mil 配線と 45mil グリッドのハッチ (25%フィル) 。ハッチは被駆動シールド信号に接続する必要があります。
- 最下層: 7mil 配線と 70mil グリッドのハッチ (17%フィル) 。ハッチは被駆動シールド信号に接続する必要があります。

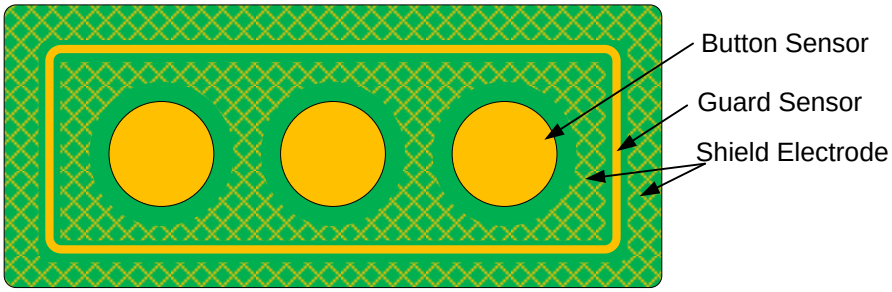
- 4 層プリント基板:
  - 最上層: 7mil 配線と 45mil グリッドのハッチ (25%フィル) 。ハッチは被駆動シールド信号に接続する必要があります。
  - 第 2 層: 7mil 配線と 70mil グリッドのハッチ (17%フィル) 。ハッチは被駆動シールド信号に接続する必要があります。
  - 第 3 層: V_{DD} 面
  - 最下層: 7mil 配線と 70mil グリッドのハッチ (17%フィル) 。ハッチはグランドに接続する必要があります。

センサーとシールド電極間の推奨エアギャップは 1mm です。

4.3.2 ガード センサー

図 4-14 に示されたように、ガード センサーは、すべてのセンサーを取り囲んでいる銅配線です。シールド電極パターンは、ガード センサーと露出している配線を取り囲み、これらから 1cm 以内に広げることを保証する必要があります。ガード センサーは、角が丸い長方形でなければなりません。ガード センサーの推奨厚さは 2mm で、ガード配線とシールド ハッチ間の最大距離は 1mm です。

図 4-14.シールド電極とガード センサーのある PCB レイアウト



4.4 サンプル レイアウト

4.4.1 FPC 上のタッチ ボタン

図 4-15 と図 4-16 には、FPC 上にタッチ ボタンを実装するよう設計されたレイアウトの最上層と最下層を示します。この設計の回路図は、図 3-6 を参照してください。

このレイアウトでは、以下のベスト プラクティスに従います。

1. 最上層のみにブランドを施すことで、センサーの C_P を最小限にします。
2. チップをセンサーの近くに配置することで、センサーの配線長を最小限にします。
3. VDD、LED、および I^2C ラインをセンサー配線から分離します。
4. 最高感度を得るために、センサーの中心にある間隔 (LED バックライティングの場合) を最小限にします。

図 4-15.FPC 上のタッチ ボタン—最上層

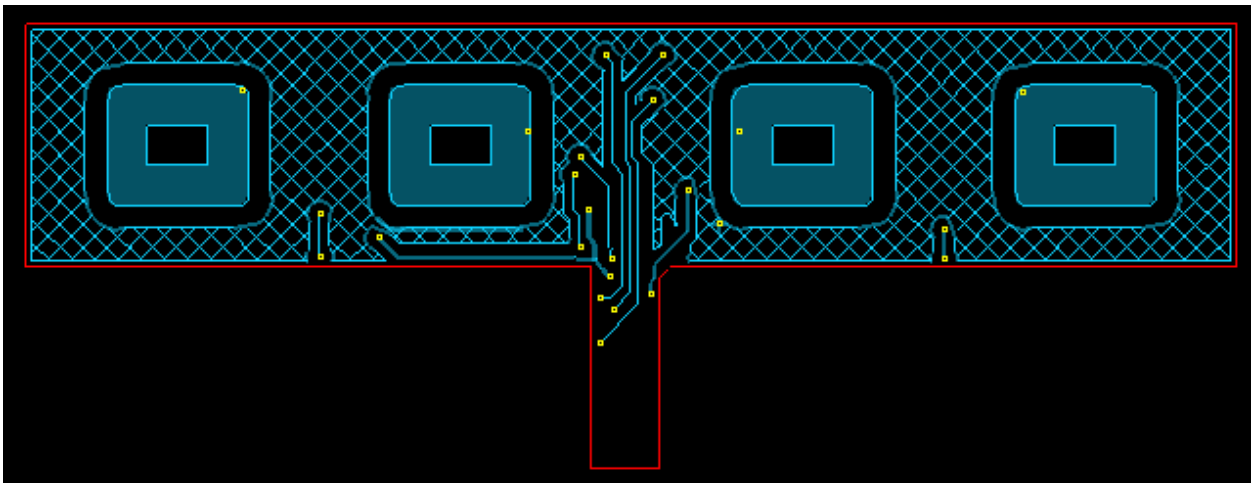
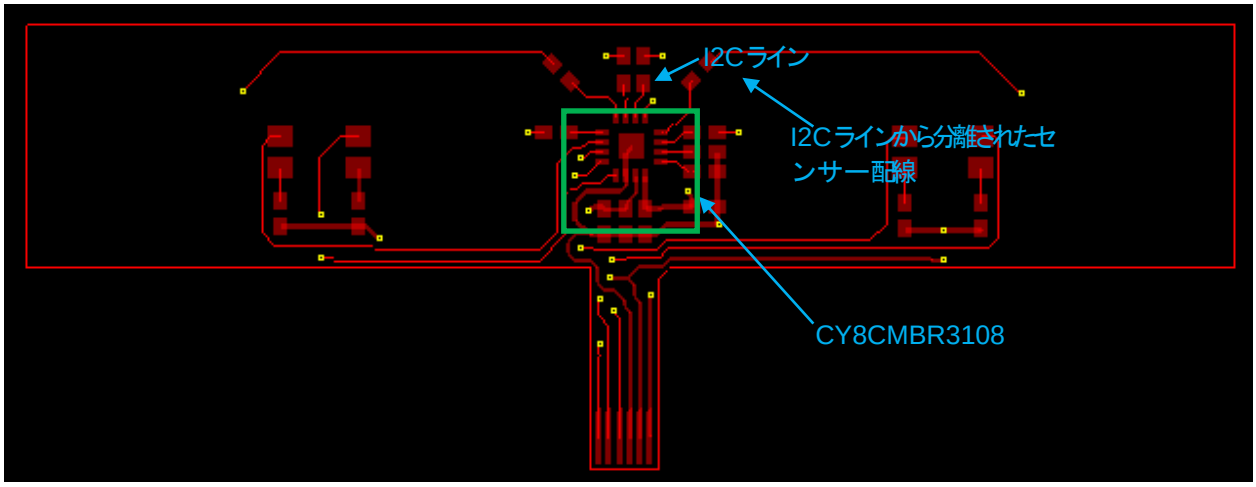


図 4-16. FPC 上のタッチ ボタン—最下層



4.4.2 FR4 PCB 上のタッチ ボタン

図 4-17 と図 4-18 には、FR4 PCB 上に耐水性タッチ ボタンを実装するよう設計されたレイアウトの最上層と最下層を示します。この設計の回路図は、図 3-7 を参照してください。

このレイアウトでは、以下のベスト プラクティスに従います。

1. シールドとガード センサーは、「耐水性のレイアウト ガイドライン」節で説明されたようにコンフィギュレーションします。
2. チップをセンサーの近くに配置することで、センサーの配線長を最小限にします。
3. VDD、LED、および I^2C ラインをセンサー配線から分離します。
4. 最高感度を得るために、センサーの中心にある間隔 (LED バックライティングの場合) を最小限にします。

図 4-17.FR4 プリント基板上の耐水性タッチ ボタン – 最上層

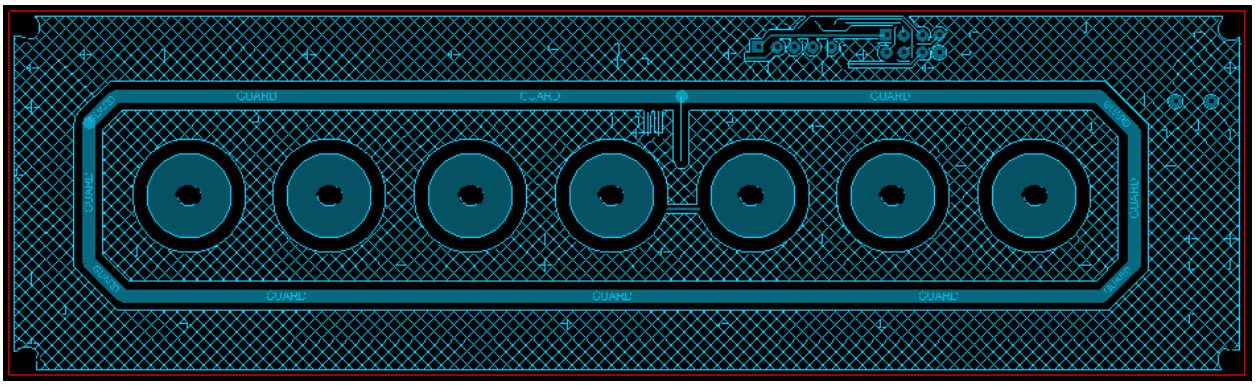
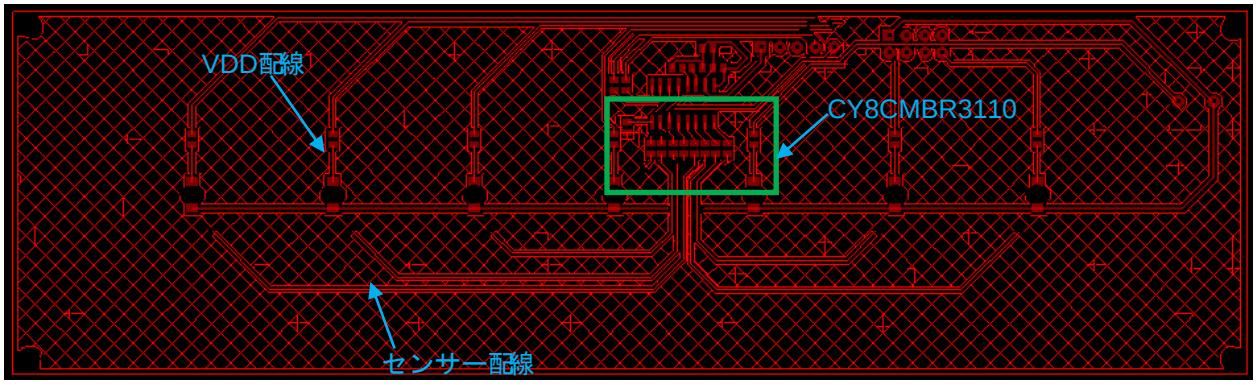


図 4-18. FR4 プリント基板上の耐水性タッチ ボタン – 最下層



4.4.3 FPC 上の近接センサー

図 4-19 と図 4-20 には、FPC 上に近接センサーを実装するよう設計されたレイアウトの最上層と最下層を示します。この設計の回路図は、図 3-8 を参照してください。

このレイアウトでは、以下のベスト プラクティスに従います。

1. ノイズ耐性および ESD 性能を向上させるために、近接センサー ループをグランド ループで取り囲みます。
2. ループ センサーを使用することで、センサーの C_P を最小限にします。
3. ノイズを最小限にするために、近接センサーをグランド リングで取り囲みます。ESD イベントに対する耐性を提供するのにも役立ちます。
4. チップをセンサーの近くに配置することで、センサーの配線長を最小限にします。
5. VDD および I^2C ラインをセンサー配線から分離します。

図 4-19.FPC 上の近接センサー—最上層

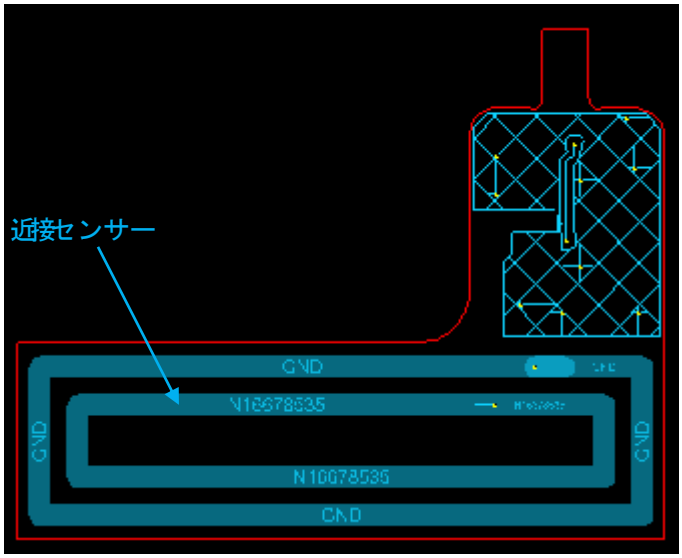
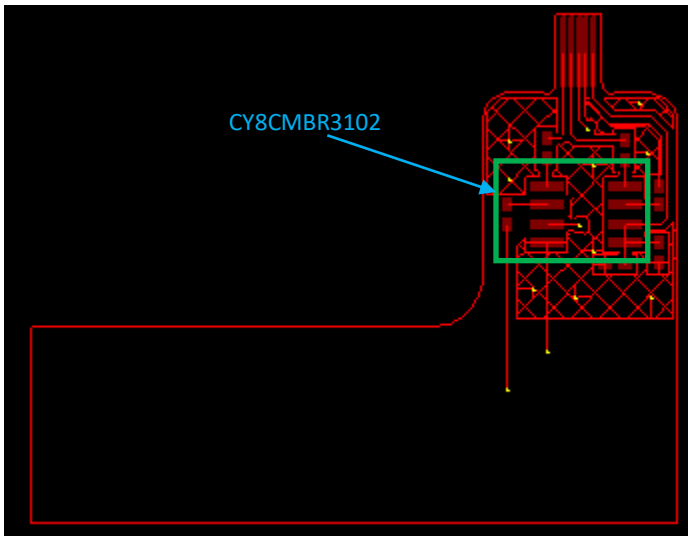


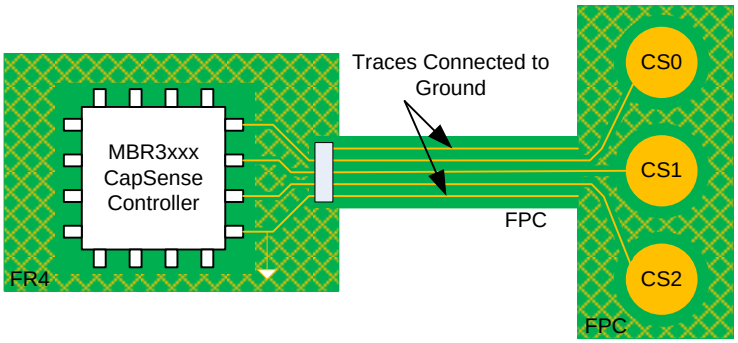
図 4-20. FPC 上の近接センサー—最下層



4.4.4 FR4 と FPC 上での CapSense 設計

図 4-21 に CapSense センサーが FPC に実装し、CapSense コントローラーが FR4 プリント基板に配置する CapSense のデザインを示します。CapSense センサーは FPC コネクタを使用してコントローラーに接続されます。

図 4-21. FR4 と FPC 上での CapSense 設計



センサーが FPC に実装され、CapSense コントローラーが FR4 プリント基板に配置される CapSense を設計するためには、以下のガイドラインに従ってください:

1. FPC では、センサー寄生容量を最小化するために、最上層のみにハッチ フィルを使用してください。ハッチ フィルはグランドまたは被駆動シールド信号のいずれかに接続します。最上層の推奨ハッチ フィルは 7mil の配線と 45mil のグリッドです。
2. FPC コネクタでセンサー配線を介する CapSense システムへのノイズの伝導を最小化するために、図 4-21 に示すように、センサー配線を短くし、センサー配線と平行にグランド配線を配置します。
3. FR4 プリント基板では、CapSense コントローラーを最上層に配置し、未使用の領域にはグランド接続のハッチ フィルを使用することができます。最上層の推奨ハッチ フィルは 7mil の配線と 45mil のグリッドです。最下層では、7mil の配線と 70mil のグリッドのハッチ グランドを使用することができます。

5. CY8CMBR3xxx のコンフィギュレーション

CY8CMBR3xxx は、I²C のコンフィギュレーション可能な CapSense コントローラーです。このコントローラは様々な機能に対応しており、EZ-Click を使用して容易に設定されます (EZ-Click は簡単かつ直観的なグラフィカル ユーザー インターフェース ツールです)。CY8CMBR3002 を除く CY8CMBR3xxx ファミリのすべてのコントローラーは EZ-Click を使用してコンフィギュレーションすることができます。CY8CMBR3002 コントローラーは固定機能デバイスであり、コンフィギュレーション不可です。次の節では、CY8CMBR3xxx ファミリでサポートされる様々なコンフィギュレーションとそれらのコンフィギュレーション方法について説明します。

5.1 CY8CMBR3xxx のコンフィギュレーション

CY8CMBR3xxx コントローラーでサポートされるコンフィギュレーションは、以下の 4 つのカテゴリに分かれています。

1. センサー コンフィギュレーション
2. GPO コンフィギュレーション
3. ブザー コンフィギュレーション
4. グローバル デバイス コンフィギュレーション

5.1.1 センサー コンフィギュレーション

5.1.1.1 感度制御

この感度パラメーターは、センサー信号の強度 (差分カウント) を増加させるか減少させるのに使用されます。低い感度値 (100fF) をセットすることにより、センサー信号の強度が高くなりますが、応答時間および消費電力は増加します。異なる使用可能な感度のセットは 100fF、200fF、300fF、および 400fF です。感度パラメーターは、ボタン センサー、ガード センサー、およびスライダーのグループそれぞれに対してコンフィギュレーションすることができます。感度パラメーターは近接センサーには適用できません。

以下の要因はセンサーの感度に影響を与えます。

1. オーバーレイの厚さ: 厚いオーバーレイは、感度の値を小さくすることが必要です。
2. システム ノイズ: システム ノイズが増加すると、センサーの誤ったトリガーを防止するために、感度を低くする必要があります。
3. 設計の形状の要因: 相対的に大きいセンサー サイズは、感度値を高くすることが必要です。より小さいサイズのセンサーは、感度値を低くすることが必要です。
4. 消費電力: 消費電力が多くなると、感度値が低くなります。低い消費電力は、感度値を高くすることが必要です。
   消費電力は、100fF>200fF>300fF>400fF の順に減少します。

感度パラメーターは、EZ-Click の「CapSense sensor Configuration」タブでセットします。感度パラメーターをセットする方法については、「CapSense 性能のチューニング」節を参照してください。CY8CMBR3002 CapSense コントローラーでは CapSense 性能のチューニングこのパラメーターは 100fF に固定され、コンフィギュレーション不可です。

5.1.1.2 近接分解能

近接距離は近接分解能パラメーターに依存します。近接分解能が高いほど、近接距離が長くなります。分解能パラメーターは、それぞれの近接センサーに対して指定する必要があります。このパラメーターは 12～16 ビットの範囲内です。近接分解能パラメーターは、EZ-Click の「CapSense sensor Configuration」タブでコンフィギュレーションします。このパラメーターのコンフィギュレーション方法については、「CapSense 性能のチューニング」節を参照してください。CY8CMBR3002 CapSense コントローラーにはこのパラメーターがありません。

5.1.1.3 スライダー分解能

スライダーの分解能は、コントローラーが分解できるユニークな位置数を示します。スライダーは、分解能が高いほど、分解できる位置数が多くなります。このパラメーターは CY8CMBR3106S コントローラーにのみ有効です。CY8CMBR3106S は最大 2 スライダー グループをサポートします: SLD1x グループと SLD2x グループ。分解能パラメーターは、それぞれのスライダー グループに対して指定する必要があります。しかし、両方のスライダーを 1 つに結合した場合、分解能を 1 つだけ指定します。このパラメーター値は 1～254 の範囲内です。スライダー分解能パラメーターは、EZ-Click の「CapSense sensor Configuration」タブで設定します。

5.1.1.4 電磁環境適合性 (EMC)

EMC 機能は、外部の放射および伝導ノイズに対するデバイスの耐性を決定します。そのノイズは、電力増幅器からの可聴周波数ノイズ、無線送信機からの無線周波数ノイズ、静電気放電や電力線サージ等です。ノイズが酷い環境で動作するシステムに対しては、EMC 機能を有効にします。EMC 機能が有効になると、消費電力および応答時間が増加します。

CY8CMBR3106S ベースの設計では、EMC 機能はセンサーCS0/PS0、CS1/PS1、および CS2～CS5 のセンサーにのみ適用されます。CS11～CS15 のすべてのスライダー セグメントおよびボタンは EMC なしに動作します。

CY8CMBR3116 ベースの設計では、EMC 機能はセンサーCS0/PS0、CS1/PS1、および CS2～CS9 のセンサーにしか適用できます。

以下の場合には、EMC 機能を無効にします。

- CS10～CS15 の任意のボタンが CY8CMBR3116 コントローラーで有効になりました。
- フィルタが有効になりました (CY8CMBR3106S ベースの設計でのみ)
- 設計が低ノイズ環境で実行されます
- ブザーが有効になりました

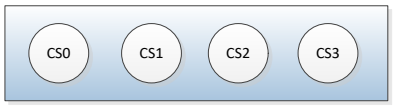
EMC 機能は、EZ-Click の「Global Configuration」タブで有効にします。CY8CMBR3002 CapSense コントローラーでは EMC 機能は常に無効にされます。

5.1.1.5 隣接センサー抑制 (FSS)

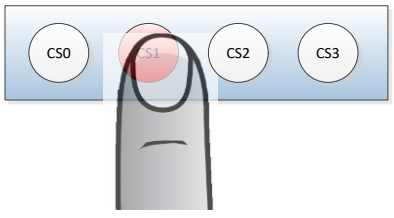
FSS 機能は、複数の隣接したボタンからのセンサー接触を識別して、誤検出を除去します。これにより、複数のセンサーが触れられた時にシステムが最初の触れられたボタンのみを認識することを保証できます。FSS は図 5-2 のように、いつでも 1 個のボタンだけがオンとして報告されることを保証します。もし指が複数のボタンに触れても、接触状態を感知した最初の 1 個だけがオンとして報告されます。もし複数のボタンが同時に触れられた場合、小さい番号のセンサーはオンとして報告されます。

FSS 機能はボタンにのみ適用され、EZ-Click の「CapSense sensor Configuration」タブでそれぞれのボタンに対して有効か無効にします。CY8CMBR3002 CapSense コントローラーには FSS 機能がありません。

図 5-1.1 個のボタンのみが触れられた場合の FSS

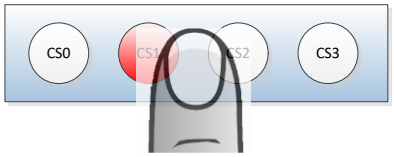


接触の前にオンになったボタンがない

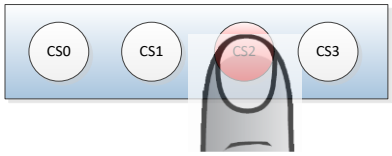


接触の時に CS1 はオンとして報告される

図 5-2.1 個のボタンが既にオンになっている状態で複数のボタンが触れられた場合の FSS



CS1 と CS2 両方が触れられた時に CS1 はオンとして報告される



CS2 のみが触れられた時に CS2 はオンとして報告される

5.1.1.6 ファームウェア フィルタ

CY8CMBR3xxx コントローラーは、センサーの raw カウントのノイズを減少させるために以下のフィルタをサポートしています。

- メジアン フィルタ
- IIR フィルタ
- 高級 LOW パス (ALP) フィルタ

メジアンおよび IIR フィルタはボタン、スライダー、およびガード センサーにのみ適用できます。ALP フィルタは近接センサーにのみ適用できます。

CY8CMBR3106S ベースの設計では、以下の注意事項に留意してください。

- ALP フィルタはサポートされていません
- EMC 機能が有効になった場合、メディアンおよび IIR フィルタを無効にします

5.1.1.6.1 メディアン フィルタ

メディアン フィルタは、*N* サイズのバッファからメディアン値を計算する非線形フィルタです。このフィルタは、センサーの raw カウントからスパイク ノイズを取り除きます。CY8CMBR3xxx コントローラーに実行されたメディアン フィルタは三次移動メジアン フィルタです。このフィルタの用途については、「CapSense 性能のチューニング」節を参照してください。

メディアン フィルターが有効になった場合、自動閾値機能を無効にし、指閾値パラメーターを手動で指定することをお勧めします。CY8CMBR3002 CapSense コントローラーのメディアン フィルターは常に有効にされます。

5.1.1.6.2 無限インパルス応答 (IIR) フィルタ

IIR フィルタは、RC フィルタと同様なステップ応答を生成します。このフィルタは、センサーの raw カウントから高周波ホワイト ノイズを減衰させます。このフィルタの用途については、「CapSense 性能のチューニング」節を参照してください。CY8CMBR3002 CapSense コントローラーの IIR フィルターは常に有効にされます。

メディアンと IIR フィルタが有効になった場合、メディアン フィルタは先に raw カウントに適用され、それに続いて IIR フィルタが適用されます。

5.1.1.6.3 アドバンスト ロー パス (ALP) フィルタ

ALP フィルタは、近接センサーの raw カウントのノイズを減衰させます。長い近接距離を得るには、近接センサーは高い分解能を必要とします。しかし、高い分解能設定の結果としては、raw カウントのノイズが増加します。この高周波ノイズを減衰させ、5:1 より高い SNR を得るために、ALP フィルタを有効にします。ALP フィルタの用途、および有効になった時にそれの調整する方法については、「近接センサーのチューニング」節を参照してください。CY8CMBR3106S と CY8CMBR3002 CapSense コントローラーではこの機能はありません。

5.1.1.7 自動閾値

自動閾値機能は、環境のノイズに応じて、ボタン センサーのすべての閾値パラメーターを動的に設定します。可変ノイズの環境では、この機能を有効にします。この機能はボタン センサーのみに適用できます。CY8CMBR3002 CapSense コントローラーではこの機能は常に有効にされます。

自動閾値機能を使用する際は、以下に留意してください。

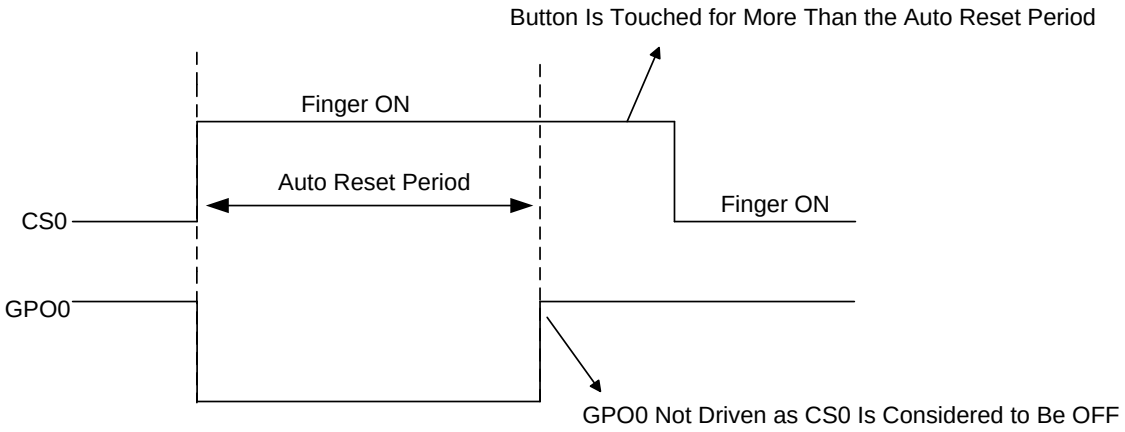
- EMC 機能が有効になった場合、自動閾値は無効になります。この場合、指閾値パラメーターを手動で設定します。
- この機能が無効になった場合、指閾値パラメーターのみを設定します。他のすべての閾値パラメーターは、指閾値パラメーターに基づいて自動的に設定されます。
- メディアン フィルタが有効になった場合、自動閾値機能を無効にし、指閾値パラメーターを手動で指定します。

5.1.1.8 センサー自動リセット時間

センサー自動リセット時間は、センサーがアクティブにされた時、センサーがオンとして見なされる最大時間を決定します。図 5-3 に示すように、センサーは自動リセット期間の後でオフになります。 この機能は、金属物体がセンサーに接近して置かれる場合にセンサーがオン状態から抜け出せないのを防止します。

自動リセット パラメーターは 2 グループに分けられています: 自動リセットと近接自動リセット。自動リセット パラメーターはボタンとスライダーに提供できます。近接自動リセットは近接センサーに適用できます。自動リセットはガード センサーに適用できません。自動リセット期間は無効になるか、または 5～20 秒にセットされます。CY8CMBR3002 CapSense コントローラーではこのパラメーターは 20 秒にセットされます。

図 5-3.センサー自動リセット



5.1.1.9 デバウンス

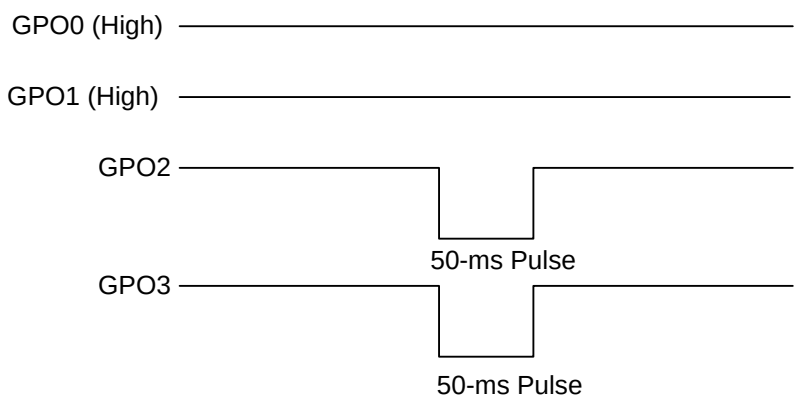
デバウンス機能は、有効なオン状態になるまでにセンサーが継続して触れられる時間を指定することで、センサーがスパイク ノイズやシステム グリッジにより誤検出されることを避けます。

デバウンス値はセンサーの機能によって異なります。例えば、電源ボタンは不用意に切り替わるのを防ぐために、長いデバウンス時間を持つ必要があります。デバウンス時間を短くすると、センサー接触や近接イベントへのデバイス応答が速くなります。デバウンス パラメーターは、スライダー センサーを除いてあらゆるセンサー タイプのに適用でき、1～15 の範囲内です。このパラメーターは、応答時間の要件に応じて設定します。詳細については、「応答時間の見積もり」節を参照してください。CY8CMBR3002 CapSense コントローラーではこのパラメーターは 3 にセットされます。

5.1.1.10 システム診断機能

システム診断は、内蔵電源投入時セルフ テスト (BIST) メカニズムです。これは有効な場合、電源投入時リセット (POR) の時またはソフトウェア リセットの後に、生産試験に有用な幾つかのテストを行います。あるセンサーにフォルトが検出されと、それが無効になり、図 5-4 のように電源投入時に 50ms パルスが対応する GPO (GPO が無効な場合) で送信されます。また、ホストが I²C インターフェースを介してテスト結果を読み取れるように、システム診断の状態はレジスタ マップでも更新されます。詳細については、CY8CMBR3xxx レジスタ TRM を参照してください。

図 5-4.BIST に CS0 と CS1 は合格し、CS2 と CS3 に問題がある例



電源投入時には、以下のテストを行います。
1. センサーのグランドへの短絡: グランドへの短絡が検出されると、センサーは無効になります。
2. センサーの VDD への短絡: VDD への短絡が検出されると、センサーは無効になります。
3. センサー間の短絡: 複数ボタンの相互短絡が検出されると、該当のボタンは無効になります。
4. 不適切な C_{MOD}: C_{MOD} の推奨値は 2.2nF±10%です。C_{MOD} の値が 1nF より小さいか 4nF より大きいと、すべてのボタンが無効になり、50ms パルスがすべての GPO で送信されます。
5. ボタンの C_P>45pF: センサーの C_P が 45pF より大きいと、センサーは無効になります。CY8CMBR3xxx で見積もった C_P の誤差は±2pF です。従って、センサーの C_P が 43～47pF になると、センサーは無効になります。
6. シールドが V_{DD}、グランド、またはセンサーに短絡されると、すべてのセンサーが無効になり、50ms パルスがすべての有効な GPO で送信されます。
7. もし、任意のスライダー セグメントが上記のテストに不合格になると、スライダー全体が無効になります。

システム診断が有効な場合、CY8CMBR3xxx コントローラーの起動時間が増加します。詳細については、CY8CMBR3xxx のデータシートの「System Specifications」節を参照してください。

注: CY8CMBR3106S は GPO をサポートしません。このため、I^2C インターフェースを介してレジスタ マップからシステム診断結果を読み取ります。

5.1.1.11 シールドおよびガード センサー

シールドおよびガード センサーは、耐水性 CapSense 設計に必要です。設計にシールド電極がある場合、EZ-Click の「Global configuration」で「Enable shield」オプションを選択します。これにより、コントローラーはシールド信号を SH ピンで駆動することができます。

設計にガード センサーがある場合、EZ-Click の「Global configuration」タブで「Enable guard sensor」オプションを選択します。これにより、コントローラーはピンをガード センサーとして設定することができます。シールドとガード センサー機能をサポートするデバイスについては、表 1-1 を参照してください。

5.1.2 GPO コンフィギュレーション

CY8CMBR3xxx コントローラーは、最大 8 個の GPO をサポートして、D/A コンバータを形成するために LED を駆動するか外部抵抗回路ネットワークを駆動します。GPO はコンフィギュレーションに応じて、PWM 信号または DC 電圧レベルを出力します。GPO ピンが CapSense センサー ピンと多重化されているため、EZ-Click で CY8CMBR3xxx ピンが CapSense ピンか GPO ピンであるかを指定します。CY8CMBR3106S CapSense コントローラーが GPO をサポートしないため、この機能はこのデバイスに適用されません。

5.1.2.1 GPO 駆動モード

CY8CMBR3xxx の GPO は 2 つの駆動モードをサポートします: オープンドレイン ローとストロング駆動。オープンドレインの LOW モードでの GPO は、デフォルトで開けられた (n/o) スイッチに相当します。センサーが非アクティブな時、対応する GPO が HI-Z 状態になります。センサーがアクティブな時、GPO は LOW にプルされます。この駆動モードは、D/A 変換を行うために外部抵抗ネットワークを駆動するのに有用です。ストロング駆動モードは LED を駆動するのに使用されます。CY8CMBR3002 CapSense コントローラーの GPO 駆動モードはオープンドレイン LOW に固定されます。

GPO 駆動モードはグローバルで、すべての有効な GPO に適用されます。

5.1.2.2 GPO 論理レベル

CY8CMBR3xxx では、GPO ピンの論理レベル (アクティブ HIGH またはアクティブ LOW) を設定することができます。表 5-1 に示されたに、この機能により、センサーの状態に応じて GPO の出力電圧を柔軟に制御することができます。この機能はグローバルで、すべての有効な GPO ピンに適用されます。CY8CMBR3002 CapSense コントローラーの GPO 論理レベルは「アクティブ LOW」です。駆動モードをオープンドレインの LOW に設定すると、表 5-1 は、ピンが外部で V_{DD} にプルされている時にのみ適用できます。

表 5-1. GPO 論理レベル

GPO 論理レベル	センサー状態	GPO での出力電圧
アクティブ HIGH	オン	V_{DD}
	オフ	GND
アクティブ LOW	オン	GND
	オフ	V_{DD}

5.1.2.3 GPO での PWM 出力

CY8CMBR3xxx は GPO ピンでの PWM 出力を提供します。PWM 信号は、LED の輝度を制御するのに有用です。GPO で PWM 出力を有効にするために、EZ-Click の「Global configuration」タブでアクティブ デューティ サイクルおよび非アクティブ デューティ サイクルのパーセンテージを指定します。100%のアクティブ デューティ サイクルと 0%非アクティブ デューティ サイクルが GPO での DC 出力となります。CY8CMBR3106S と CY8CMBR3002 CapSense コントローラーにはこの機能がありません。

PWM 出力機能はグローバルで、すべての有効な GPO ピンに適用されます。PWM 信号の周波数は 106.6Hz です。PWM デューティ サイクルは、6.67%ステップ サイズでそれぞれの GPO に対して設定することができます。

5.1.2.4 GPO ホスト制御

CY8CMBR3xxx では、各 GPO (GPO0～GPO7) は CapSense ピン (CS0～CS7) に論理的にリンクされています。センサーがアクティブになると。対応する GPO が自動的にオンになります。但し、ホスト プロセッサが GPO ピンを制御する際は、GPO ホスト制御機能を有効にする必要があります。この機能により、ホスト プロセッサは GPO_OUTPUT_STATE レジスタに書き込むことで GPO ピンを制御することができます。詳細については、CY8CMBR3xxx レジスタ TRM を参照してください。CY8CMBR3106S と CY8CMBR3002 CapSense コントローラーにはこの機能がありません。

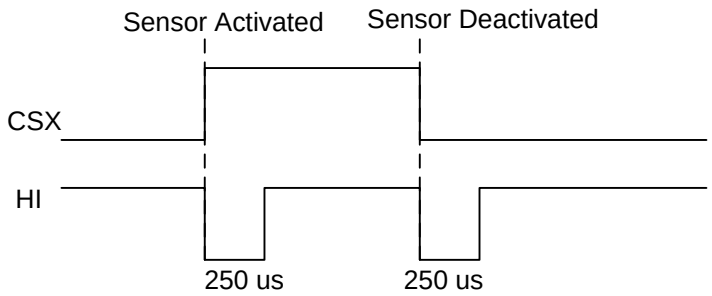
ホストがコマンドを書き込んでから CY8CMBR3xxx コントローラーがこれらのコマンドに対応するまでの遅延時間は、コントローラーが Look-for-Touch モードと Look-for-Proximity モードで動作している時の 1 つのスキャン期間に等しいです。コントローラがアクティブ モードで動作する時、この遅延時間はすべてのセンサーの総スキャン時間に依存します。

注 EZ-Click では、GPO が無効にされず、該当する GPOx/CSx ピンが CapSense センサーとしてコンフィギュレーションされない場合でも、GPO は該当するセンサーのオン／オフ ステータスに基づいて駆動されます (オン／オフにされます)。GPO ピンを無効にされるためには、EZ-Click でピンを CapSense 入力としてコンフィギュレーションし、ピンをグランドに接続する必要があります。

5.1.2.5 ホスト割り込み

ホスト割り込み機能は、図 5-5 で示すように、センサー状態またはスライダー位置が変化するたびに、ピンで 250µs のアクティブ LOW パルス信号をホスト プロセッサに送信します。この機能を有効にするために、EZ-Click でホスト割り込みピンを指定します。ホスト割り込みピンにはオープンドレイン LOW 駆動モードがあり、正常な動作のために外部プルアップ抵抗を必要とします。この機能をサポートするデバイスの一覧は表 1-1 をご覧ください。

図 5-5. CSx ボタンが接触される時のホスト割り込みライン



5.1.2.6 トグル

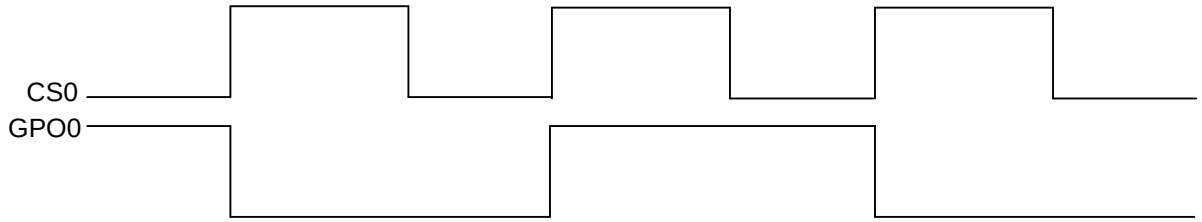
トグル機能がイネーブルされる時、GPO ピンの状態は、ボタンの立ち上がりエッジの度に近接センサーの状態変化に応じて変わります。この機能は、CapSense コントローラーでメカニカル ボタンの切り替え動作を実施します。トグルのコンフィギュレーションを

図 5-6 に示します。トグル機能は、各 GPO ピンで有効／無効にします。CY8CMBR3002 CapSense コントローラーにはトグル機能がありません。

トグル機能を使用する時に以下のことを考慮してください。

- トグルと LED ON 時間機能は相互排他的な関係を持っています。両方の機能が有効にされたら、トグル機能はもっと高い優先順位を付けられるため、LED ON 時間の機能は無効にされます。
- トグル機能が有効にされ、いずれかの GPO がオンにされると、デバイスは Look-for-Proximity モードに移行しません。ユーザーのデザインはウェーク オン アプローチ (近づいた時にオンになる) 機能を必要とする場合、トグル機能を無効にします。

図 5-6.トグル機能動作例

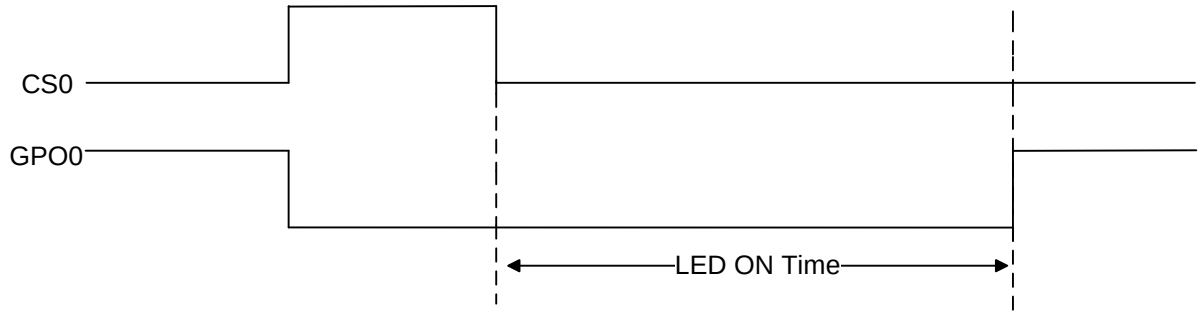


5.1.2.7 LED ON 時間

LED ON 時間は、図 5-7 に示すように、対応センサーが開放された後に GPO ピンが LOW か High に駆動する期間を示します。トグル機能は、各 GPO ピンで有効／無効にします。LED ON 時間はすべての GPO ピンに共有され、その範囲が 0～2000ms で、分解能が 20ms です。CY8CMBR3002 CapSense コントローラーには LED ON 時間機能がありません。

トグルと LED ON 時間機能は相互排他的な関係を持っています。両方の機能が有効にされたら、トグル機能はもっと高い優先順位を付けられるため、LED ON 時間の機能は無効にされます。

図 5-7.LED ON 時間



5.1.2.8 アナログ電圧出力

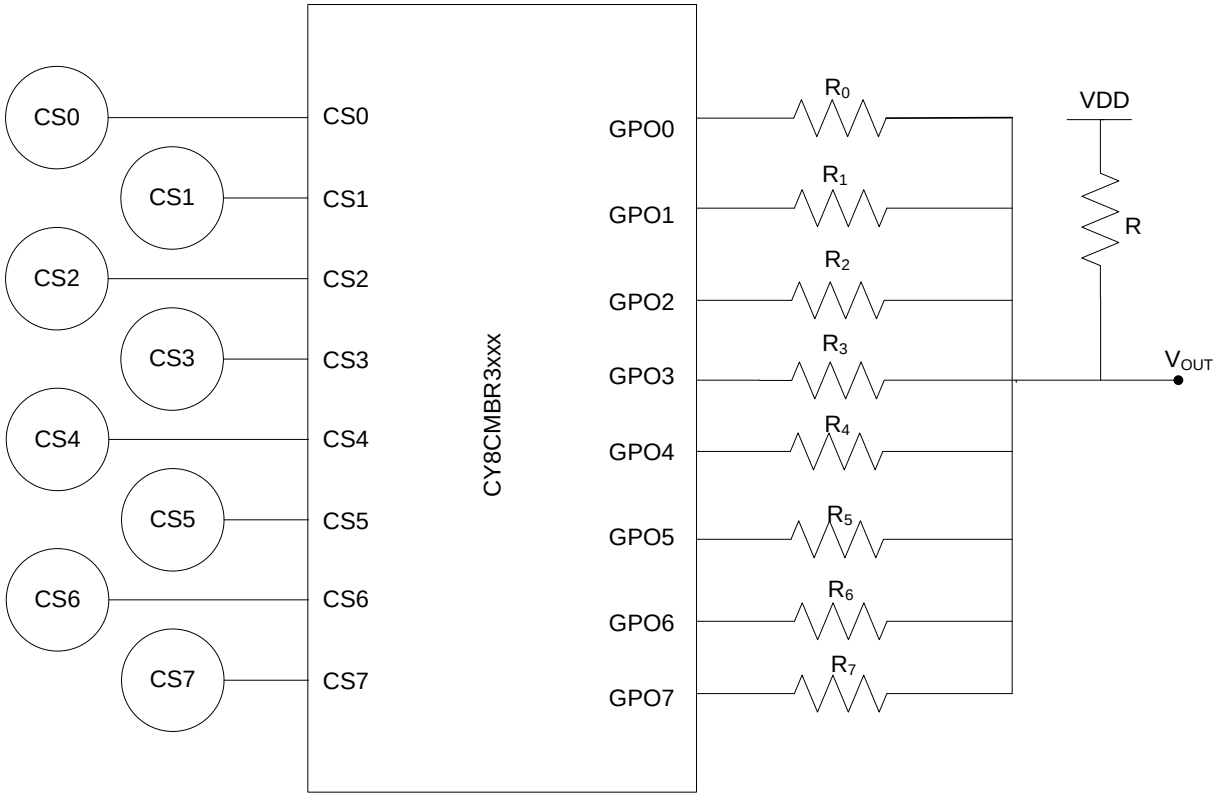
図 5-8 に示すように、アナログ電圧出力機能は抵抗を使ってセンサー状態をホストへのアナログ電圧として示します。この設定では、GPO 駆動モードをオープン ドレインの LOW 駆動モードに設定する必要があります。センサーが接触されると、対応 GPO は論理 LOW 信号に駆動され、電圧分周器は V_{OUT} で設定されます。センサーが開放されると、対応 GPO は HI-Z 状態に移行し、電圧 V_{OUT} は V_{DD} です。特定の時点で 1 つのボタンのみがオンにされるために、FSS 機能を有効にすることができます。

アナログ電圧 V_{OUT} は、以下の式で計算されます。

$$Vout = \frac{VDD \times Rn}{R + Rn}$$ 式 3

ここで、R が VDD と VOUT 間の抵抗
Rn が V_{OUT} とグランド間の実効抵抗

図 5-8. GPO と抵抗ネットワーウを使ったアナログ電圧出力



5.1.3 ブザーのコンフィギュレーション

5.1.3.1 ブザー出力

CY8CMBR3xxx は単一入力の Piezo ブザーをサポートします。ブザーは PWM 信号により駆動され、その周波数が設定可能です。ブザーをイネーブルするには、EZ-Click でブザー ピンを選択します。CY8CMBR3102 と CY8CMBR3002 CapSense コントローラーにはこの機能がありません。

様々な Piezo-ブザ駆動要求に対応する、且ついろいろな異なったトーンを提供するために、CY8CMBR3xxx は、1.0、1.1、1.3、1.6、2.0、2.6、および 4.0kHz のブザー周波数をサポートします。ブザー出力のデューティ サイクルは、50%に固定されます。

5.1.3.2 ブザーON 時間

図 5-9 に示すように、ブザーON 時間により、センサーがアクティブになった時にブザー出力もアクティブになる期間を指定することができます。ブザー信号出力は設定した時間の間駆動され、センサー接触時間に依存しません。出力は、センサーがアクティブであっても、ブザーON 時間が経過した後にアイドル状態になります。

図 5-10 および図 5-11 に示すように、ブザーON 時間が経過する前に同じセンサーまたは他のセンサーが接触される場合、ブザー信号出力機能は再開しません。ブザーON 時間の範囲は 100ms～12.7s で、その分解能が 100ms です。

図 5-9. 接触イベントによるブザー起動

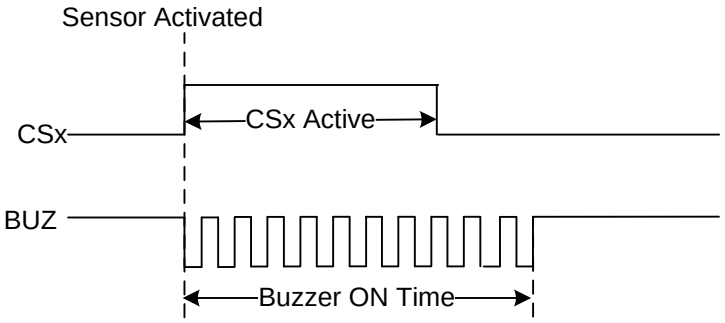


図 5-10.同センサーで連続接触時のブザー動作

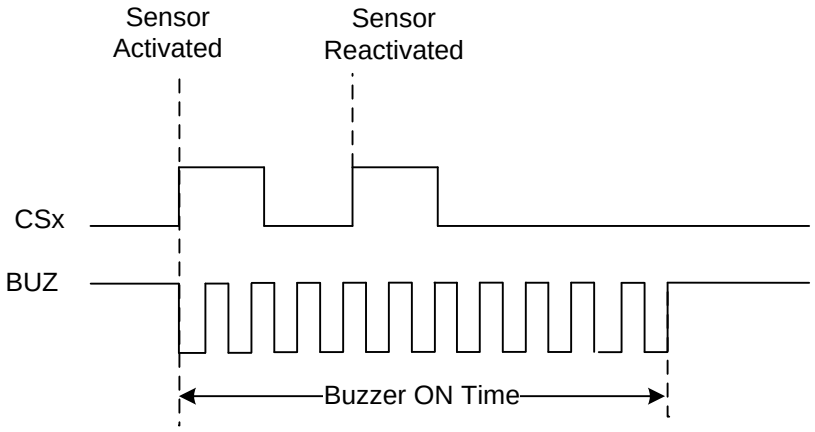
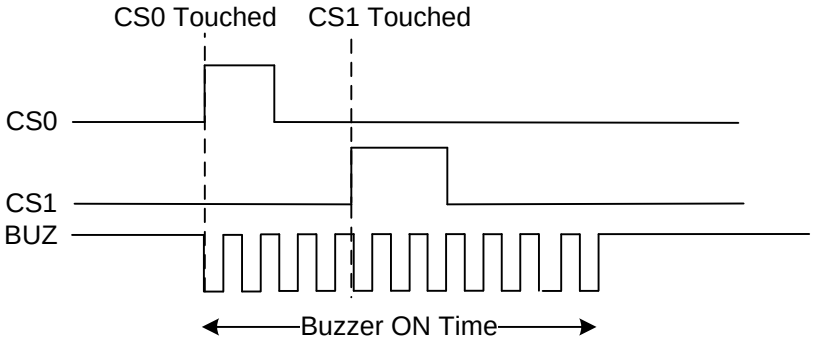


図 5-11.複数センサーがアクティブな時のブザー動作



5.1.4 デバイスのコンフィギュレーション

5.1.4.1 電源電圧

より適切な動作のために、CY8CMBR3xxx コントローラーの電源電圧を EZ-Click で指定する必要があります。動作電圧範囲が 1.71～1.89V の場合、1.8(±5%) V オプションを選択します。そうしない場合、1.8～5.5 V のオプションを選択します。CY8CMBR3002 CapSense コントローラーにはこの機能がありません。

5.1.4.2 I²C アドレス

CY8CMBR3xxx コントローラーは、I²C インターフェースを介してホスト プロセッサに連絡します。通常の I²C 通信では、I²C バスに 2 つ以上のスレーブ デバイスがあります。スレーブ デバイス間の I²C アドレスの衝突を回避するために、CY8CMBR3xxx は、コントローラーの I²C アドレスを指定するオプションを提供します。I²C アドレスは、7 ビットの値で、8 (0x08) ～119 (0x77) の範囲内にあります。CY8CMBR3002 コントローラーは I²C 通信をサポートしません。

5.1.4.3 スキャン期間

スキャン期間パラメーターは、Look-for-Touch モードおよび Look-for-Proximity モードでの逐次のセンサー スキャン間の時間を示します。パラメーターの範囲は 20～500ms で、分解能が 20ms です。CY8CMBR3002 CapSense コントローラーではこのパラメーターは 120ms に固定されます。

コントローラーの応答時間と消費電力は、スキャン期間に依存します。詳細は、低消費電力設計上の考慮事項を参照してください。

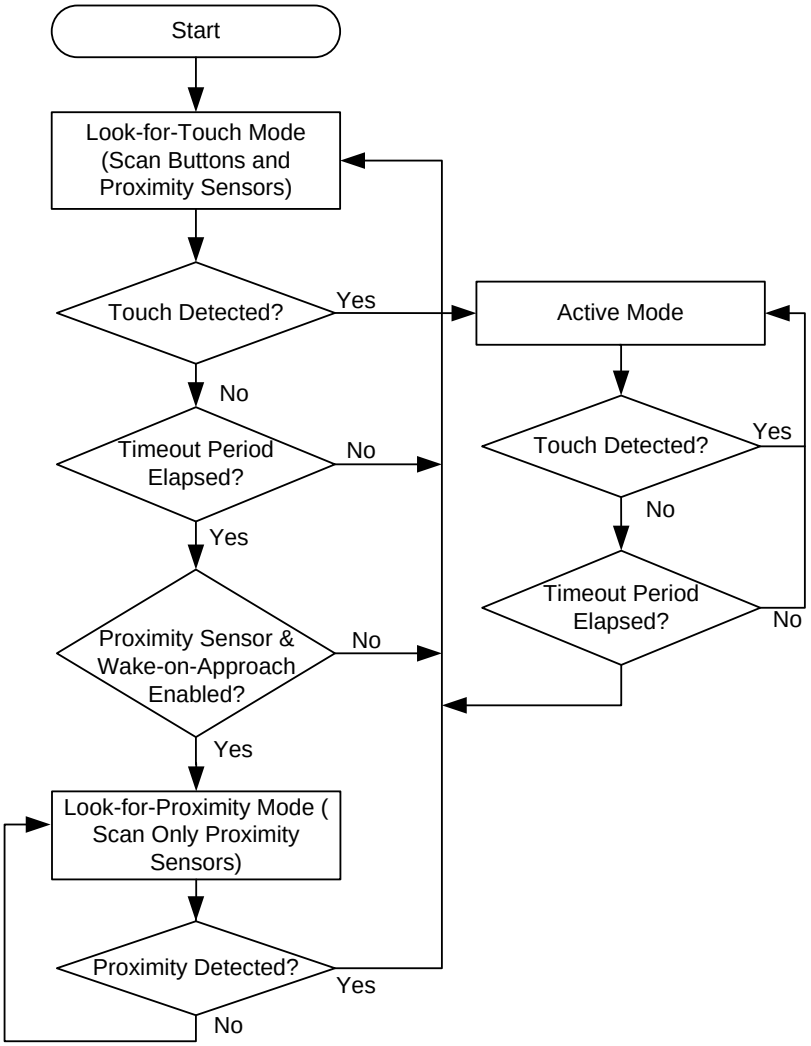
5.1.4.4 状態タイムアウト

状態タイムアウトの間隔は、センサーが接触されない時に、デバイスが現在の動作モードから他の動作モードに移行するのにかかる時間を指定します。このパラメーターは、連続接触時にセンサーの応答時間に影響を及ぼします。設計要件に応じてこのパラメーターをセットする必要があります。これにより、コントローラーがアクティブ モードに移行でき、連続接触に対して速く応答できます。パラメーターの範囲は 0～63s で、分解能が 1s です。CY8CMBR3002 CapSense コントローラーではこのパラメーターは 10s に固定されます。

5.1.4.5 ウェイク オン アプローチ (近づいた時にオンになる)

消費電力を減少するために、CapSense デザインにウェイク オン アプローチ機能を適用します。この機能により、近接センサーがイネーブルされる時に、デバイスは、Look-for-Proximity モードに移行できます。look-for-proximity モードでは、近接センサーのみがスキャンされます。近接が検知されたら、コントローラーは、図 5-12 に示すように、センサーがスキャンされる Look-for-Touch モードに移行します。詳細は、「低消費電力設計上の考慮事項」を参照してください。CY8CMBR3002 CapSense コントローラーにはこの機能がありません。

図 5-12. Look-for-Touch モードから Look-for-Proximity モードへの遷移



5.2 CY8CMBR3xxx の設定

CY8CMBR3xxxファミリは、レジスタ設定が可能なCapSenseコントローラーです。このコントローラーはアドレス0x00から0x7Eまでの128バイトのコンフィギュレーション レジスタを持ちます。CY8CMBR3002 CapSenseコントローラーを除き、CY8CMBR3xxxファミリのすべてのコントローラーはI²Cインターフェースを介してコンフィギュレーションすることができます。コンフィギュレーションデータは、そのコントローラーがパワーオフ状態であっても、コンフィギュレーション データを保持するように不揮発性メモリに保存することができます。

CY8CMBR3xxxレジスタの詳細については、CY8CMBR3xxxレジスタTRMを参照してください。

CY8CMBR3xxxコンフィギュレーション レジスタは、次のいずれかの方法で、I²Cインタフェースを介してコンフィギュレーションされます。

1. EZ-Click 2.0
2. サードパーティ プログラマ
3. ホスト API
4. ブリッジ コントロール パネル

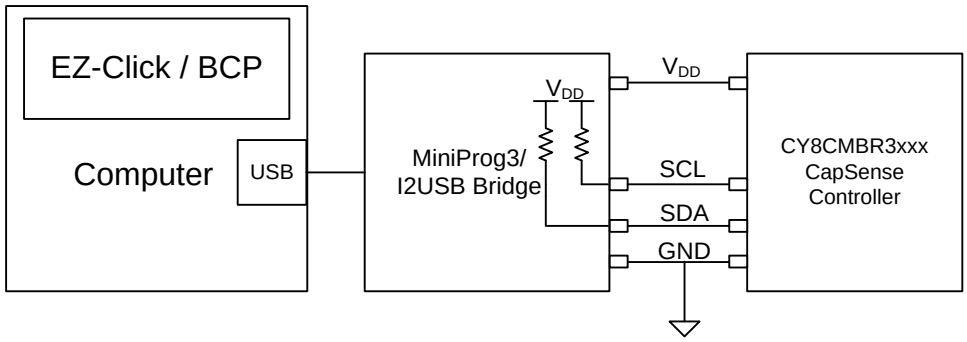
5.2.1 EZ-Click を使った CY8CMBR3xxx 設定

EZ-Click は、簡単かつ直観的なグラフィカル ユーザー インターフェース (GUI) ツールであり、CY8CMBR3xxx CapSense コントローラーの設定に使用されます。すべての必要なパラメーターで、I²C インタフェースを介してデバイスを設定します。

5.2.1.1 ハードウェア接続

図 5-13 に EZ-Click 2.0 またはブリッジ制御パネルを使用して I²C インターフェースを介して CY8CMBR3xxx CapSense コントローラーをコンフィギュレーションするために必要なハードウェアセットアップを示します。EZ-Click／BCP は MiniProg3 または CY3240-I2USB ブリッジのいずれかを使用して CY8CMBR3xxx に通信します。MiniProg3 または I2USB ブリッジと CY8CMBR3xxx コントローラー間の接続は図 5-13 に示されます。MiniProg3 または I2USB ブリッジの SCL と SDA ピンは内部で V_{DD} にプルアップされる (プルアップ抵抗値は MiniProg3／I2USB ブリッジ キット ユーザー ガイドを参照) ため、これらのラインで接続される外部プルアップ抵抗がないようにしてください。MiniProg3 または CY3240-I2USB ブリッジから V_{DD} を CY8CMBR3xxx コントローラーに供給することは必須ではありません。CY8CMBR3xxx コントローラーは外部から電源を供給することができます。これらの場合では、EZ-Click と BCP において、電源がオンにされ、MiniProg3/I2USB ブリッジの電圧設定は CY8CMBR3xxx V_{DD} の設定と同様に選択されるようにしてください。

図 5-13. EZ-Click または BCP で CY8CMBR3xxx をコンフィギュレーションするのに必要なハードウェア セットアップ



注: CY3280-MBR3 キットでは、PSoC 5LP デバイスが I2USB ブリッジとして使用するため、CY8CMBR3116 CapSense コントローラーと通信するのに MiniProg3 または CY3280-I2USB ブリッジが不要です。詳細は、CY3280-MBR3 キット ユーザー ガイドを参照してください。

5.2.1.2 *EZ-Click*

この節では、EZ-Click に簡単な紹介をしています。CY8CMBR3xxx コントローラーを設定する方法の詳細については、EZ-Click ユーザーガイドを参照してください。

EZ-Click は以下の機能を提供します。

- CY8CMBR3xxx コントローラーをコンフィギュレーションするためのグラフィカル ユーザー インターフェース (GUI)
- センサーパラメーターを調整するのとリアルタイムでセンサーのデバッグ データを表示できるようにします。
- 製造ラインの試験をサポートし、システム診断と CapSense センサーSNR を表示します。
- コンフィギュレーションを保存し、複数のサンプルでそれを使用できるようにします。
- 図 5-16 に示すように、ホストの API を使用して CY8CMBR3xxx コントローラーを設定するために 128 バイトのコンフィギュレーション データのアレイでコンフィギュレーションされたヘッダ ファイルを生成します。詳細は 5.2.3 節を参照してください。
- サードパーティのプログラミング ツールを使用してデバイスを設定するには、設定の HEX ファイルを生成します。詳細な説明は 5.2.2 節を参照してください。
- 図 5-23 に示すように、ブリッジ コントロール パネルを使ってデバイスを設定するための I²C 命令を含む設定ファイルを生成します。詳細は 5.2.4 を参照してください。

図 5-15 に示すように、EZ-Click は、5 つのタブがあります。

- 「Start Page」タブ: プロジェクトの新規作成、コンフィギュレーション ファイルの生成、および MBR ファミリのコントローラーを設定するステップについて説明します。

- CapSense sensor configuration: このタブで、センサー数を選択し、すべてのセンサーのパラメーターと機能を設定することが可能です。
- Global configuration: このタブで、GPO を有効にし、グローバル パラメーターを指定することができます。
- CapSense output: このタブで、センサーのデバッグ データは、デバッグ目的のために監視することができます。
- System Diagnostics: このタブで、システム診断結果をセンサー毎に表示されます。

注: 「CapSense sensor configuration」、「Global configuration」、「CapSense output」、および「System diagnostics」のタブは、新しいプロジェクトが作成された場合にのみアクセスすることができるか、既存のプロジェクトが開かれます。

図 5-14. EZ-Click で生成された各種フィルタを使う CY8CMBR3xxx 設定

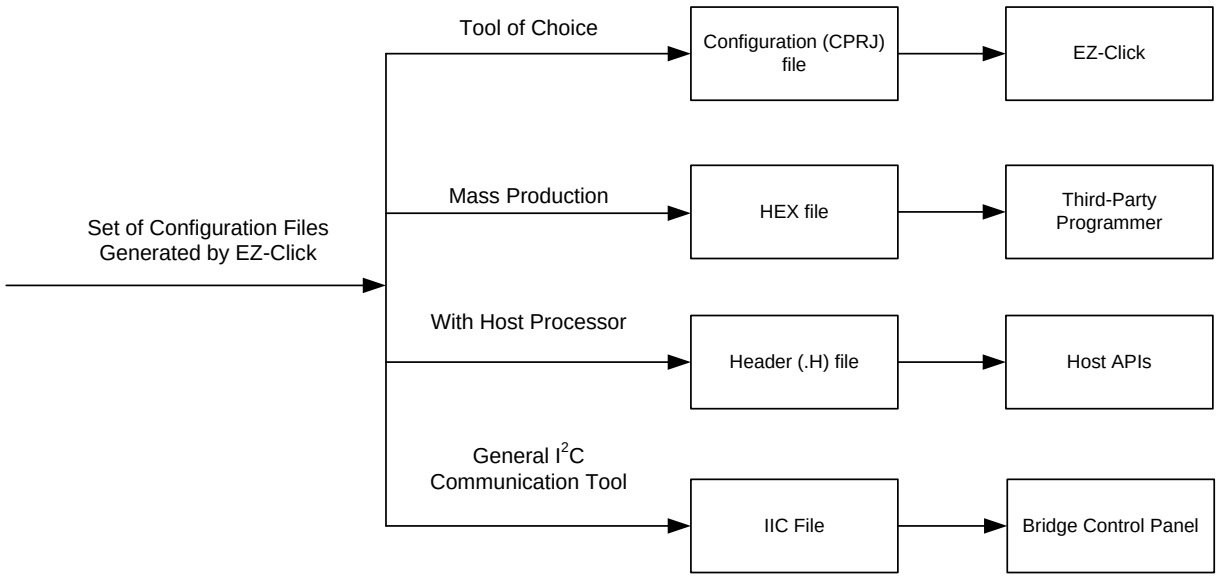


図 5-15. EZ-Click のカスタマイザ ツール

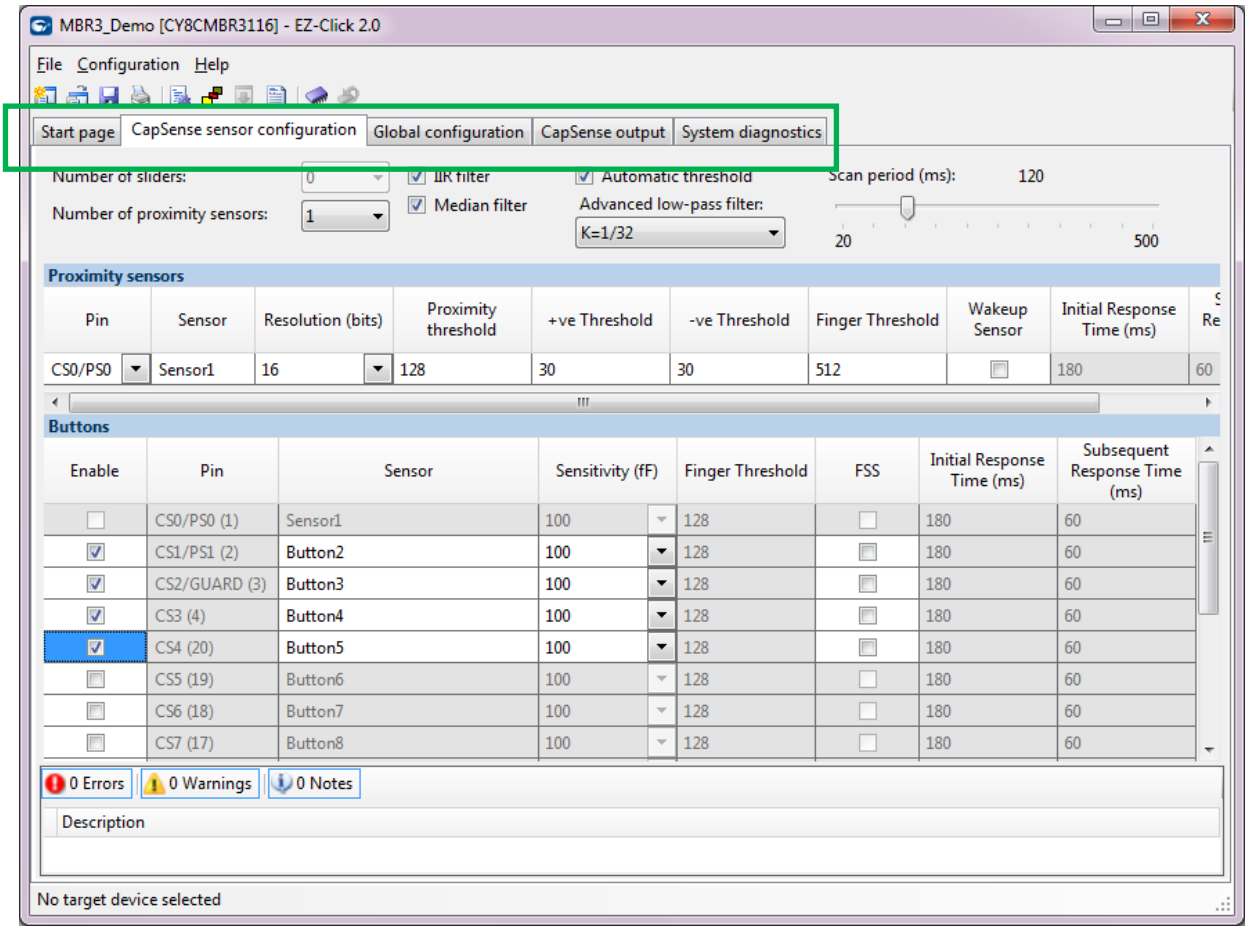


図 5-16. EZ-Click カスタマイザ ツールが生成した C ヘッダー ファイルの例

MBR3_Demo.h

```
/* Project: C:\MBR3_Demo\MBR3_Demo.cprj
 * Generated: 02/18/2014 11:20:08 AM +05:30 */
const unsigned char CY8CMBR3116_configuration[128] = {
    0x78u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u,
    0x00u, 0x00u, 0x00u, 0x00u, 0x7Fu, 0x7Fu, 0x7Fu, 0x80u,
    0x80u, 0x80u, 0x80u, 0x7Fu, 0x7Fu, 0x7Fu, 0x7Fu, 0x7Fu,
    0x7Fu, 0x7Fu, 0x7Fu, 0x7Fu, 0x03u, 0x00u, 0x00u, 0x00u,
    0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x80u,
    0x05u, 0x00u, 0x00u, 0x02u, 0x00u, 0x02u, 0x00u, 0x00u,
    0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u,
    0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x01u, 0x01u,
    0x00u, 0xFFu, 0xFFu, 0xFFu, 0xFFu, 0xFFu, 0xFFu, 0xFFu,
    0xFFu, 0x00u, 0x00u, 0x00u, 0x00u, 0x03u, 0x01u, 0x58u,
    0x00u, 0x37u, 0x06u, 0x00u, 0x00u, 0x0Au, 0x00u, 0x00u,
    0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u,
    0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u,
    0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u,
    0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u,
    0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x00u, 0x41u, 0x41u
};
```

5.2.2 サードパーティ プログラマを使用した CY8CMBR3xxx のコンフィギュレーション

大量生産中に速やかに数多くのデバイスを設定するために、サイプレスは、RPM Systems Corporation のサードパーティ プログラマの使用を推奨します。サードパーティのプログラマを使って CY8CMBR3xxx コントローラーを設定するには、EZ-Click が生成する設定の hex ファイルを使用します。詳細は、RPM Systems Corporation にご連絡ください。

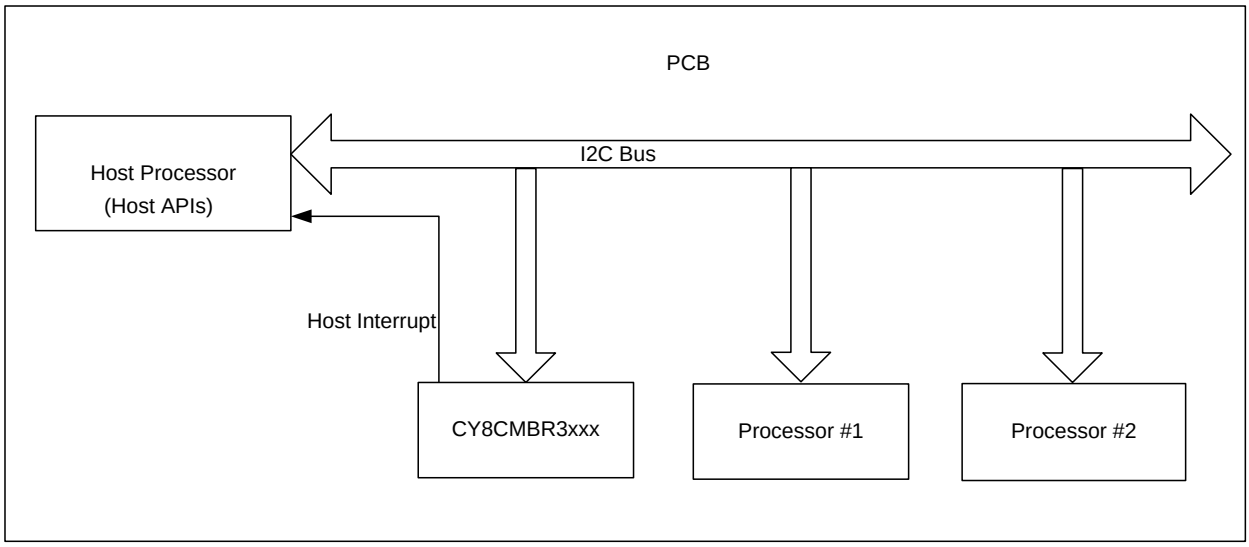
CY8CMBR3xxxファミリのプログラミング仕様の情報は、CY8CMBR3xxx Device Programming Specificationsを参照してください。

5.2.3 ホスト API を使用した CY8CMBR3xxx のコンフィギュレーション

図 5-17 に、ホスト プロセッサーが同じプリント基板で I²C を介して他の幾つかのプロセッサと共に CY8CMBR3xxx コントローラーと通信するシステムを示します。ホストは、必要に応じて調整し、実行時に CY8CMBR3xxx コントローラーからのセンサーのステータスをリードバックし、その動作を別の要件に合うように再設定できます。

基板搭載のホスト プロセッサを使用して CY8CMBR3xxx コントローラーをコンフィギュレーションするために、サイプレスは、ホストプロセッサのファームウェア内に統合できる「Host APIs」(ホスト API) と呼ばれるライブラリー ファイル一式を提供します。この節ではホスト API の説明を行い、CY8CMBR3xxx コントローラーをコンフィギュレーションするためにそれらの使い方を説明します。

図 5-17.ホスト API の使用を実演するシステム



5.2.3.1 ホスト API の概要

ホスト API は、サイプレスが提供するホスト プロセッサを有効にする一組の C スタイルの関数であり、直接、I²C インタフェースを介して、CY8CMBR3xxx コントローラーと対話します。

これらの API は以下のことに使用できます。

1. CY8CMBR3xxx コントローラーを設定
2. I²C のコマンドを CY8CMBR3xxx コントローラーに発行
3. CY8CMBR3xxx のレジスタ マップから情報を読み出す

5.2.3.2 ホスト API の説明と使用

ホスト API は、高レベル API と低レベル API の 2 つの節に分かれます。

5.2.3.2.1 高レベル API

高レベル API は C スタイルのソフトウェアで純粋に必要な機能を実装していて、CY8CMBR3xxx コントローラーとの物理的な相互作用に応じて次々に低レベルの API に依存しています。表 5-2 に高レベル API 一覧とそれらの使用を記載します。API プロトタイプ、パラメーター、および戻り値は、付録に記載されています。

表 5-2. 高レベルのホスト API

#	API	説明と使用
1	CY8CMBR3xxx_WriteData()	デバイスにデータを書き込む汎用 API この API を使用して、CY8CMBR3xxx コントローラーの 1 つまたは複数のレジスタに書き込む
2	CY8CMBR3xxx_ReadData()	CY8CMBR3xxx コントローラーからデータを読み出す汎用 API この API を使用して、CY8CMBR3xxx コントローラーの 1 つまたは複数のレジスタに読み出す
3	CY8CMBR3xxx_WriteDualByte()	CY8CMBR3xxx コントローラーの 2 バイトのレジスタに書き込む API 例えば、16 ビット センサーの有効レジスタはこの API を使って書き込める
4	CY8CMBR3xxx_ReadDualByte()	CY8CMBR3xxx コントローラーの 2 バイトのレジスタから読み出す API 例えば、16 ビット センサーの状態レジスタはこの API を使って読み出せる
5	CY8CMBR3xxx_SendCommand()	不揮発性メモリにコンフィギュレーションを保存したり、CY8CMBR3xxx コントローラーをリセットしたりするように、CY8CMBR3xxx コントローラーにコマンドを発行する API
6	CY8CMBR3xxx_CheckCommandStatus()	最後のコマンドの状態をチェックする API は、CY8CMBR3xxx コントローラーに送る。CY8CMBR3xxx コントローラーに新しいコマンドを送信する前に、ホスト プロセッサ内のこの API を呼び出すことを推奨
7	CY8CMBR3xxx_Configure()	CY8CMBR3xxx コントローラーにコンフィギュレーション レジスタの全 128 バイトを設定する API この API を使用するには、EZ-Click 内のコンフィギュレーションを作成し、ホスト ファームウェアに生成されたヘッダファイルを使用する
8	CY8CMBR3xxx_CalculateCrc()	指定されたコンフィギュレーションの CRC チェックサムを計算する API
9	CY8CMBR3xxx_VerifyDeviceOnBus()	I²C バス上 CY8CMBR3xxx デバイスが意図したものであることを確認する API[10]。右の I²C スレーブ デバイスがバス上にあることを確認するために、ホストプロセッサからこの API を呼び出す
10	CY8CMBR3xxx_SetDebugDataSensorId()	この API は、センサーのデバッグデータが CY8CMBR3xxx コントローラーから読み出されるために備えているセンサー番号を設定する
11	CY8CMBR3xxx_ReadSensorDebugData()	センサーのデバックデータを読み出す API この API はセンサーの性能を知るのに役立つ
12	CY8CMBR3xxx_ReadDiffCounts()	CY8CMBR3xxx コントローラーのすべての異なるカウントを読み出す API
13	CY8CMBR3xxx_ReadSensorStatus()	CY8CMBR3xxx コントローラーのすべての状態を読み出す API CY8CMBR3xxx コントローラーはホスト割り込みパルスを発行すると、トリガーされたセンサーを知るために、この API を呼び出す

5.2.3.2.2 低レベル API

低レベル API は、ホスト プロセッサと I²C インタフェース経由の CY8CMBR3xxx デバイスとの間の物理的な通信を担当していて、ハードウェアに依存しています。これらの API は、デフォルトで PSoC4 のアーキテクチャをサポートし、特定のホスト プロセッサをサポートするように変更する必要があります。

[10] ホスト API は、同じ I²C バス上に存在する CY8CMBR3xxx ファミリーの異なるコントローラーをコンフィギュレーションするために使用できません。

CY8CMBR3xxxコントローラーがディープスリープ モードにあるとき、最初のI²Cトランザクションは否定応答 (NACK応答) されます。CY8CMBR3xxxコントローラーはI²CのトランザクションをNACKする場合、LOWレベルAPIは成功なトランザクションのために次の340ms以内に再実行する必要があります。CY8CMBR3xxxデータシートの「ホスト通信プロトコル」を参照してください。これを実現するためには、I²Cを除くホスト プロセッサのすべての割り込みを無効にし、または実行待ちの割り込みが340ms以内に完了するようにしてください。2つ後続のI²Cトランザクション間の340ミリ秒以上の遅延がある場合、デバイスはディープスリープ モードに入る可能性があり、ホストがNACKを受信する可能性があります。

低レベルAPIが3つあります: Host_LOWLevelWrite()、Host_LOWLevelRead、とHost_LOWLevelDelay()。APIプロトタイプ、パラメータと戻り値は、付録に記載されています。

表 5-3. 低レベルのホスト API

#	API	説明と使用
1	Host_LOWLevelWrite()	I²C インターフェース経由の CY8CMBR3xxx コントローラーのレジスタ マップに書き込む API
2	Host_LOWLevelRead()	I²C インターフェース経由の CY8CMBR3xxx コントローラーのレジスタ マップからデータを読み出す API
3	Host_LOWLevelDelay()	この API は、高レベルの API によって使用される時間遅延機能を実現します。遅延周期はミリ秒単位。この遅延は、必要量の時間に応じてコードが実行されるブロックで成される

5.2.3.3 ホスト API ファイルの構造

ホスト API のライブラリのファイル一覧は表 5-4 に記載します。

表 5-4. ホスト API 用のファイル構造

#	ファイル名	説明
1	CY8CMBR3xxx_Device.h	このファイルには、CY8CMBR3xxx ファミリの各種デバイスを定義するためにマクロが含まれています。また、どのデバイスが現在のホストの API に使用されているかを定義します。
2	CY8CMBR3xxx_Registers.h	このファイルには、CY8CMBR3xxx ファミリーのすべてのレジスタのオフセット アドレスを定義するマクロが含まれています。直接そのアドレスの代わりに、レジスタを参照するこれらのマクロを使用します。各レジスタのマクロは、レジスタは現在使用されているデバイスが利用可能である場合にのみアクセス可能です。
3	CY8CMBR3xxx_CommandsAndConfig.h	このファイルには、コマンド レジスタ (CTRL_CMD) の各種オペコードを定義するためにマクロが含まれています。
4	CY8CMBR3xxx_APIs.h	このファイルには、すべての高レベル API の宣言が含まれています。このファイルでは、デバイスから読み取らセンサー デバッグ データとセンサーの状態を保持するための構造定義も宣言しています。
5	CY8CMBR3xxx_CRC.c	このファイルには、CY8CMBR3xxx デバイスの特定のコンフィギュレーション用に CRC を計算するための API が含まれています。
6	CY8CMBR3xxx_APIs.c	このファイルには、CRC を計算するための API を除いて、すべての高レベル API の定義が含まれています。
7	CY8CMBR3xxx_HostFunctions.h	このファイルには、ホスト依存、低レベルの API の宣言が含まれています。
8	CY8CMBR3xxx_HostFunctions.c	このファイルには、低レベルの API の宣言が含まれています。ホストプロセッサの I²C の実装に合わせて、これらの API の内容を変更します。

5.2.3.4 マクロ

「CY8CMBR3xxx_Device.h」とそれぞれ「CY8CMBR3xxx_APIs.h」内で定義された 2 つのマクロ「CY8CMBR3XXX_DEVICE」と「CY8CMBR3xxx_SYNC_COUNTER_MATCH_RETRY」があります。これら二つのマクロは CY8CMBR3xxx コントローラーおよび同期カウンタ不一致の場合に I²C の再試行回数を定義するために修正される必要があります。

1. CY8CMBR3xxx_DEVICE
   このマクロはホストでアクセスしたコントローラーを定義します。このコントローラーは、CY8CMBR3002 を除く CY8CMBR3xxx ファミリー内の異なるコントローラーのいずれかになります (このコントローラーは I²C 通信をサポートしていないので) 。CY8CMBR3xxx ファミリーの異なるコントローラーは、CY8CMBR3xxx_Device.h 内で定義されています。

2. CY8CMBR3xxx_SYNC_COUNTER_MATCH_RETRY
   このマクロは、同期カウンタが読み出し動作中に一致しない場合に、CY8CMBR3xxx コントローラーとで I²C 通信を再試行する API の CY8CMBR3xxx_ReadSensorDebugData () と CY8CMBR3xxx_ReadDiffCounts () の回数を定義します。通常、読み出し動作は最初の試みで失敗しても、2 回目の試みで成功するでしょう。10 以上の値を設定することを推奨します。同期カウンタの詳細は、CY8CMBR3xxx Registers TRM を参照してください。

5.2.3.5 デモ プロジェクトと使用法

ホスト API の使用に直ぐに慣れ会得するために、デモ プロジェクトをホスト プロセッサとして PSoC4 で作成します。このデモでは、PSoC4 パイオニア キットを使用して CY3280-MBR3 評価キット上の CY8CMBR3116 コントローラーを設定するホスト API を使用する方法を紹介しています。

このデモに必要なハードウェアー覧は以下の通りです。

- CY3280-MBR3 評価キット
- CY8CKIT-042 PSoC4 のパイオニア キット
- USB Aーミニ B ケーブル
- パソコン／ノート パソコン

このデモに必要なソフトウェアー覧は以下の通りです。

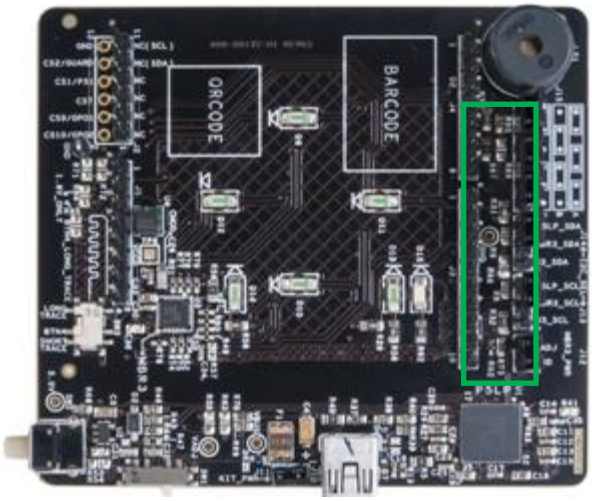
- PSoC Programmer 3.20.0

注: PSoC4 のパイオニア キット用にキットのドライバが既にインストールされている必要があります。キットドライバをインストールする詳細は、CY8CKIT-042 PSoC 4 のパイオニア キット ガイド を参照してください。

PSoC4 のパイオニア キットを使用して MBR3 評価キットを設定するには、次の手順に従ってください。

1. MBR3 評価キット上のヘッダ J13 のジャンパの位置を変更して、J13-2 ピンと J13-3 ピンを接続します。
2. ヘッダ J14 のジャンパの位置を変更して、J14-2 ピンと J14-3 ピンを接続します。
3. 位置 A への MBR3 評価キット上のヘッダ J15 のジャンパー位置を変更します。これで、ブザーとホスト割り込み出力をサポートするキットを有効にします。

図 5-18. MBR3 評価キット上のジャンパ J13、J14、J15 の位置



4. PSoC4 のパイオニア キット上に MBR3 キットを取り付け、両キットの USB ポートを互にあるべき位置に調整します。USB ポートを調整する時に、PSoC 4 のパイオニア キットの J1、J2、J3、J4 ヘッダは CY3280-MBR3 EVK の J1、J2、J3、J4 ヘッダと接続します。
5. MBR3 評価キットの一部として利用可能な、ミニ B ケーブル、USB A を介して PC に PSoC4 パイオニア キットを接続します。MBR3 キットのグローの電源 LED (赤色) 最終的な接続を図 5-19 に示します。ハードウェアのセットアップが完了すると、プロジェクトが PSoC4 ホストにダウンロード可能になります。

図 5-19. PSoC4 のパイオニア キットの最上部に搭載した MBR3 キット



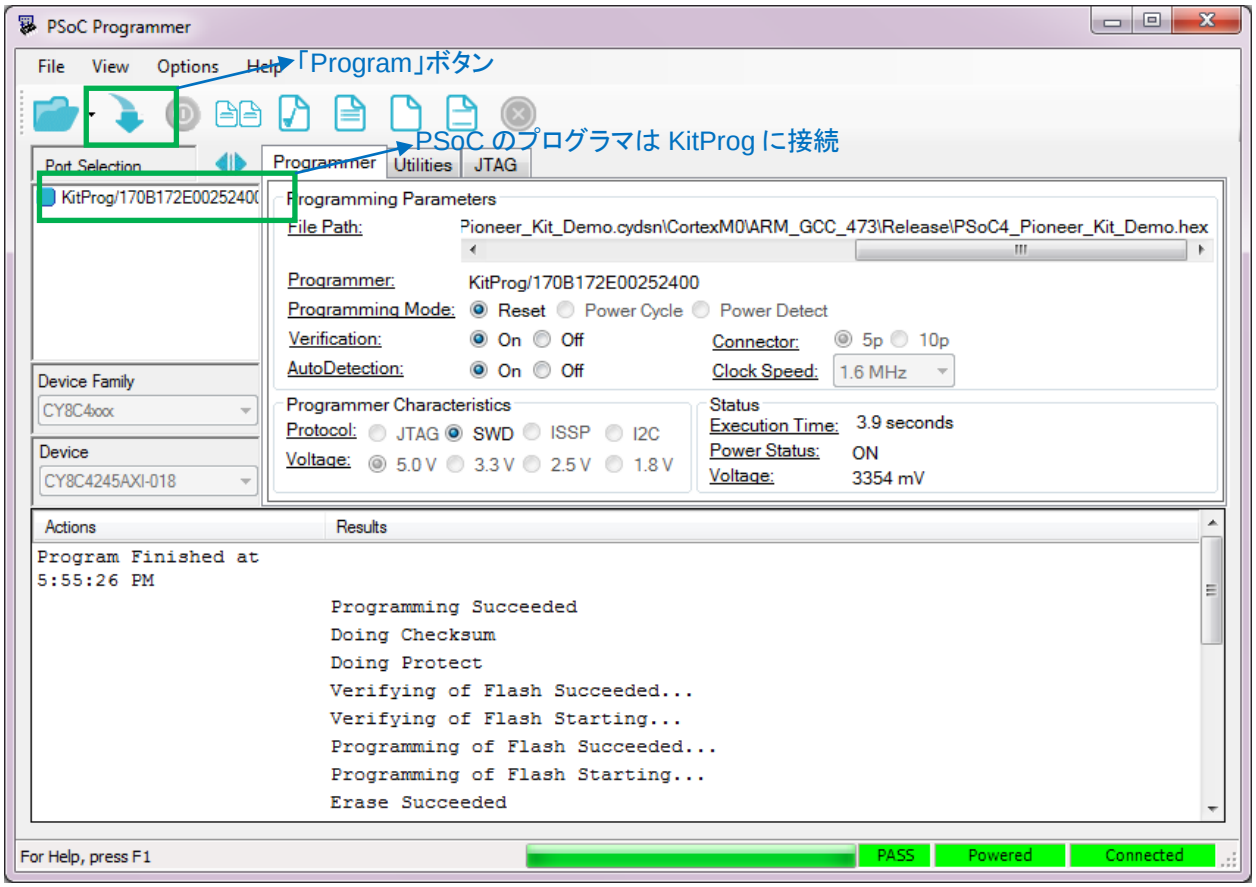
6. ここから「CY8CMBR3xxx_Host_APIs_Demo_Project.zip」プロジェクトをダウンロードし、.zip ファイルからコンテンツを抽出します。
7. PSoC のプログラマを開きます。これは、Windows のスタートメニュー→すべてのプログラム→Cypress→PSoC プログラマー 3.20.0 で置かれます。PSoC のプログラマが PC にインストールされていない場合、PSoC プログラマ 3.20.0 からダウンロードして、インストールしてください。

8. File→File ロード上に PSoC プログラマクリックで「CY8CMBR3xxx_Host APIs_Demo_Project」が抽出されているディレクトリに移動します。
9. CY8CMBR3xxx_Host_APIs_Demo_Project→ PSoC4_Pioneer_Kit_Demo→ PSoC4_Pioneer_Kit_Demo.cydsn→CortexM0→ ARM_GCC_473→Release→PSoC4_Pioneer_Kit_Demo.hex に移動して、オープンをクリックしてください。
10. 図 5-20 に示すように、PSoC Programmer が KitProg に接続されていることを確認してください。図 5-20 に示すように、与えられた hex ファイルと PSoC4 のパイオニア キットをプログラムするプログラムのボタンをクリックします。
11. PSoC4 パイオニア キットがプログラムされると、PSoC4 パイオニア キットは、デモのコンフィギュレーションを伴い MBR3 評価キット上の CY8CMBR3116 コントローラーを設定します。CY8CMBR3116 コントローラーが正常に設定されている場合は、PSoC4 パイオニア キット上の RGB LED D9 が緑色に点灯します。CY8CMBR3xxx コントローラーが正常に設定されていない場合は、PSoC4 パイオニア キット上の RGB LED D9 が赤色に点灯します。

    MBR3 評価キットは以下の用に設定されています。
    - 4 個の CapSense のボタン センサー
    - 1 個の近接センサー
    - 1 個の近接 LED
    - ブザー出力
    - ホスト割り込み出力
    - GPO を制御するホスト

    MBR3 評価キットの LED は、CY8CMBR3116 コントローラーがホスト割り込みパルスを送信するたびに、ホスト プロセッサー (PSoC4) によってオン／オフされます。ホスト割り込みがトリガーされると、ホストは CY8CMBR3116 コントローラーのすべてのセンサーの状態を読み出し、センサーの状態が「1」に設定されている場合、対応する LED をオンにします。
12. PSoC4 ホストが CY8CMBR3116 コントローラーを設定した後、MBR3 評価キット上で次の機能を確認します。
    - 近接検出: MBR3 評価キットの上に手／指 (3cm の距離) を被せると、一回近接 LED が点滅し近接検出を示します。近接センサー (MBR3 キットの縁に白い四角形でマークされている) に触れると、ブザーは 500 ミリ秒間鳴り、近接 LED は近接センサーが触れている限り ON を維持します。手／指が既に近接センサーに触れている場合は、LED は OFF になり近接接触がもう検出されません。手／指を離すと近接 LED が点滅して、再度近接が検出されたことを示します。
    - CapSense ボタン: いずれかの CapSense のボタンをタッチすると、対応する LED が点灯し、ブザーが 500 ミリ秒間鳴ります。

図 5-20. PSoC のプログラマを使用した PSoC4 パイオニア キットのプログラミング



5.2.4 ブリッジ コントロール パネルを使用し CY8CMBR3xxx のコンフィギュレーション

ブリッジ コントロール パネル (BCP) は、図 5-21 に示すように、CY3240-I2USB または MiniProg3 ハードウェアを使用して、I²C スレーブ デバイスと通信するために I²C マスターとして使用できるソフトウェア ツールです。

図 5-21. I²C スレーブ デバイスとブリッジ コントロール パネルの通信



注: CY3280-MBR3 評価キットは、追加の I2USB/ MiniProg3 ハードウェアが不要です。このキットは、BCP に接続するための基板搭載の I2C-USB ブリッジを持っています。オンボード I2C-USB ブリッジを BCP に接続するには、MBR3 評価キット上のヘッダ J13 のジャンパー位置を変更し、J13-1 ピンと J13-2 ピンとの接続と J13-1 ピンと J13-2 ピンを接続するためにヘッダ J14 上に接続します。

このツールは、I²C インタフェースを使用して CY8CMBR3xxx コントローラーを設定し、デバッグ データを読み出すために使用できます。BCP の代わりに CY8CMBR3xxx コントローラーを設定するために、EZ-Click を使うのを強くお勧めします。しかし、BCP が CY8CMBR3xxx コントローラーをコンフィギュレーションするために使用できる状況があるかもしれません。例えば、閾値パラメーターを無視して、それらを手動で指定するのに、BCP ツールを使用します。BCP ツールの使用方法の詳細については、BCP ツールにある BCP の「Help」メニューをクリックします。

この節では、CY8CMBR3xxx コントローラー内のさまざまなレジスタ名を参照します。これらのレジスタの定義とアドレスについては、CY8CMBR3xxx のレジスタ TRM を参照してください。

5.2.4.1 ブリッジ コントロール パネルを使用して CY8CMBR3xxx のコンフィギュレーション手順

BCP ツールを使用して CY8CMBR3xxx をコンフィギュレーションするためのフローチャートを図 5-24 に示します。BCP を使用して CY8CMBR3xxx と通信するために必要なハードウェア接続の詳細は、図 5-13 を参照してください。

BCP ツールを使用して CY8CMBR3xxx を設定するための手順は、以下の 4 ステップに分けることができます。

1. SeCY8CMBR3xxx コントローラーにコンフィギュレーション データを送信します。コンフィギュレーション データは以下のいずれかです。

   a) EZ-Click で生成されたコンフィギュレーション ファイル (<project_name>.iic file) :
   EZ-Click により生成されるコンフィギュレーション ファイルを使用して CY8CMBR3xxx コントローラーをコンフィギュレーションする方法は図 5-22 に示します。CY8CMBR3xxx が EZ-Click で直接コンフィギュレーションすることができるため、この方法はあまり使用されません。スレーブ デバイスにコマンドを BCP ツール内いずれかの IIC (コンフィギュレーション) ファイルを開いて送信する手順については、BCP ツールで「Help」オプションをクリックします。

図 5-22. BCP を使用して CY8CMBR3xxx コントローラーをコンフィギュレーションするために EZ-Click によって生成された IIC ファイル

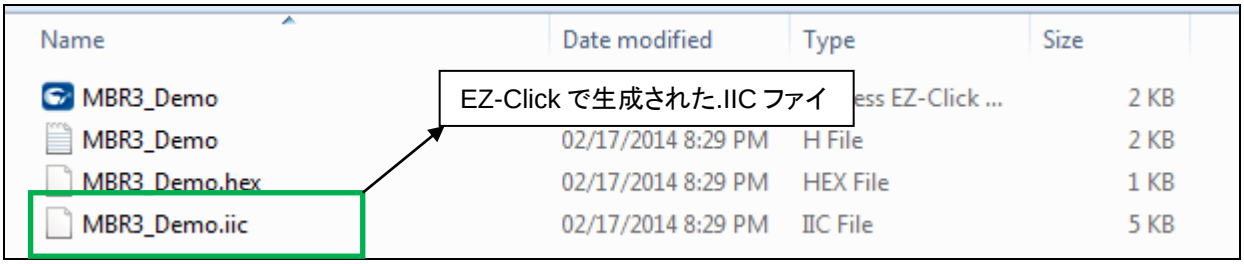
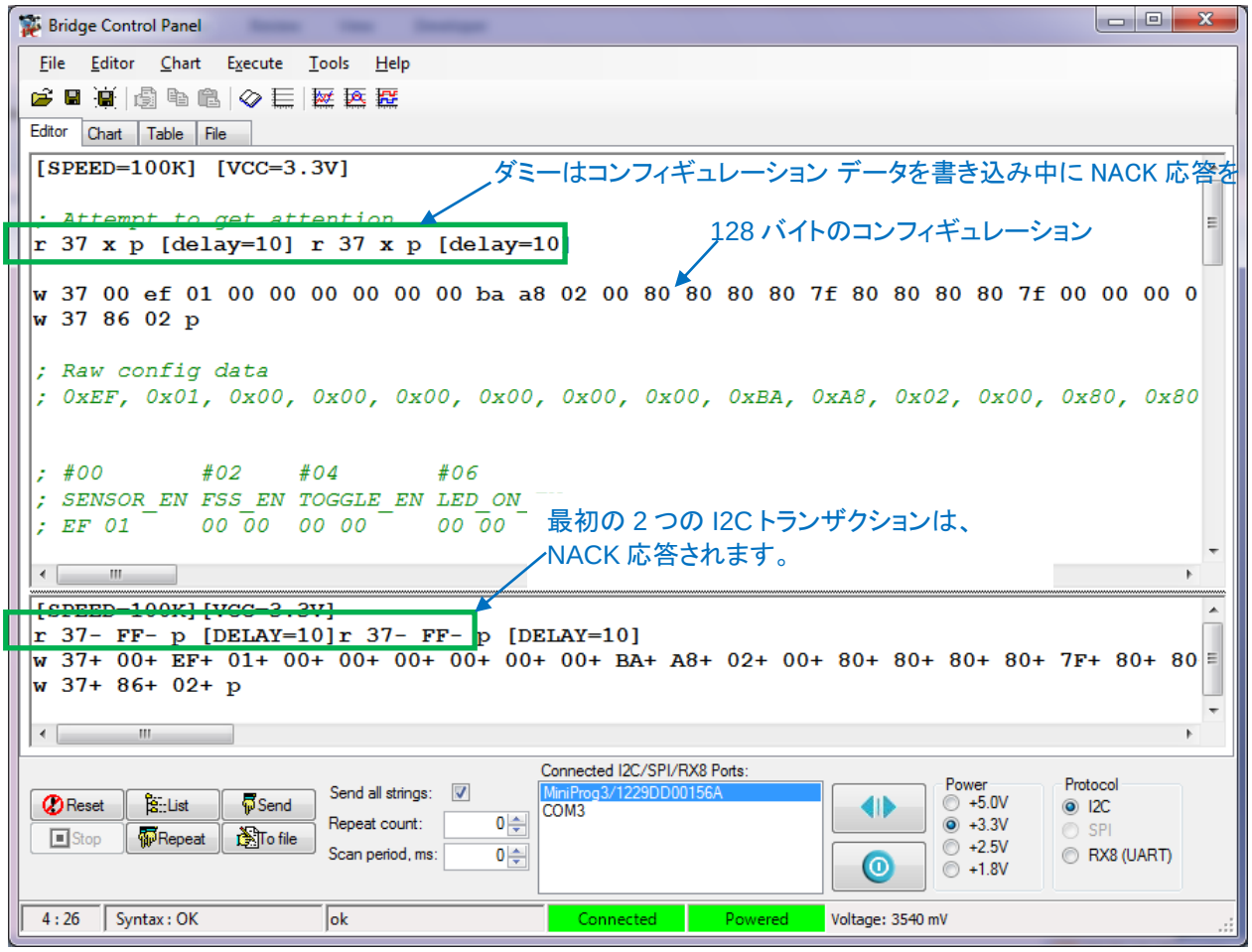


図 5-23. EZ-Click カスタマイザ ツールが生成したコンフィギュレーション ファイルの例



b) 以下の BCP ツールに入力されたカスタムコマンド:

BCP 内にカスタムコマンドを入力した CY8CMBR3xxx のコンフィギュレーションを図 5-26 に示します。この方法で全 128 バイトのコンフィギュレーション レジスタ、または任意の特定のレジスタを設定できます。CY8CMBR3xxx がカスタム コマンドを使用して設定されている場合は、コンフィギュレーション データの最初の 126 バイトの CRC (巡回冗長検査) を計算して、それを CONFIG_CRC (0x7E の) レジスタに書き込む必要があります。この CRC 値はコンフィギュレーション データの整合性を検証するために CY8CMBR3xxx コントローラーで使用されます。

注: I²C コマンド (読み取り/書き込み) では、I²C トランザクションで NACK かもしれない CY8CMBR3xxx コントローラーに送信されます。この場合は、図 5-26 に示すように、任意のレジスタを読み書きしている間にダミーの I²C の読み出しトランザクションを挿入します。

コンフィギュレーション データの最初の 126 バイトの CRC 値は、次の 2 つの方法で計算することができます。

- CRC 値を計算するために CY8CMBR3xxx コントローラを使用します。図 5-25 に示すように、CY8CMBR3xxx コントローラは、「3」の CMD_OP_CODE 値が、CTRL_CMD (0x86) レジスタに書き込まれる時にコンフィギュレーション データのチェックサムを計算します。計算された CRC 値は、CALC_CRC (0x94) レジスタに 220 ミリ秒後に格納されます。ユーザーが CALC_CRC (0x94) から CRC 値を読み出し、手動で CONFIG_CRC (0x7E を) に書き込む必要があります。この方法は、テストして CY8CMBR3xxx コントローラーをデバッグしている間にのみ使用してください。本番環境での CY8CMBR3xxx コントローラーを設定するために、ホスト API で提供された CRC API を使用します。

- RC 値を計算するために、ホストの API を使用します。サイプレスは、ホスト プロセッサを使用して CY8CMBR3xxx を設定するためにホスト API を提供しています。ホスト API は、126 バイトのコンフィギュレーション データを CRC 値を計算するために使用することができるソース コードを持っています。図 5-26 に示すように、CRC を計算するためにこのコードを使用して、計算された CRC 値を CONFIG_CRC (0x7E) に書き込む必要があります。CRC を計算するための手順は、5.2.4.2 節で説明します。CRC 値を計算するためにこの方法を使用することを強くお勧めします。

2. コンフィギュレーション データと CRC の値が、CY8CMBR3xxx コントローラーのレジスタに書き込まれると、CTRL_CMD (0x86) レジスタに「2」の CMD_OP_CODE 値を書き込み、不揮発性メモリにンフィギュレーション データを保存します。
3. CTRL_CMD (0x86) レジスタに「2」の CMD_OP_CODE 値を書き込んだ後で、220 ミリ秒間待ってから、コンフィギュレーション データが正常に不揮発性メモリに保存されているかをチェックするために、CTRL_CMD_STATUS (0x88) レジスタを読み出します

   コンフィギュレーション データが正常に不揮発性メモリに保存されている場合、CTRL_CMD_STATUS (0x88) レジスタの値が「0」になります。CTRL_CMD_STATUS レジスタの値が、「0」である場合、255 の CMD_OP_CODE 値を CTRL_CMD (0x86) レジスタに書き込むことによって、コマンドをリセットします。

   CTRL_CMD_STATUS (0x88 の) レジスタの値が「1」である場合、コンフィギュレーション データは、不揮発性メモリに保存されていないことを意味します。この場合、失敗の原因を知るために CTRL_CMD_ERR (0x89) を読み出し、コンフィギュレーション データを不揮発性メモリに保存します。

   - CTRL_CMD_ERR の「253」の値は失敗した不揮発性メモリに書き込むことを示しています。この場合、コンフィギュレーション データを再作成し、手順 2 から繰り返します。
   - CTRL_CMD_ERR レジスタ内の「254」の値は CONFIG_CRC (0x7E) に書き込まれた CRC 値が正しくないことを示しています。この場合、コンフィギュレーション データの全 126 バイトに応じて CRC を再計算し、CONFIG_CRC (0x7E) レジスタに書き込み、手順 2 から繰り返します。
   - CTRL_CMD_ERR レジスタ内の「255」の値は、CTRL_CMD (0x86) レジスタに書き込まれた CMD_OP_CODE 値が有効でないことを示します。この場合、有効な CMD_OP_CODE=「2」を送信し、ステップ 3 を繰り返します。

図 5-24. CY8CMBR3xxx を使用した BCP コンフィギュレーションのフローチャート

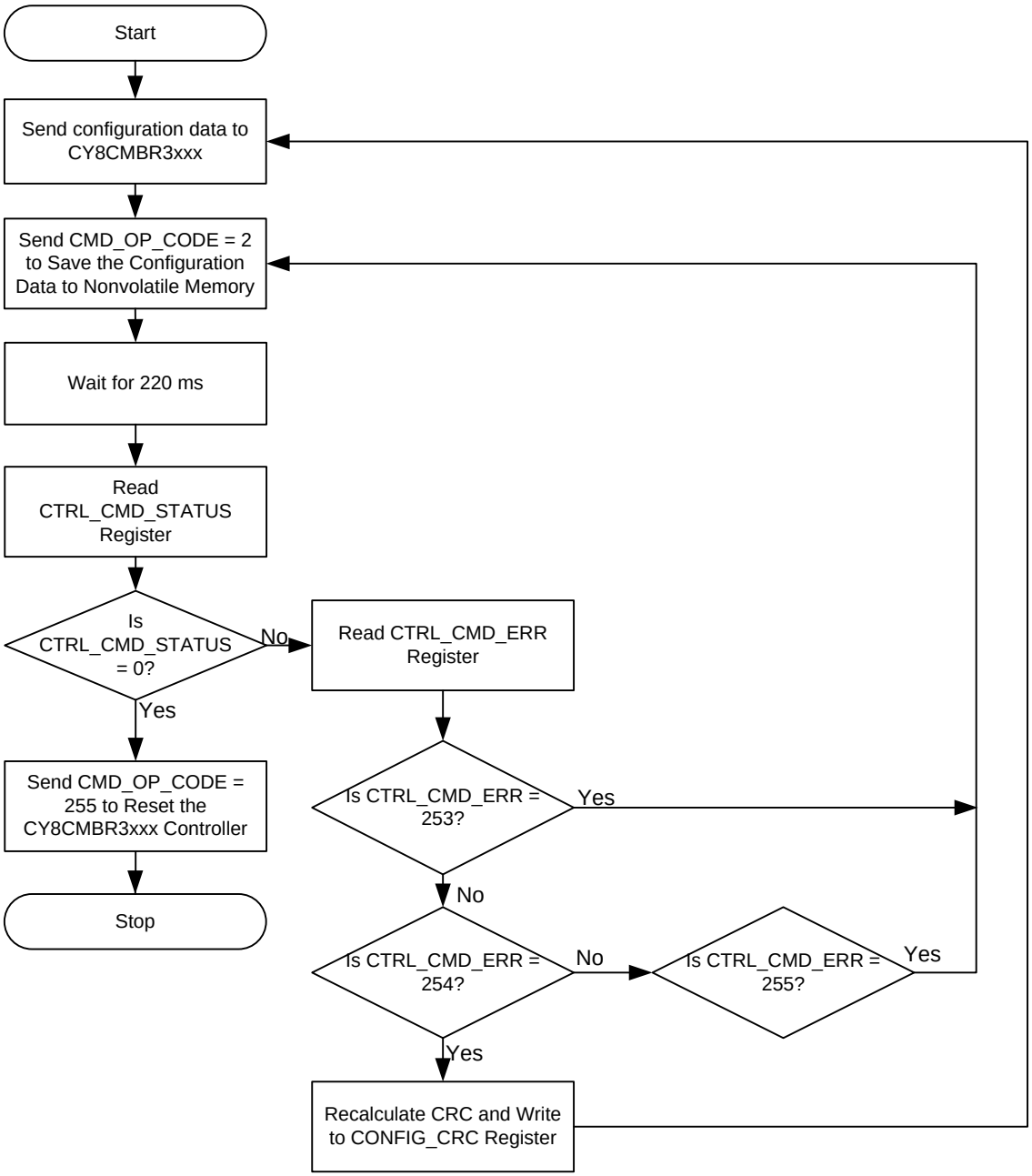
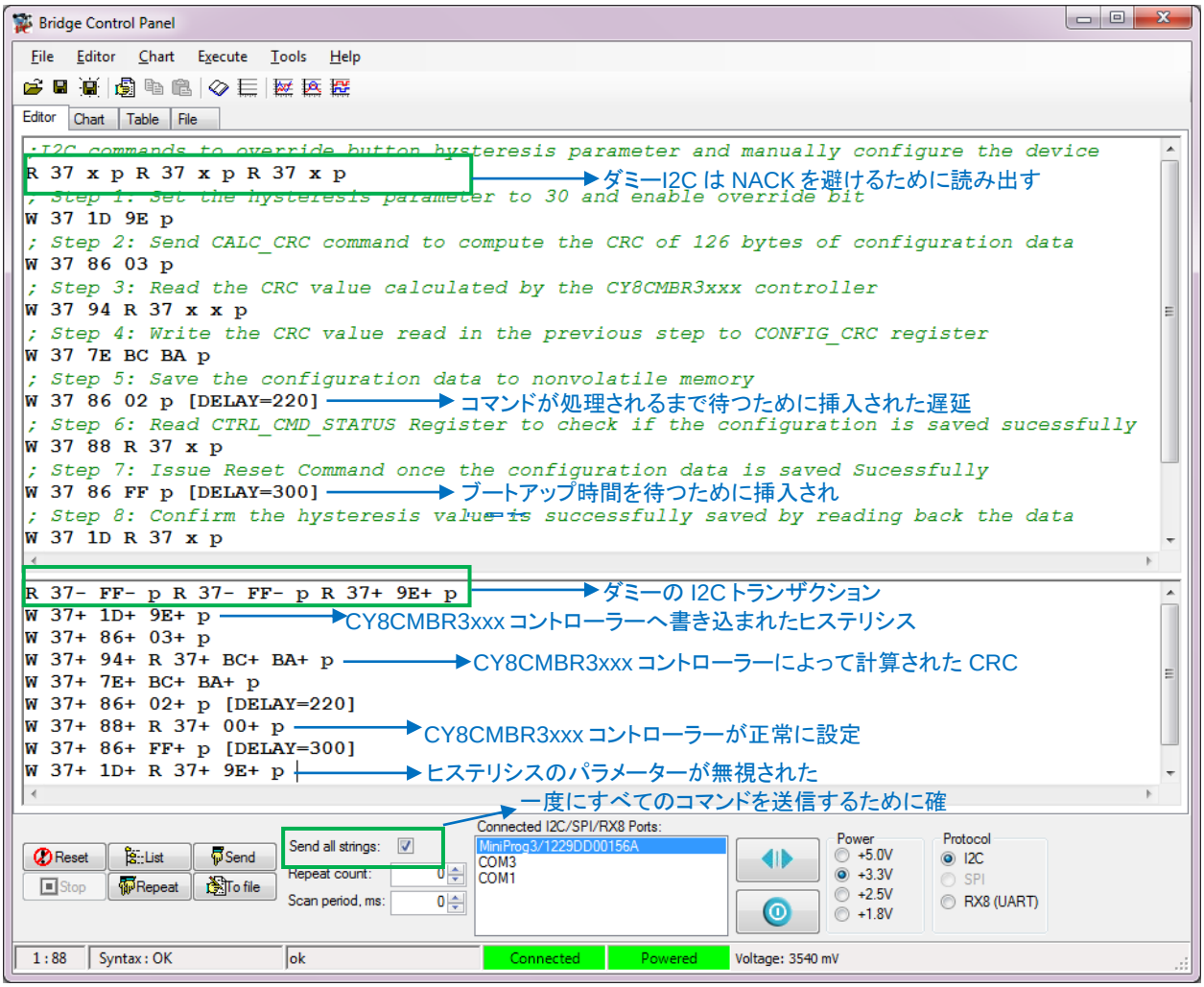
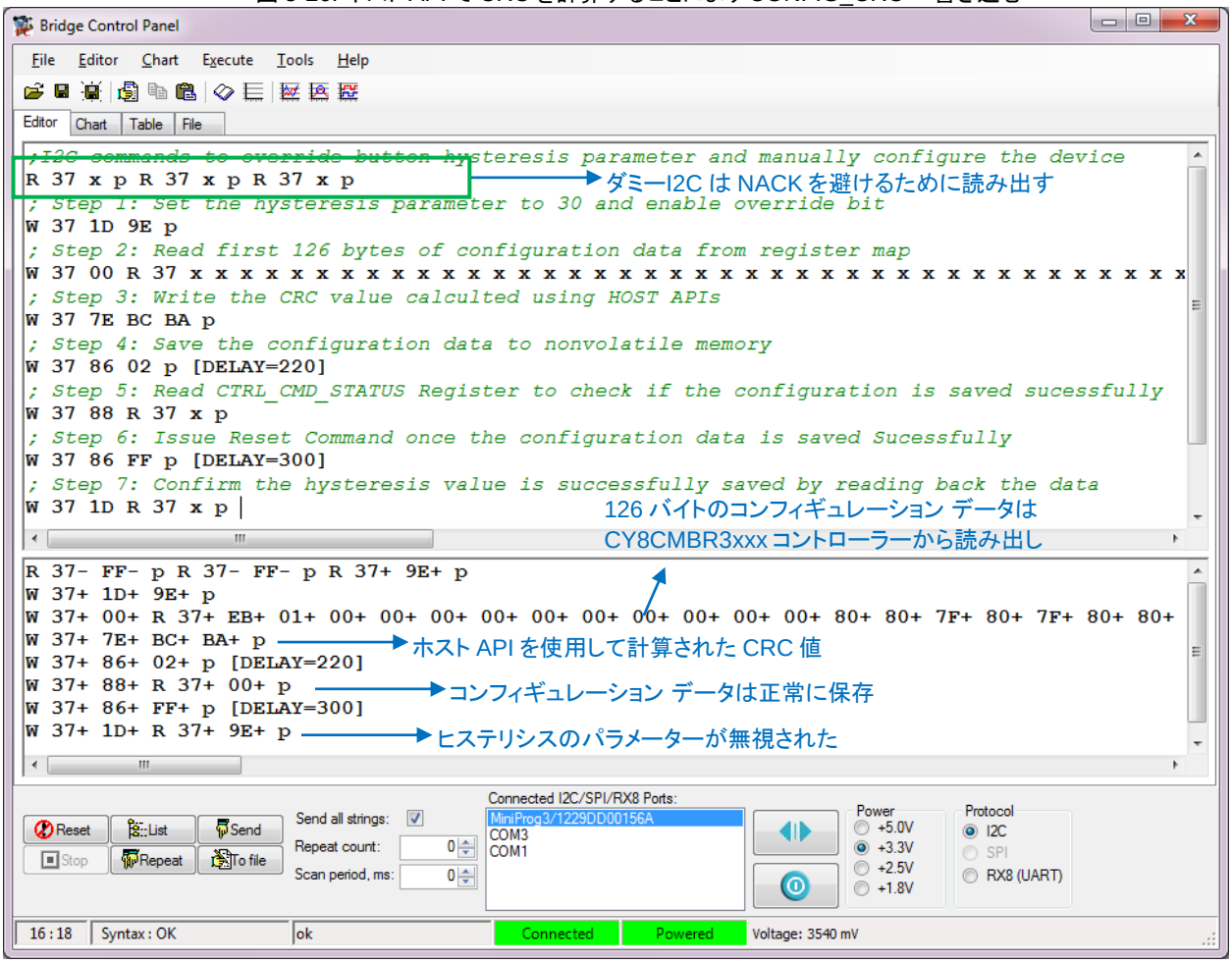


図 5-25. CY8CMBR3xxx コントローラーを使用した CRC の計算



注: BCP に入力されたすべてのコマンドは、「send all strings」ボックスをチェックし、送信ボタンをクリックし、一度に送信することができます。

図 5-26. ホスト API で CRC を計算することにより CONFIG_CRC へ書き込む


Bridge Control Panel
File Editor Chart Execute Tools Help
Editor Chart Table File
;I2C commands to override button hysteresis parameter and manually configure the device
R 37 x p R 37 x p R 37 x p
ダミーI2C は NACK を避けるために読み出す
; Step 1: Set the hysteresis parameter to 30 and enable override bit
W 37 1D 9E p
; Step 2: Read first 126 bytes of configuration data from register map
W 37 00 R 37 x x x

```
{
        /* Variable to store the 16-bit CRC value*/
        unsigned short crcValue;

        /* Call the "CY8CMBR3xxx_CalculateCrc" API with configuration data address as the argument*/
        crcValue = CY8CMBR3xxx_CalculateCrc(CY8CMBR3xxx_configuration);

        /* Print the calculated 16-bit CRC value*/
        printf("The CRC for the given configuration data is: 0x%x",crcValue);

        return 0;
}
```

5. BCP を使用して CY8CMBR3xxx コントローラから 126 バイトのコンフィギュレーション データを読み出し、テキスト エディタ ソフトウェアにこれをコピー (Notepad などに) し、図 5-27 と図 5-28 に示すようにコピーされたアレイからすべての「+」記号を削除するために、データなどをフォーマットして、各バイトデータ前の「,0x」キャラクタに置き換えます。

図 5-27. BCP から Notepad にコピーする CY8CMBR3xxx コンフィギュレーション データ

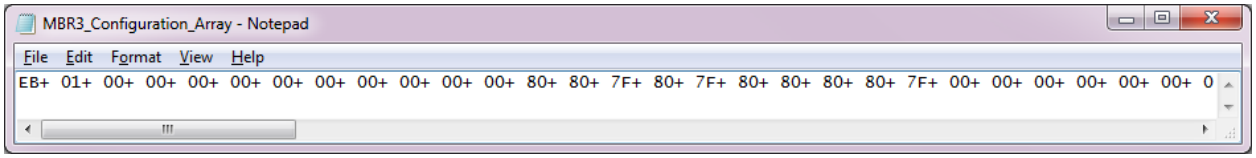
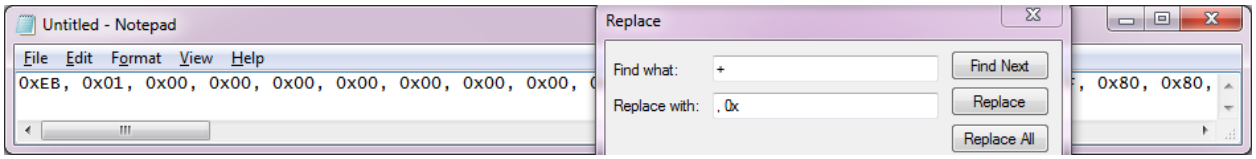
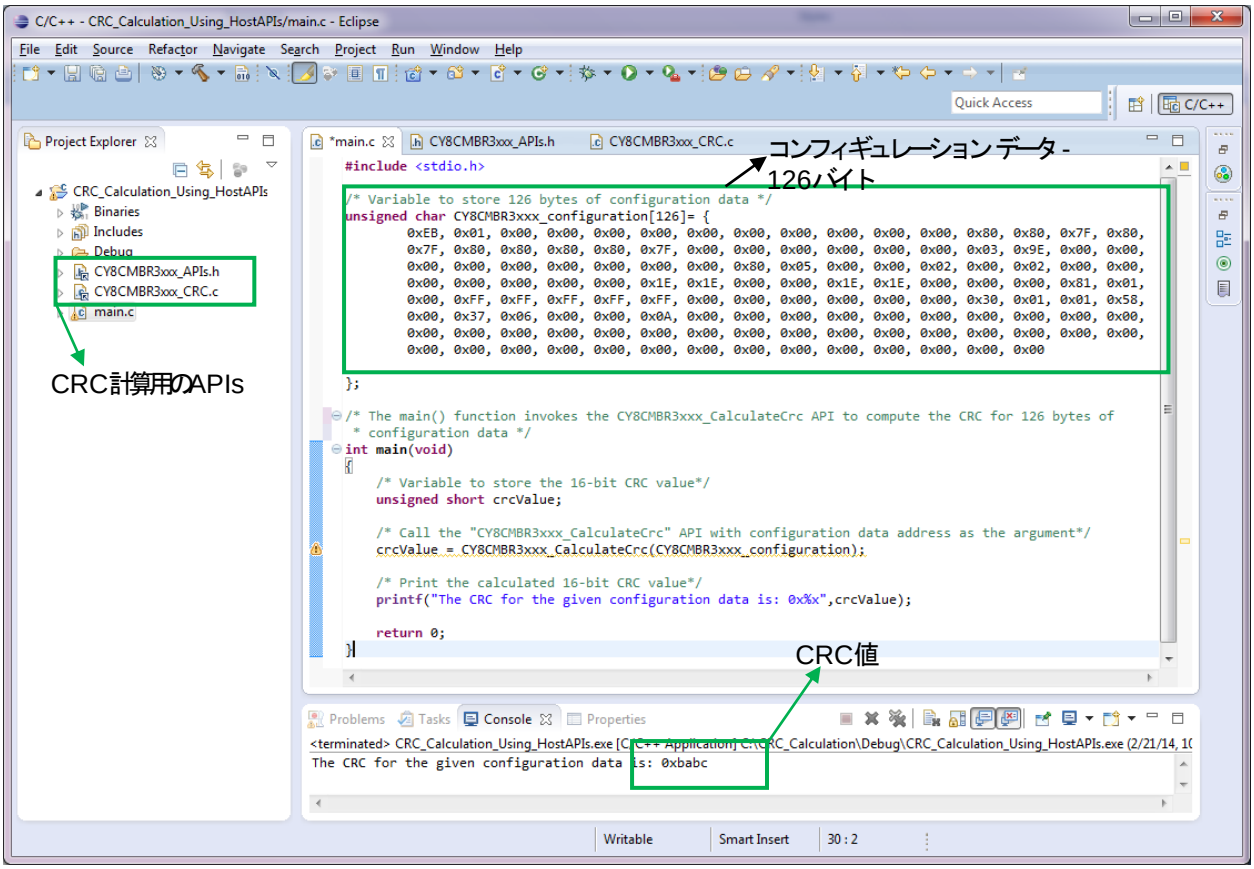


図 5-28. Notepad 内でフォーマットされる CY8CMBR3xxx コンフィギュレーション データ



6. 図 5-28 に示すように、main.c 内の「CY8CMBR3xxx_configuration[126]」アレイの中に、このフォーマット化されたアレイをコピーします。
7. プロジェクトを保存し構築します。エラーがないことを確認してください。
8. プロジェクトが正常に構築されると、CRC 値を取得するには「Run」ボタンをクリックしてください。

図 5-29. Eclipse の C/C++ IDE 内でホスト API を使用して CRC 値を計算



6. CapSense 性能のチューニング

「CapSense の技術」の章で説明したように、サイプレスの SmartSense の自動チューニング アルゴリズムは、ハードウェアとファームウェアのパラメータを自動的に調整することにより手動でのチューニングのニーズを解消します。しかし、SmartSense アルゴリズムは、各センサーの SNR が 5:1 を超えることを確認するためにセンサーの感度パラメータの設定を必要とします。CY8CMBR3xxx により、閾値パラメーターを手動で設定することができます。本節で、信頼できる接触が検知される時のセンサー感度と閾値パラメーターを設定する方法について説明します。CY8CMBR3002 CapSense コントローラーは手動チューニングをサポートしません。このコントローラーでは SmartSense アルゴリズムは自動的にすべてのハードウェアとファームウェア パラメーターをチューニングします。そのため、次の節は CY8CMBR3002 コントローラーに適用されません。

6.1 一般的な注意事項

6.1.1 信号、ノイズ、および SNR (信号対ノイズ比)

適切にチューニングされた CapSense システムによって、オンとオフのセンサー状態が確実に識別されます。良い性能を得るには、CapSense 信号は CapSense ノイズよりかなり大きくする必要があります。CapSense 信号は、信号対ノイズ比 (SNR) という量を使用して CapSense ノイズと比較されます。SNR の詳細については、Getting Started with CapSense に掲載している「CapSense 技術」の章を参照してください。CapSense のベスト プラクティスに応じて、すべてのセンサーに対し最小 SNR を 5:1 に設定する必要があります。センサーの SNR は、以下の式で計算されます。

$$SNR = \frac{Signal}{Peak\text{-}to\text{-}Peak\ Noise}$$ 式 4

ここでは、信号 = Raw カウント – ベースライン

Peak-to-Peak Noise=3000 サンプルにわたって Raw カウントで測定される Peak-to-Peak ノイズ

そのため、高い SNR を達成するための重要な方法は信号を増やし、ノイズを減少することです。

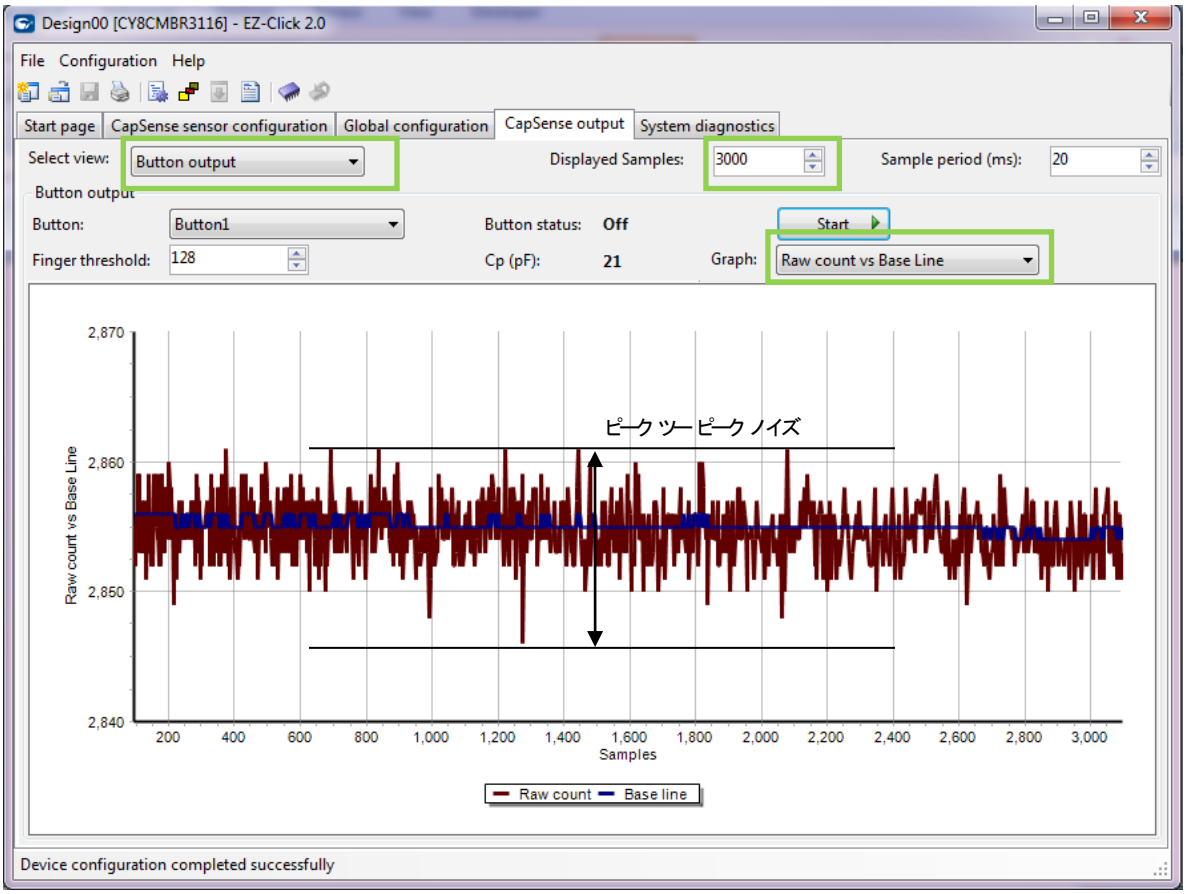
6.1.1.1 ノイズ カウントの測定

SNR を計算するために、センサーの信号とノイズ カウントを測定する必要があります。

EZ-Click を使ってノイズ カウントを測定するには、以下の手順を行ってください。

1. コンフィギュレーション ファイルを生成して、設定をデバイスにダウンロードします。
2. 「CapSense output」タブへ進んで、「Raw count vs Baseline」のグラフを選択します。「Displayed Samples」フィールドを 3000 に設定します。
3. 「Start」ボタンをクリックして、3000 サンプルとも取得するまで待ちます。
   注: ノイズ カウントを測定するために Raw カウント サンプルを取得中にセンサーに触れないでください。
4. 図 6-1 に示すように、「Raw カウント (最大値) ー Raw カウント (最小値) 」の式を使ってノイズ カウントを測定します。図 6-1 に示しているグラフのノイズ カウントは、「2860 – 2845 = 15」カウントです。

図 6-1.EZ-Click カスタマイザでのノイズの測定



6.1.1.2 信号測定

ノイズ カウントを計算した後、センサーの信号を測定し SNR を計算する必要があります。

信号を測定するために、以下の手順に従ってください。

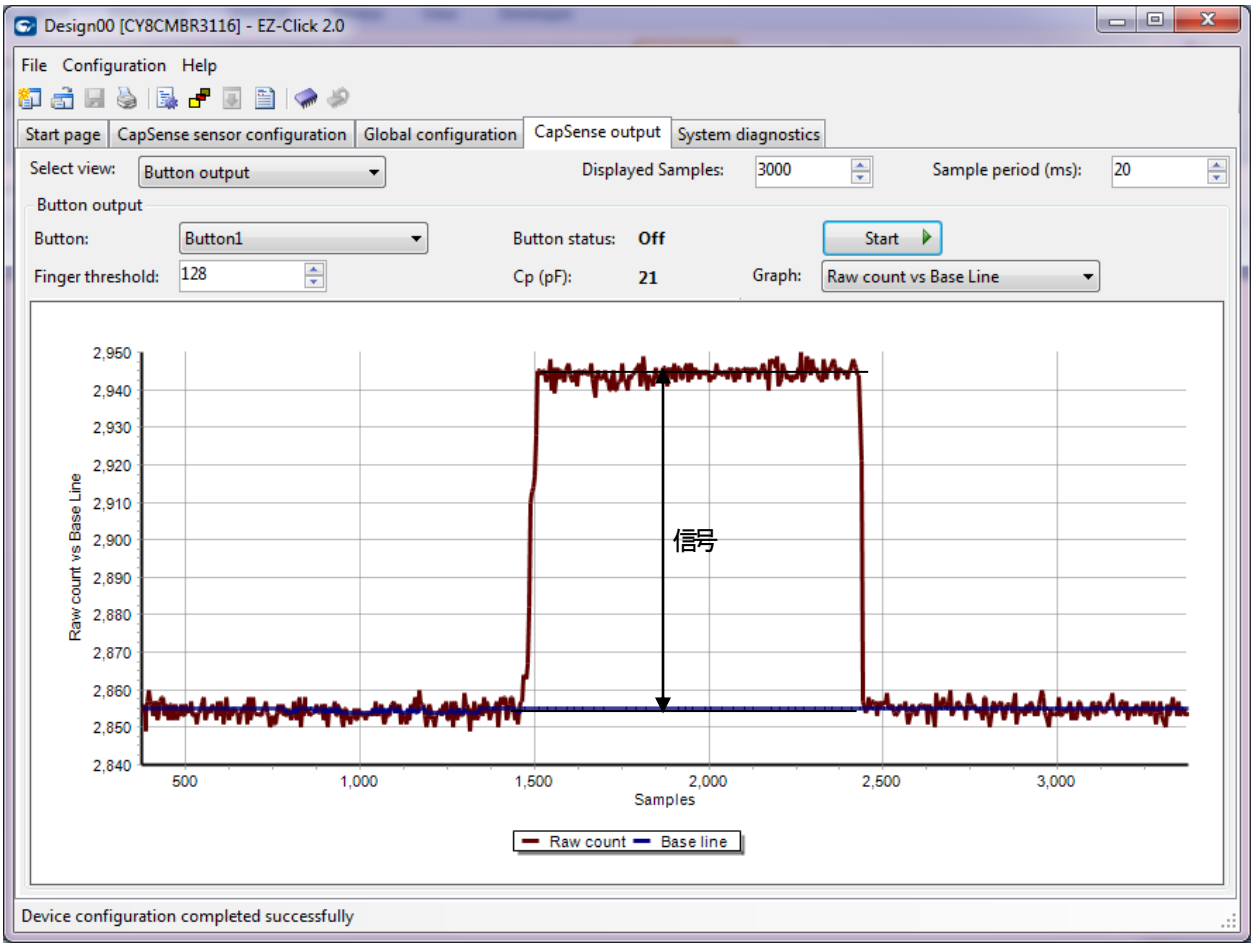
1. 「CapSense output」タブでは、「Start」ボタンをクリックして、「Raw count vs Baseline」グラフを表示させます。
2. 1000 個のサンプルの間でセンサーに接触してから指を離します。
3. 図 6-2 に示すように、*非接触*状態から接触状態への遷移を Raw カウントで測定します。

信号=Raw カウントの平均値 (センサー接触時) - Raw カウントの平均値 (センサー非接触時)

図 6-2 に示すグラフの場合、信号=2945-2855=90 カウントです。

SNR = $\frac{90}{15}$= 6:1

図 6-2.EZ-Click のカスタマイザでの信号の測定



6.2 ボタン、スライダー、およびガード センサーのチューニング

図 6-3 に、ボタン、スライダー、およびガード センサーのチューニングの標準的なフローを示します。

CY8CMBR3xxx コントローラー内のセンサーのチューニングは、2 つの段階があります。

1. SNR が 5:1 より大であることを確認
2. 閾値のパラメーターを推奨値に設定

6.2.1 最小限の SNR の確保方法

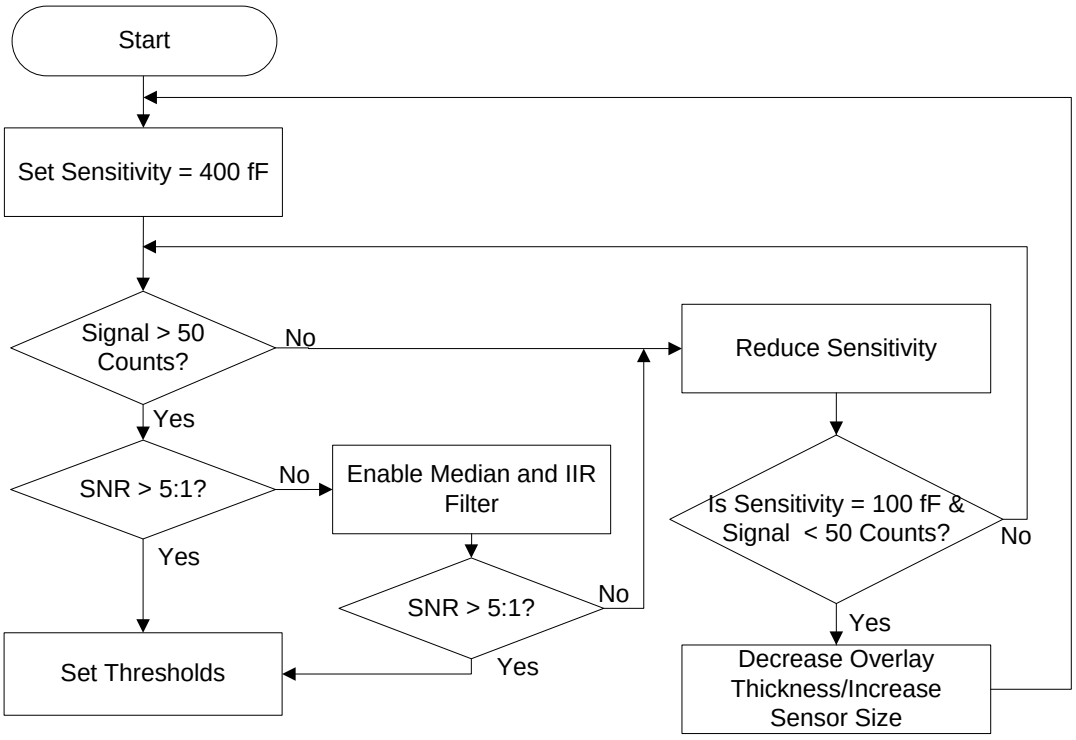
SNR が 5:1 より大きいことを確認するために、これらの手順を行ってください。

1. センサーの感度を 400fF に設定し、信号を測定します。
2. 信号が 50 カウントより小さい場合、感度値を信号が 50 を超えるまで下げます。感度値を最小値 (100fF) に設定しても信号がまだ 50 カウント以下の場合、オーバーレイの厚さを薄くするかセンサーのサイズを増加します。
3. 信号が 50 カウントを超える場合、ノイズ カウントを測定します。ノイズが信号の 5 分の 1 より大きい場合、ノイズを制限するためにフィルタを有効にします。IIR フィルタを有効にして、ノイズ カウントを測定します。ノイズが依然として信号の 5 分の 1 より大きい場合、メジアン フィルタを有効にします。メジアン フィルタが有効になった場合、自動閾値機能を無効にし、指閾値パラメーターを手動で指定する必要があります。
4. SNR を計算し、SNR が 5:1 より大きいなら、閾値のパラメーターを設定します。

5. フィルタを有効にしても SNR がまだ 5:1 以下の場合、感度値を低下して、SNR を計算します。感度値を最小値 (100fF) に設定して、フィルタを有効にした後、SNR が 5:1 未満の場合、SNR が 5:1 を超えるようにオーバーレイの厚さを薄くするか、またはセンサーのサイズを増加します。

注: スライダーの場合、正確な図心位置を取得するために、信号はいつでも 255 未満でなければなりません。信号が 255 を超える場合、信号が 255 未満になるまで感度のパラメーターを上げます。

図 6-3.ボタン、スライダー、およびガード センサーのチューニング



6.2.2 ボタン閾値パラメーター

すべてのセンサーの SNR が 5:1 より大きいであると確認した後、閾値パラメーターを指定します。

以下の閾値パラメーターはボタンとガード センサーに適用可能です。

- 指閾値
- ヒステリシス
- ノイズ閾値
- 負のノイズ閾値
- 低ベースライン リセット

6.2.3 節に、これらのパラメーターを設定する時点と設定方法について説明します。

6.2.2.1 指閾値

指閾値パラメーターはセンサーのアクティブ／非アクティブ状態を判定するために使用されます。センサーの差分カウントの値が「指閾値+ヒステリシス」より大きい場合、センサーはオンとして判定され、センサーの差分カウントが「指閾値–ヒステリシス」より小さい場合、センサーはオフとして判定されます。指閾値の範囲は31～200で、初期値が128です。表6-2を参照してこのパラメーターを設定します。

指閾値とセンサーのオン／オフ状態の関係は以下の式て示します。

$$\text{Sensor State} = \begin{cases} \text{ON} & \text{if (Signal} \geq \text{Finger Threshold} + \text{Hysteresis)} \\ \text{OFF} & \text{if (Signal} \leq \text{Finger Threshold} - \text{Hysteresis)} \end{cases}$$ 式5

6.2.2.2 ヒステリシス

ヒステリシス パラメーターは指閾値パラメーターとともに使用され、式センサーの状態を決定します。ヒステリシスはセンサー状態のノイズ遷移に対する耐性を提供します。このパラメーターはボタンとガード センサーに適用可能です。このパラメーターの範囲は0～31で、初期値が12です。表6-2を参照してこのパラメーターを設定します。

6.2.2.3 ノイズ閾値

図6-4に示すように、ノイズ閾値パラメーターはRawカウントの限界を設定します。この限界を超えたらベースラインが更新されません。つまり、Rawカウントが「ベースライン+ノイズ閾値」を超える限り、ベースラインが一定のままです。これにより、指の接触に起因してベースラインが高くなりすぎないように防ぎます。このパラメーターの範囲は0～127で、初期値が51です。表6-2を参照してこのパラメーターを設定します。

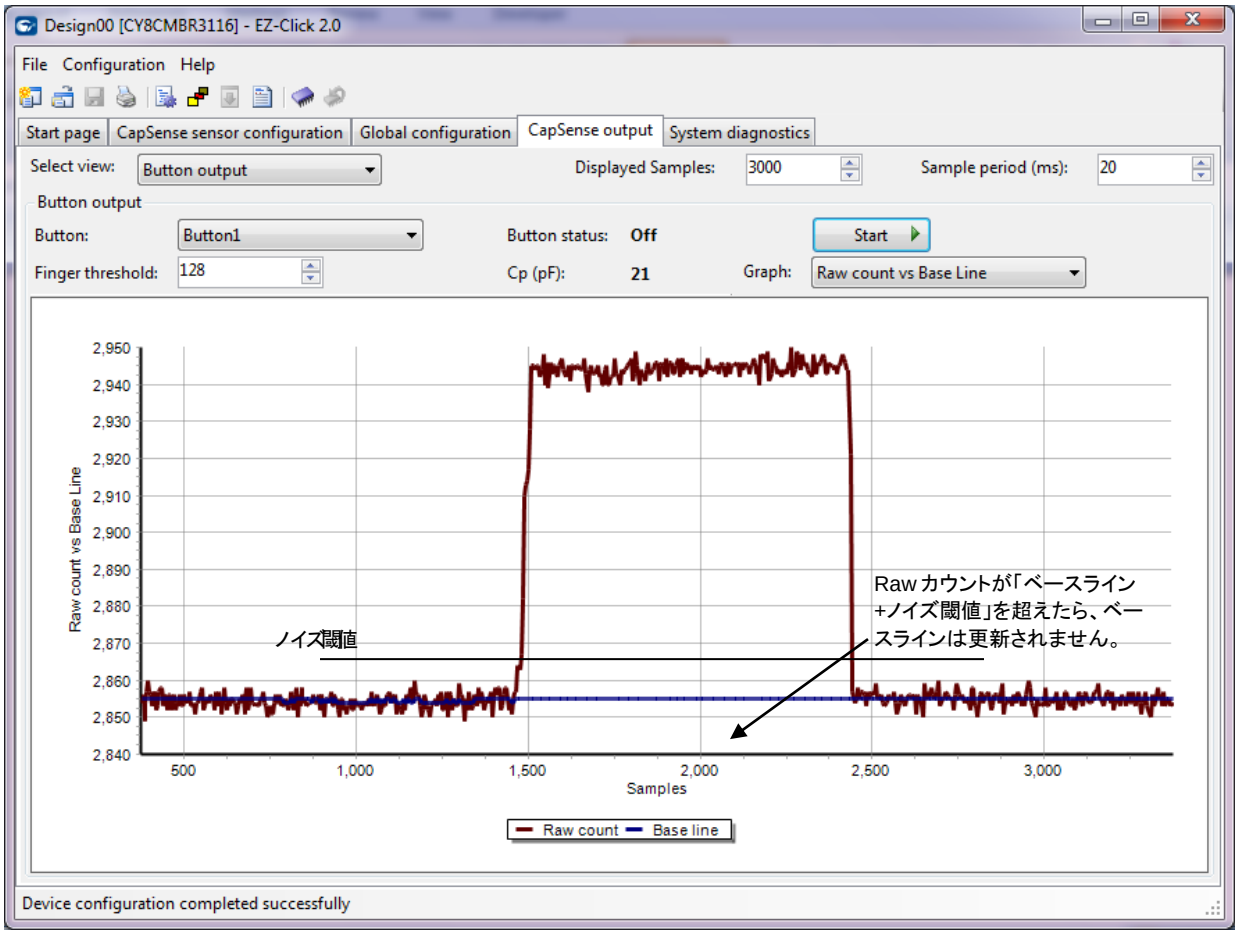
6.2.2.4 負のノイズ閾値

Rawカウントが、低ベースライン リセット パラメーターが指定するサンプル数の「ベースライン–負のノイズ閾値」より小さい場合、ベースラインは新しいRawカウント値にリセットされます。ベースラインのこの変化は、デバイスの電源がオン状態中にアクティブ状態になったままのセンサー (例えば、コントローラーの電源投入中に接触されるセンサー) をリセットします。このパラメーターの範囲は0～127で、初期値が51です。表6-2を参照してこのパラメーターを設定します。

6.2.2.5 低ベースライン リセット

ベースラインがRawカウントを超える時に、このパラメーターは、負のノイズ閾値パラメーターと共に使用されて、ベースラインをリセットします。これは、ベースライン値をリセットするのに必要となる異常に低い raw カウントの数を数えます。これは、デバイスが起動中に指がセンサーに触れてから離す場合、ベースラインのリセットに使用されます。このパラメーターの範囲は0～127で、初期値が50です。表6-2を参照してこのパラメーターを設定します。

図 6-4.ノイズ閾値



6.2.3 ボタン、スライダー、およびガード センサーの閾値設定

表 6-1 にボタン、スライダー、およびガード センサーに適用可能な閾値パラメーターを示します。

CY8CMBR3xxx は、ボタンの自動閾値機能を提供します。ユーザーが手動で指閾値を指定して要求しない限り、この機能は常に有効にする必要があります。

スライダーのセンサーは、自動閾値機能がスライダーセンサーに適用されないため、手動で指閾値パラメーターを指定するようにユーザーに要求します。SmartSense の自動チューニング アルゴリズムは、すべての他の閾値パラメーターを自動的に設定します。

ノイズ閾値、負のノイズ閾値、低ベースライン リセットを手動で指定するために、各閾値レジスタ内のそれぞれの閾値オーバーライド ビットをセットし、これらの閾値を手動で指定します。閾値のパラメーターを無効にしないことをお勧めします閾値オーバーライド ビットをセットし、ノイズ閾値、負のノイズ閾値、および低ベースライン リセット パラメーターを指定するには、ブリッジ コントロール パーネル (BCP) ツールを使用します。EZ-click は、閾値パラメーターを無効、指定するオプションを提供しません。

表 6-1.ボタン、スライダー、およびガード センサーの閾値パラメーター

閾値パラメーター	センサーの種類		
	ボタン	スライダー	ガード
指閾値	自動閾値機能が無効にされる時のみに設定	常に手動で設定	自動閾値機能が無効にされる時のみに設定
ノイズ閾値	ノイズ閾値オーバーライド ビットが有効にされる時のみに設定	スライダーのノイズ閾値オーバーライド ビットが有効にされる時のみに設定	ノイズ閾値オーバーライド ビットが有効にされる時のみに設定
負のノイズ閾値	負のノイズ閾値オーバーライド ビットが有効にされる時のみに設定	負のスライダーノイズ閾値オーバーライド ビットが有効にされる時のみに設定	負のノイズ閾値オーバーライド ビットが有効にされる時のみに設定
ヒステリシス	ヒステリシス閾値オーバーライド ビットが有効にされる時のみに設定	該当なし	ノイズ閾値オーバーライド ビットが有効にされる時のみに設定
低ベースライン リセット	低ベースライン リセット オーバーライド ビットがイネーブルされる場合にみ設定される	スライダーの低ベースライン リセット オーバーライド ビットがイネーブルされる場合にみ設定される	低ベースライン リセット オーバーライド ビットがイネーブルされる場合にみ設定される

注意: CY8CMBR3xxx が、ノイズ閾値を設定するためにボタンとスライダー専用の独立したレジスタ、および低ベースライン リセット パラメーターを備えています。

これらの閾値パラメーターを設定するには、以下の手順に従ってください。

1. CapSense 出力タブで、「Diff count vs. Finger threshold」グラフを選択して、スタート ボタンをクリックします。
2. 図 6-5 に示すようにセンサーに触れて、差分カウントを測定します。
   注:
   - CY8CMBR3xxx コントローラー の差分カウントは正規化されます。それは、「Raw カウント (接触時) -ベースライン」の結果に等しくありません。
   - 差分カウントが 255 より大きい場合、255 に切り捨てられます。この場合、閾値パラメーターを正確に設定できません。感度パラメーターの値を差分カウントが 255 未満になるまで上げます。6.2.1 節に示す手順を従って感度パラメーターの値を変更した後に、SNR が 5:1 を超えたことを確認します。
3. 閾値のパラメーターを表 6-2 で示す値に設定します。

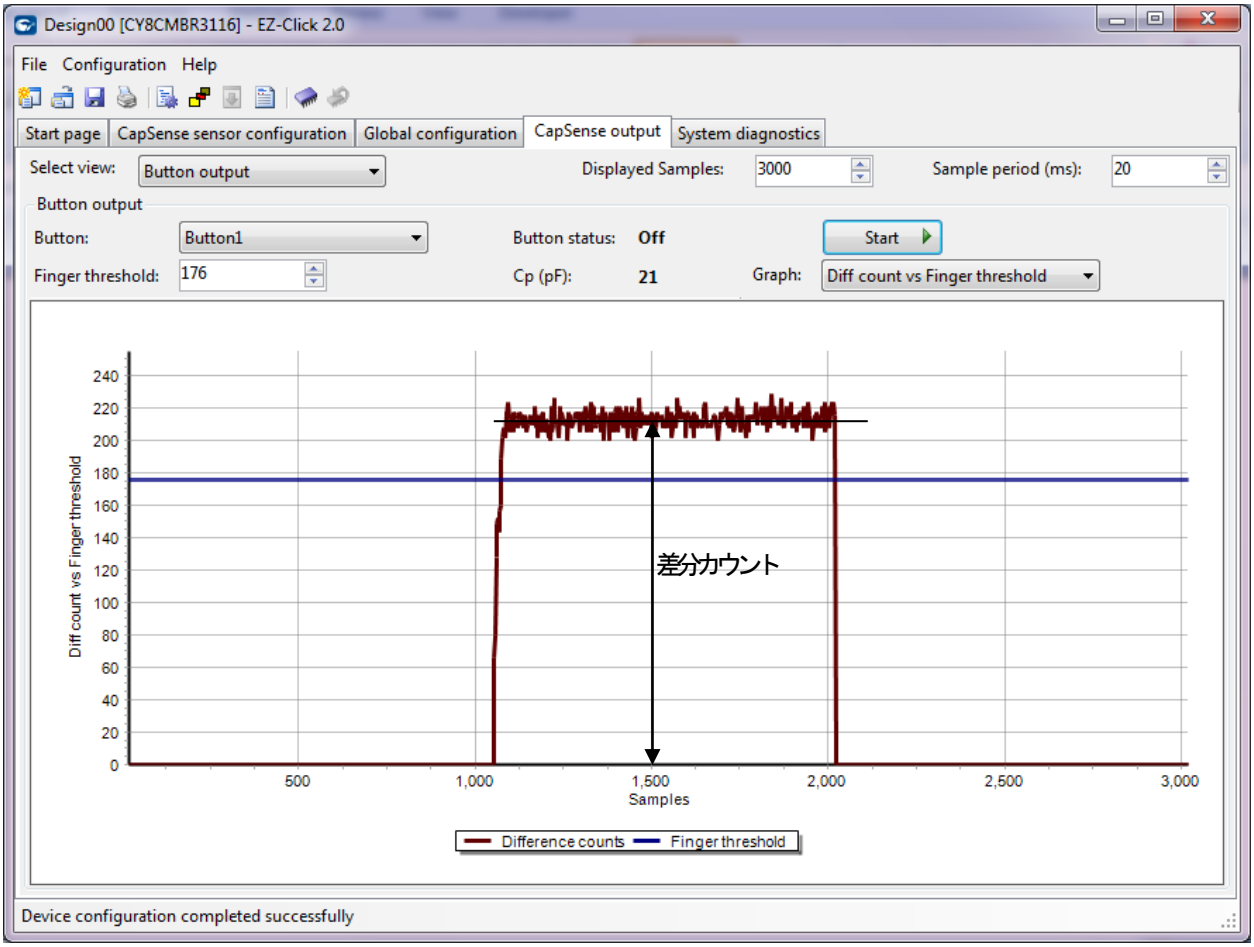
表 6-2.閾値パラメーターの値

閾値パラメーター	推奨値
指閾値	差分カウントの 80%
ノイズ閾値	差分カウントの 40%
負のノイズ閾値	差分カウントの 40%
ヒステリシス	差分カウントの 10%
低ベースライン リセット	50 に設定

図 6-5 に示す例では、差分カウントが、約 210 カウントで、閾値のパラメーターが以下のように設定されます。

指閾値=80%×210=168
ノイズ閾値=40%×210=84
負のノイズ閾値=ノイズ閾値 =84
ヒステリシス=10%×210=21
低ベースライン リセット=50

図 6-5.差分カウントの測定



6.3 近接センサーのチューニング

図 6-6 に近接センサーのチューニングの標準的な流れを示します。

CY8CMBR3xxx コントローラー内の近接センサーのチューニングは 2 段階があります。

1. SNR が 5:1 より大きいであると確認
2. 適切な閾値のパラメーターを設定

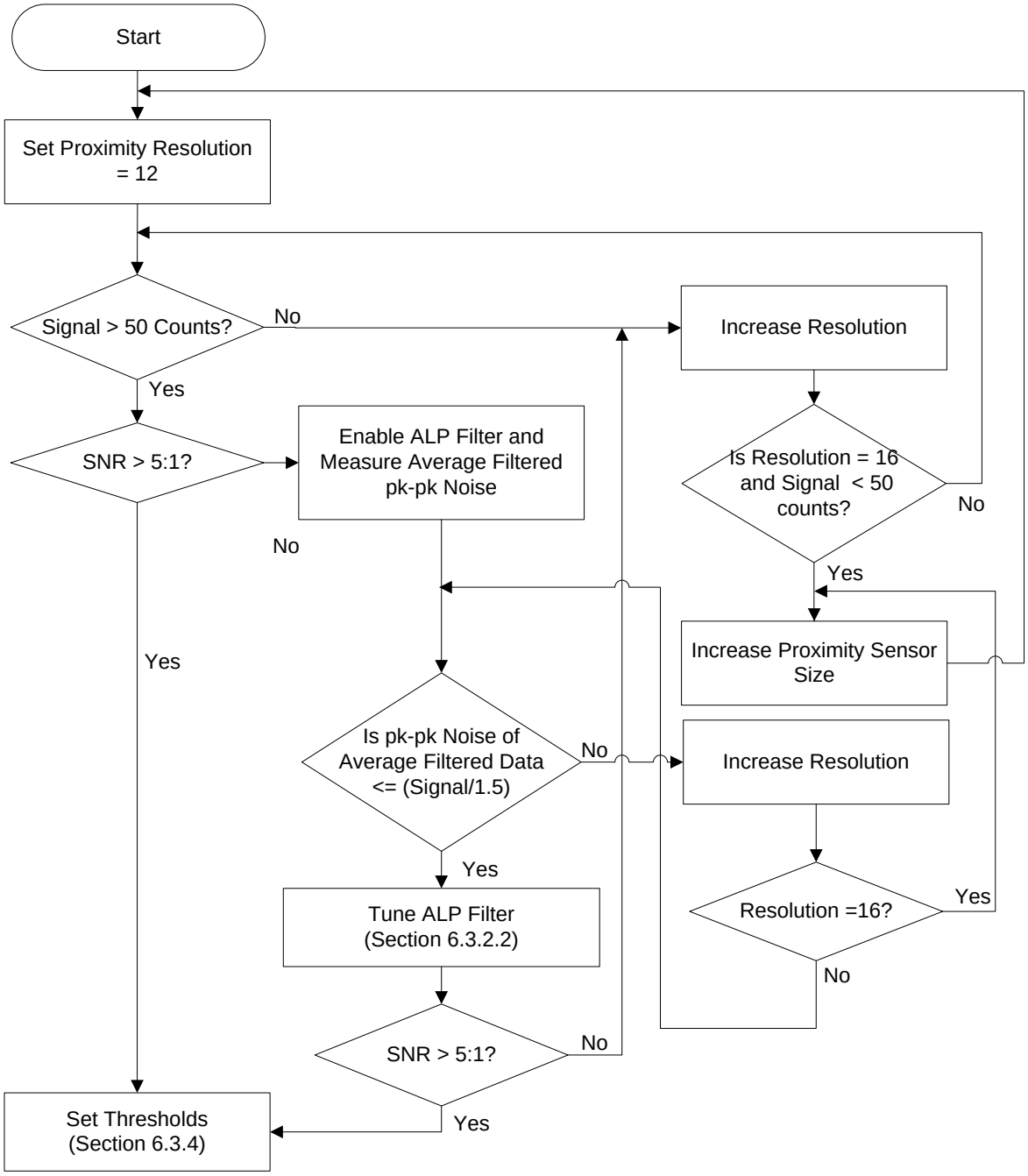
6.3.1 最小値の SNR の確保方法

SNR が 5:1 を超えることを確認するには、以下の手順に従ってください。

1. 近接センサーの分解能を 12 ビットにセットし、必要な近接距離での信号を測定します。
2. 信号が 50 カウントより小さい場合、近接の分解能を信号が 50 を超えるまで上げます。分解能を最高値 (16 ビット) に設定しても信号がまた 50 カウント以下の場合、近接センサーのサイズを増えます。
3. 図 6-7 に示すように、信号が 50 カウントを超えた後、ノイズ カウントを測定します。ノイズが信号の 5 分の 1 より大きい場合、ノイズを制限するために ALP フィルタをイネーブルします。

   注意: 近接センサーの場合、メジアン フィルタと IIR フィルタは適用されません。
4. SNR を計算し、SNR が 5:1 より大きいなら、閾値のパラメーターを設定します。
5. ALP フィルタを有効にしても SNR が 5:1 を超えない場合、分解能を上げて、SNR を計算します。分解能を最高値 (16 ビット) にセットしても SNR が 5:1 より小さい場合、SNR が 5:1 を超えるために近接センサーのサイズを増加します。

図 6-6.近接センサーのチューニング



6.3.2 高級 LOW パス (ALP) フィルタ

ALP フィルタが、近接センサーのノイズを減衰するために特別に設計される複数の LOW パス フィルタの組み合わせです。ノイズを最小限に減衰し、適切な応答時間を提供するために ALP フィルタは、複数の LOW パス フィルタの間で切り替えます。適切な動作のために表 6-3 に示される ALP パラメーターを設定します。

表 6-3.ALP フィルタ パラメーター

パラメーター	推奨値
K 値	ALP フィルタが有効になる時に設定
正の近接閾値	ALP フィルタが有効になる時に設定
負の近接閾値	ALP フィルタが有効になる時に設定

6.3.2.1 ALP フィルタ パラメーター

6.3.2.1.1 K 値

ALP フィルタの K 値は、近接センサーの Raw カウント内ノイズの減衰を決めます。CY8CMBR3xxx コントローラーは、1/16、1/32、および 1/64 の 3 つの異なった K 値を持っています。

ノイズ減衰は、1/64>1/32>1/16 という順序で低下します。「ALP フィルタのチューニング」の節を参照してこのパラメーターを設定してください。

6.3.2.1.2 正の近接閾値

このパラメーターは、近接センサーのターンオン時間を決めます。その範囲は 0～255 で、初期値が 30 です。「ALP フィルタのチューニング 」の節を参照してこのパラメーターを設定してください。

6.3.2.1.3 負の近接閾値

このパラメーターは、近接センサーのターンオフ時間を決めます。その範囲は 0～255 で、初期値が 30 です。「ALP フィルタのチューニング」の節を参照してこのパラメーターを設定してください。

6.3.2.2 ALP フィルタのチューニング

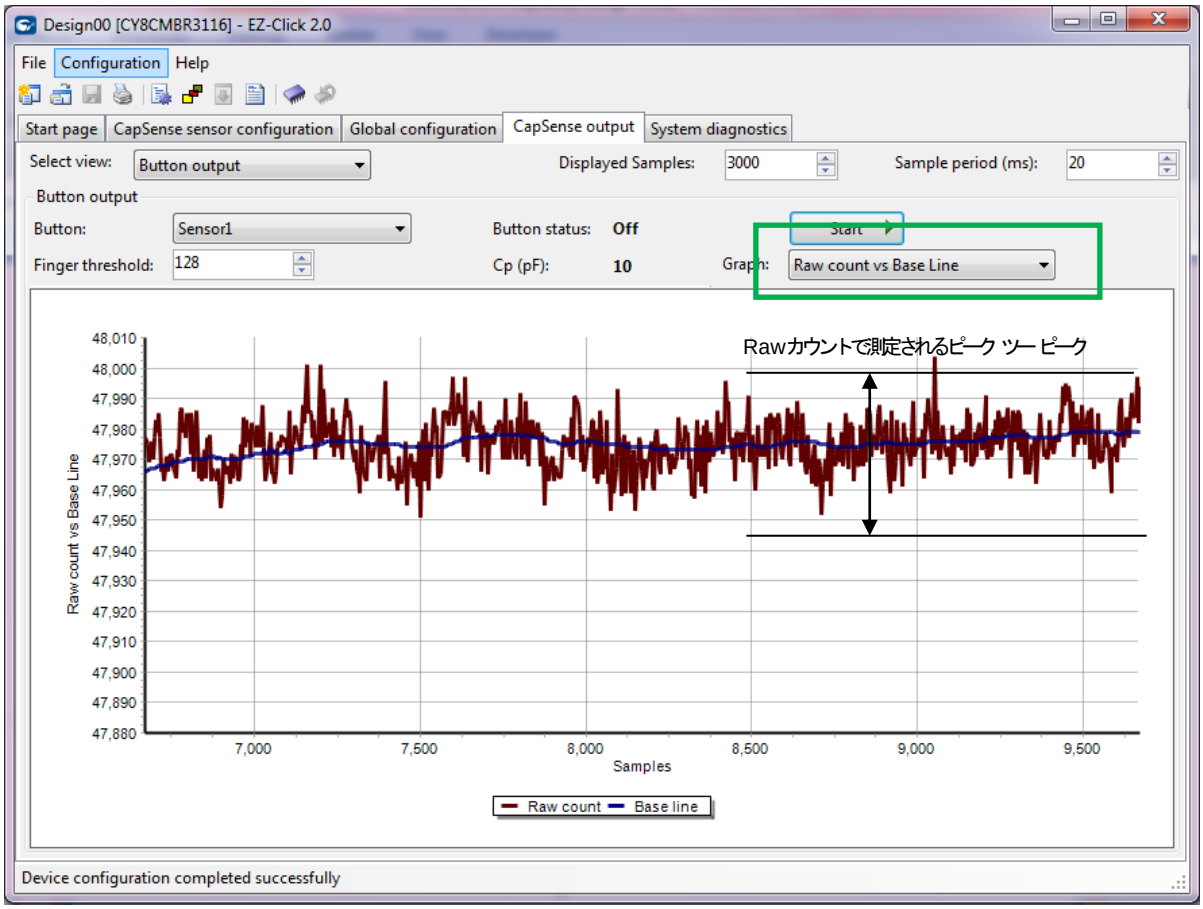
ALP フィルタを調整するには、以下の手順に従ってください。

1. 図 6-7 に示すように、ALP フィルタを無効にし、3000 サンプルにわたってピーク ツー ピーク ノイズを Raw カウントで測定します。詳細は、「ノイズ カウントの測定」を参照してください。
2. 表 6-4 に基づいて EZ-Click の「CapSense sensor configuration」タブで K 値を設定します。

表 6-4.K 値の選択

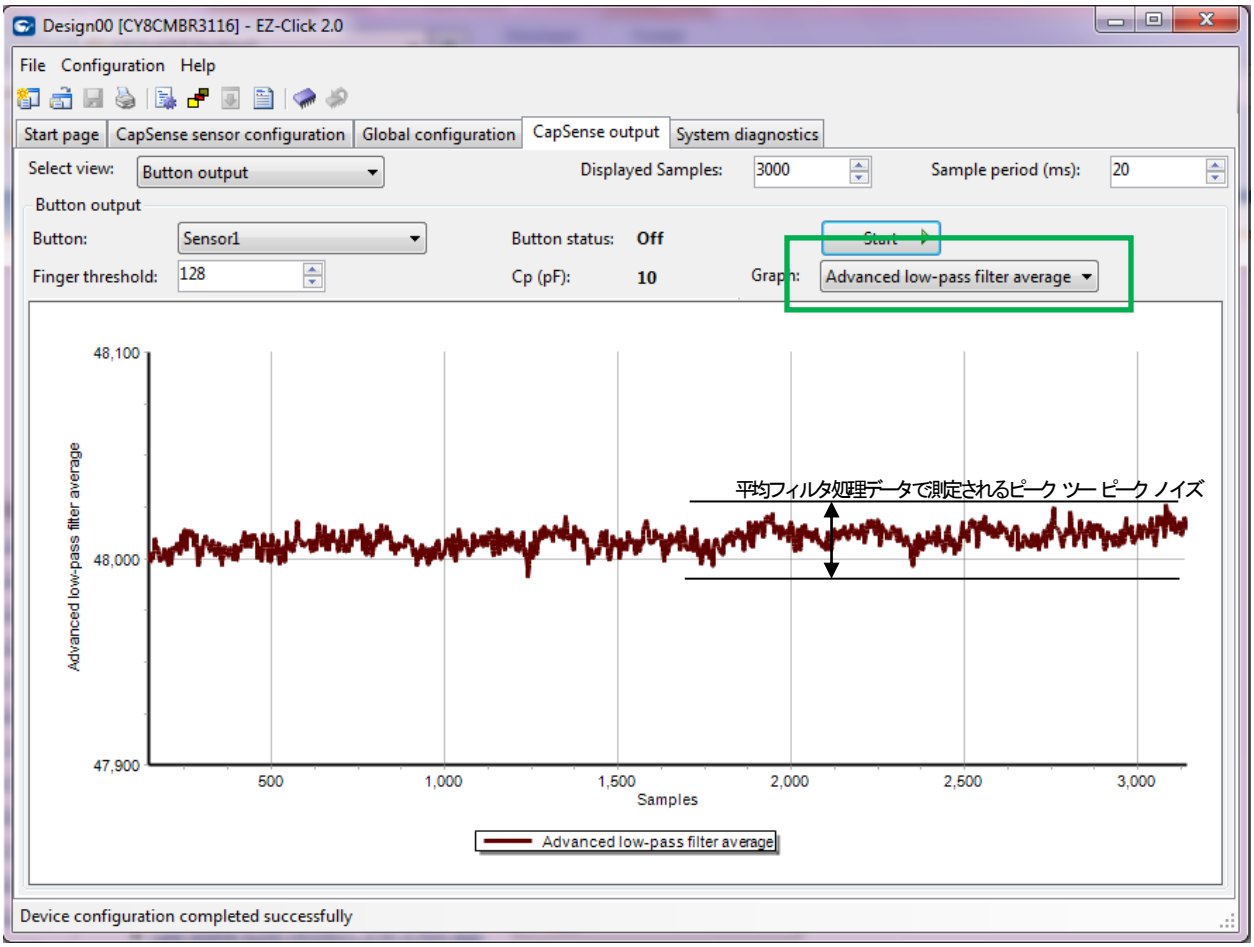
ピーク ツー ピーク ノイズ	推奨 K 値
32 カウント未満	1/16
32 カウントを超える	1/32
64 カウントを超える	1/64

図 6-7.ALP フィルタが無効にされる時の近接センサーのノイズ カウントの測定



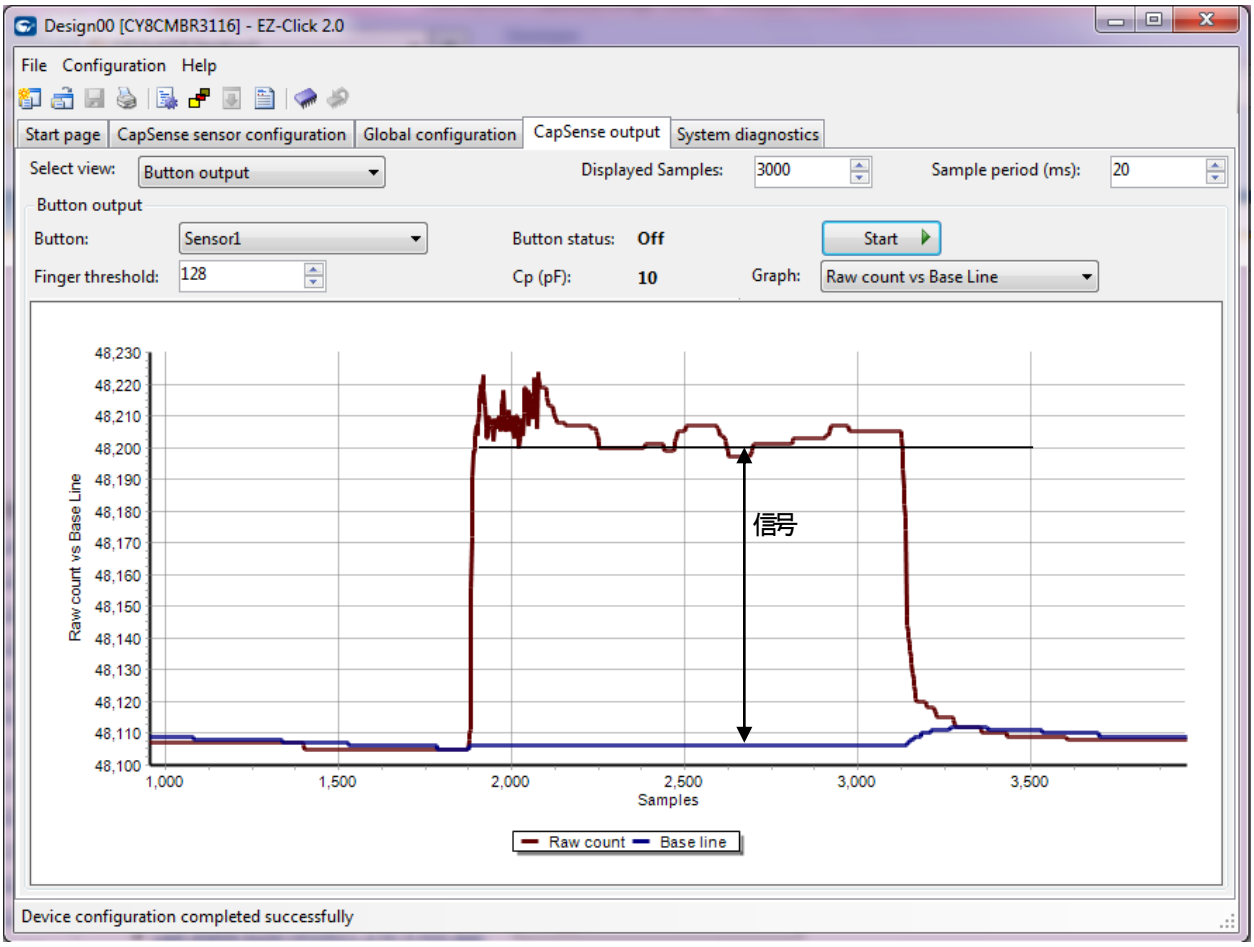
3. 図 6-8 に示すように、ALP フィルタを有効にし (K 値がセットされると、EZ-Click が ALP フィルタを自動的に有効にする) 、3000 サンプルのピーク ツー ピーク ノイズを高度な LOW パス平均フィルタ処理データで測定します。平均フィルタ処理データは、EZ-Click 内の「CapSense output」タブに表示できます。

図 6-8.平均フィルタ処理データで測定されるピーク ツー ピーク ノイズ



4. 正の近接閾値を「1.5×平均フィルタ処理データで測定されるピーク ツー ピーク ノイズ) 」に等しい値に設定します。
5. 負の近接閾値を「0.5×平均フィルタ処理データで測定されるピーク ツー ピーク ノイズ) 」に等しい値に設定します。
6. 指閾値を正の近接閾値に等しい値に設定します。これを行うには、ALP フィルタの応答を決めることが必要です。
7. デバイスをこれらの設定でコンフィギュレーションされ、raw カウントでノイズを測定します。
8. 図 6-9 で示すように、必要な近接距離で離れたところに手を置いて、信号を測定します。
9. SNR を計算します。SNR が 5:1 以上の場合、表 6-5 に示すように、閾値の設定を行います。そうでない場合、近接の分解能を増加し、ステップ 1 から繰り返します。

図 6-9. 近接センサー信号の測定



6.3.3 近接センサーの閾値パラメーター

表 6-5 に、近接センサーに適用できる閾値のパラメーターの一覧を示します。

表 6-5. 近接センサーの閾値

パラメーター	推奨値
近接閾値	常に手動で設定
近接接触閾値	常に手動で設定
近接ヒステリシス	ヒステリシスのオーバーライドが発生する時に設定
近接ノイズ閾値	ノイズ閾値が有効にされる時に設定
負の近接ノイズ閾値	負のノイズ閾値が有効にされる時に設定
近接低ベースライン リセット	低ベースライン リセットのオーバーライドが有効になる時に設定

6.3.3.1 近接閾値

近接閾値のパラメーターは、ボタンの指閾値のパラメーターと同じです。近接センサーのオン／オフ状態を決めます。差分カウントが「近接閾値+ヒステリシス」の値を超える場合、近接状態がオンです。差分カウントが「近接閾値+ヒステリシス」の値より小さい場合、近接状態がオフです。この閾値が、手動で設定すべきです。このパラメーターの範囲は 31～200 で、初期値が 128 です。推奨値は、表 6-6 を参照してください。

6.3.3.2 近接接触閾値

近接センサーは 2 つイベントがあります。それらは、近接検知イベントと (近接センサーに直接に触れる時の) 接触検知イベントです。近接閾値パラメーターは、近接イベントのオン／オフ状態を決めます。近接接触閾値が近接センサーの接触イベントを検知するために使用されます。このパラメーターの範囲は 62～65000 で、初期値が 512 です。推奨値は、表 6-6 を参照してください。

6.3.3.3 近接ヒステリシス

近接ヒステリシスのパラメーターはボタンのヒステリシスと同じです。また、近接センサーのオン／オフ状態を決めるために近接閾値パラメーターと共に使用されます。このパラメーターは 0～127 の範囲内にあり、初期値が 5 です。推奨値は、表 6-6 を参照してください。

6.3.3.4 近接ノイズ閾値

近接ノイズ閾値のパラメーターは、ボタンのノイズ閾値のパラメーターと同じです。このパラメーターの範囲は 0～27 で、初期値が 20 です。推奨値は、表 6-6 を参照してください。

6.3.3.5 負の近接ノイズ閾値

負の近接ノイズ閾値のパラメーターは、ボタンの 負のノイズ閾値のパラメーターと同じです。このパラメーターの範囲は 0～127 で、初期値が 20 です。推奨値は、表 6-6 を参照してください。

6.3.3.6 近接の低ベースライン リセット

近接低ベースライン リセットのパラメーターは、ボタンの低ベースライン リセットのパラメーターと同じです。このパラメーターの範囲は 0～127 で、初期値が 50 です。推奨値は、表 6-6 を参照してください。

6.3.4 近接センサー閾値パラメーターの設定

近接センサーでは、近接閾値と近接接触閾値のみを指定する必要があります。他のすべての閾値は SmartSense アルゴリズムにより内部で計算されます。

もし、何らかの理由で、これらのパラメーターを無効にする必要がある場合、Bridge Control Panel (ブリッジ コントロール パーネル) を使用して I²C コマンドを送信して、これらのパラメーターを無効にします。詳細については、CY8CMBR3xxx レジスタ TRM を参照してください。閾値のパラメーターを無効にしないことを強くお勧めします

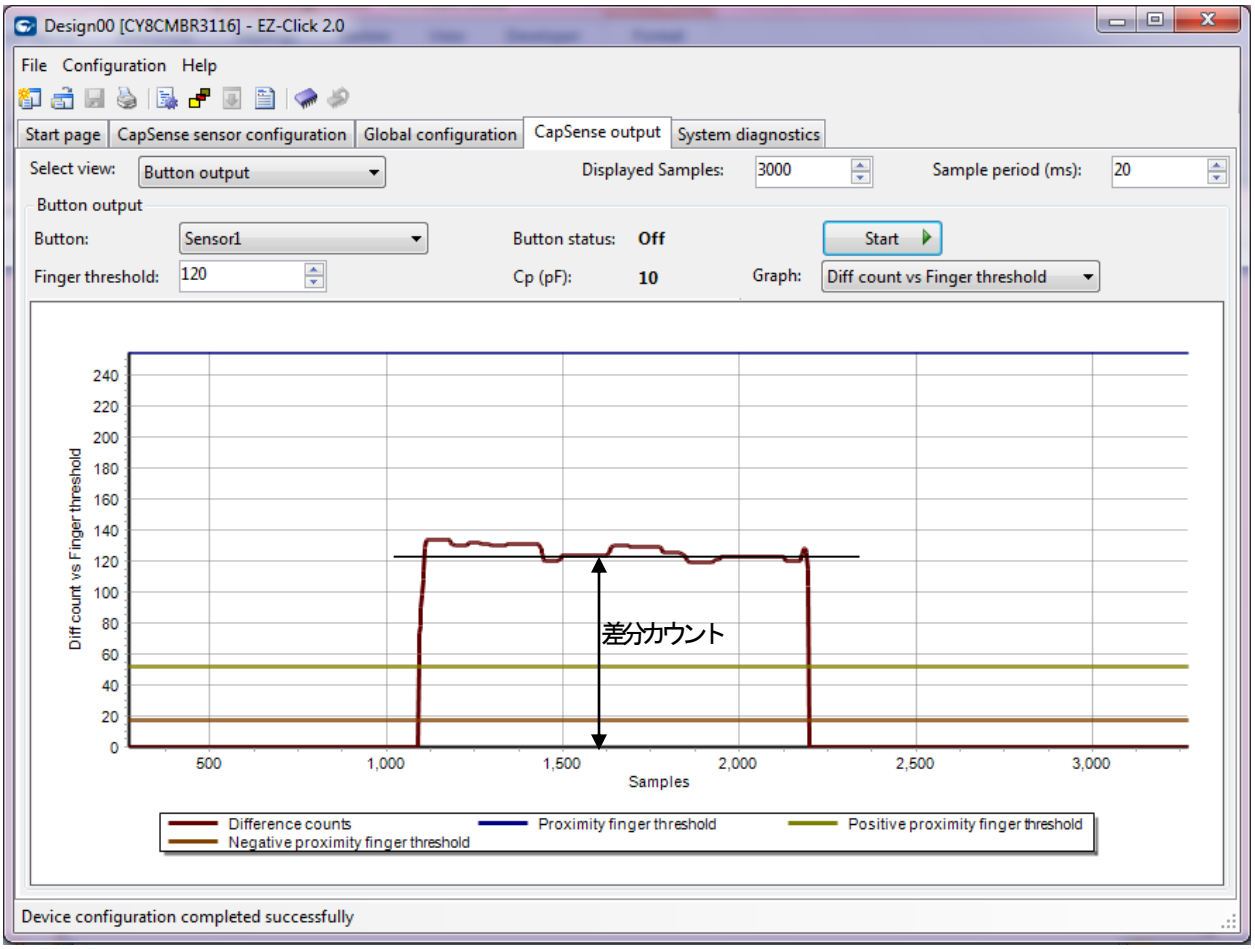
近接センサー用の閾値パラメーターを設定するには、以下の手順に従ってください。

1. CapSense 出力タブで、「Diff count vs. Finger threshold」グラフを近接センサー用に選択して、スタート ボタンをクリックします。
2. 図 6-10 で示すように、必要な近接距離で離れたところに手を置いて、信号を測定します。
   注:
   - CY8CMBR3xxx コントローラー の差分カウントは正規化されます。それは、「Raw カウント (接触時) -ベースライン」の結果に等しくありません。
   - 差分カウントが 255 より大きい場合、255 に切り捨てられます。この場合、閾値パラメーターを正確に設定できません。感度パラメーター値を差分カウントが 255 未満になるまで上げます。感度パラメーターの値を変更した後に、SNR が 5:1 を超えたことを確認します。
3. 閾値のパラメーターを表 6-6 に示す値に設定します。
4. 近接接触閾値を設定するために、近接センサーに接触し差分カウントを測定します (近接センサー接触時) 。
5. 近接接触閾値を差分カウントの 80%に相当する値に設定します (近接センサー接触時) 。

表 6-6.近接閾値の設定

閾値パラメーター	推奨値
近接閾値	差分カウントの 80%
近接接触閾値	差分カウントの 80% (近接センサーが接触される時)
近接ノイズ閾値	差分カウントの 40%
負の近接ノイズ閾値	差分カウントの 40%
近接ヒステリシス	差分カウントの 10%
近接の低ベースライン リセット	50 に設定

図 6-10.近接センサーの閾値をセットするための差分カウント測定



7. 低消費電力設計上の考慮事項

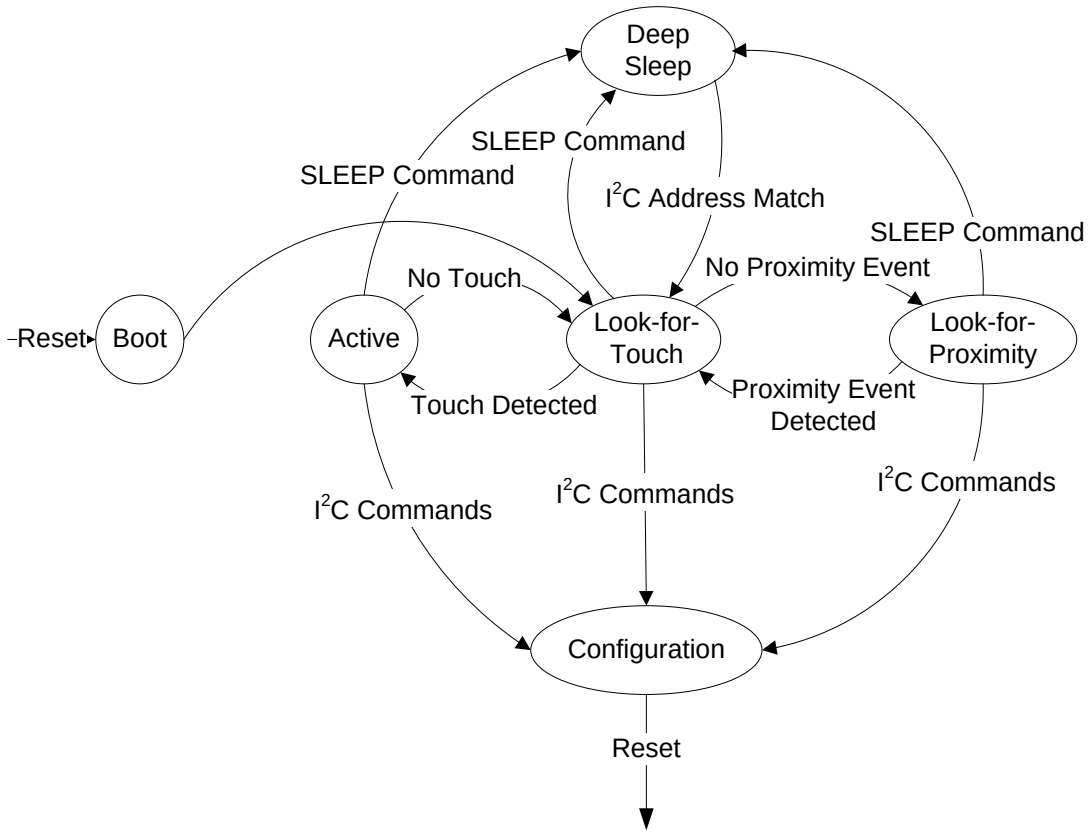
容量性接触センシングは、蓄電池式でハンドヘルドの携帯用電子機器で使用されます。これらのデバイスが蓄電池式であるため、消費電力とエネルギー効率は重要になります。この節で、CY8CMBR3xxx コントローラーの消費電力に影響を及ぼす要素について説明し、それに、消費電力を最小限にする方法も示します。

7.1 CY8CMBR3xxx の動作モード

CY8CMBR3xxx コントローラーは、5 つの動作モードをサポートします。

1. アクティブ モード
2. Look-for-Touch モード
3. Look-for-Proximity モード
4. ディープ スリープ モード
5. 設定モード

図 7-1.CY8CMBR3xxx の動作モード



7.1.1 アクティブ モード

CY8CMBR3xxx コントローラーは、図 7-1 で示すようにリセット状態からアクティブ モードに入ります。アクティブ モードでは、コントローラーはすべてのセンサーをスキャンし、センサー データを更新し、そしてセンサーの状態に応じてすべての設定した出力 (GPO、ブザー、およびホスト割り込み) を駆動します。初期設定では、アクティブ モード中に、高速の接触応答の確保のため、スキャン速度が 20ms に設定されます。センサーのスキャンとセンサー データの処理が 20ms 内で完成される場合、コントローラーを「20ms–アクティブ時間」の期間でスリープ モードに移行させます。図 7-2 に示すように、20ms タイマの期限が切れると、センサーは繰り返しスキャンされ、スキャン サイクルは引き続きます。

すべてのセンサーのスキャン期間とセンサー データの処理期間が 20ms を超える場合、デバイスはスリープ モードには移行しません。その代わりに、図 7-3 に示すように、次のスキャン サイクルが開始されます。

図 7-2.アクティブ時間が 20ms 未満の時のアクティブ モードでのスキャン／スリープのシーケンス

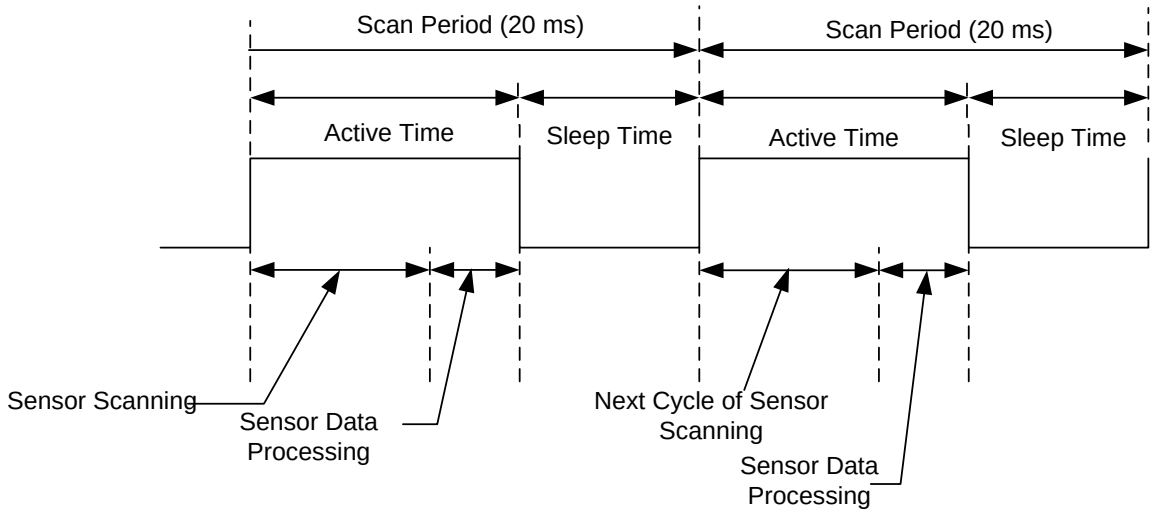
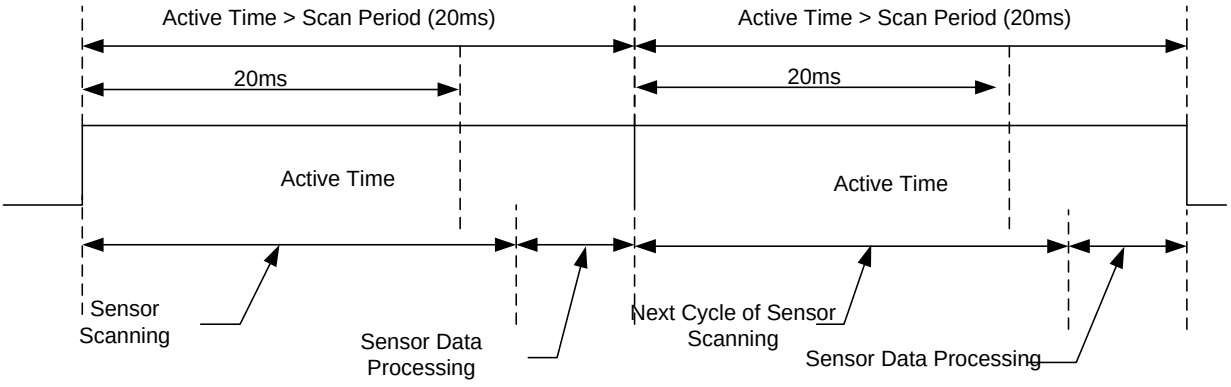


図 7-3.アクティブ時間が 20ms を超える時のアクティブ モードでのスキャン／スリープのシーケンス



7.1.2 Look-for-Touch モード

アクティブ モード中に、状態タイムアウト値に相当する期間内に接触が検知されない場合、CY8CMBR3xxx は Look-for-Touch モードに移行します (図 7-1 を参照してください) 。スキャン期間が設定可能なことを除いて、Look-for-Touch モードはアクティブ モードと同様です。スキャン期間の初期値が 120ms です。このモードでは、すべてのセンサーがスキャンされ、センサー データが処理されます。いずれかのセンサーがオンになると、出力 (GPO、ブザーおよびホスト割り込み) は駆動され、コントローラーはアクティブ モードに入ります。

7.1.3 Look-for-Proximity モード

設計に近接センサーがあり、ウェイク オン アプローチ (近づくとオンになる) 機能がイネーブルされる場合、Look-for-Proximity モードは適用可能です。Look-for-Touch モード中に、状態タイムアウト値以上の期間内に接触が検知されない場合、コントローラーは Look-for-Proximity モードに移行します (図 7-1.を参照してください) 。

Look-for-Proximity モードでは、コントローラーは近接センサーのみスキャンします。近接イベントが検知されると、コントローラーは Look-for-Touch モードに移行します。Look-for-Proximity モードで接触イベントが検知されると、コントローラーはアクティブモードに移行します。このモード内のスキャン時間は設定可能で、Look-for-Touch モードのスキャン期間に等しく、初期値が 120ms です。

7.1.4 ディープスリープ モード

ディープスリープ モードは最低の低消費電力モードです。ホストがスリープのコマンドを発行すると、CY8CMBR3xxx デバイスはディープスリープ モードに移行します。詳細は、CY8CMBR3xxx レジスタ TRM に示す CTRL_CMD レジスタに関する説明を参照してください。ディープスリープ モードでは、デバイスはセンサーをスキャンせず、出力は無効にされます。I²C アドレスが一致する時に、デバイスはウェイクアップします。最初の I²C コマンドを受け入れなく、Look-for-Touch モードに移行します。

7.1.5 設定モード

設定データが不揮発性メモリに保存されるように CMD_OP_CODE = 「2」を CTRL_CMD レジスタに書き込む時、デバイスは設定モードに移行します。詳細については、CY8CMBR3xxx レジスタ TRM を参照してください。デバイスのリセット (ソフトウェア リセットかハードウェア リセット) 時のみにデバイスは設定モードを終了することができます。

7.2 消費電力に影響する要因

CY8CMBR3xxx コントローラーの消費電力に影響を与える要因は以下です。

- **センサー数**:センサーの数が多ければ多いほどセンサーの合計スキャン時間は多くなります。固定したスキャン期間に対して、スキャン時間が増えると、デバイスがスリープ モードにある期間は少なくなります。これにより、デバイスの消費電力が増加されます。
- センサーの **C_P:** センサーのスキャン時間に影響を与えます。C_P が高ければ高いほど、センサーのスキャン時間も多くなります。よって、消費電力が増えます。センサーの C_P を最小限にするために、Getting Started with CapSense ガイドに説明している設計ガイドラインに従ってください。
- **感度**: このパラメーターはセンサーのスキャン時間に影響を及ぼします。センサー感度のパラメーター値が上がると、センサーのスキャン時間は増えます。その結果として、感度のパラメーター値が小さければ小さいほど、消費電力は多いです。

  消費電力は 100fF>200fF>300fF>400fF の順序で減少します。
- **電磁環境適合性 (EMC)**: EMC 機能が有効にする時、センサーのスキャン時間が増えることに従って消費電力も多くなります。
- **ブザー期間**:ブザー出力している間、CY8CMBR3xxx コントローラーはアクティブ状態にあります。ブザー期間が長ければ長いほど、アクティブ時間は長くなって、消費電力も多くなります。
- **GPO の PWM**:GPO 上の PWM を有効にすることは、ブザー出力のイネーブルがもたらす影響と同じ影響を与えます。PWM 出力が GPO ピンで有効にされると、デバイスはスリープ モードに移行しません。よって、消費電力が増加します。
- **スキャン期間:** スキャン期間のパラメーターは、デバイスがスリープ モードにある期間に影響を与えます。スキャン期間が多ければ多いほど、消費電力は少なくなります。
- **状態タイムアウト**:状態タイムアウトのパラメータがデバイスのアクティブ モードから look-for-touch モードへの移行速度、または look-for-touch モードから look-for-proximity モードへの移行速度に影響を与えます。状態タイムアウトの間隔が短ければ短いほど消費電力は少なくなります。
- **I²C 通信:** アクティブ モードの節で説明したように、センサーのスキャン処理が完成した後に CY8CMBR3xxx はスリープ モードに移行します。ホスト プロセッサとの I²C トランザクションがあったら、CY8CMBR3xxx はスリープ モードに移行しません。また、ディープスリープ モードにある場合、このモードを終了します。そのため、よく行われる I²C トランザクションによって、平均の消費電力が増加します。

平均の消費電力を減少するために、専用のホスト割り込みピンを実装し、CY8CMBR3xxx がホスト割り込みパルスを送信する時のみに、I²C トランザクションを開始します。

平均電流の見積もりの節に説明している設計入力に基づいて、CY8CMBR3xxx デザイン ツールボックスを使用してコントローラーの平均の電流消費を見積もります。

7.3 低消費電力向けのシステム設計推奨項目

サイプレスの CY8CMBR3xxx は電池式のアプリケーションの低電力要件に対応しています。消費電力を最小化するには、以下の手順を行います。

- Getting Started with CapSense および CY8CMBR3xxx デザイン ツールボックス で説明しているガイドラインに従ってセンサー C_P を最小限にします。
- センサーの感度を適切な値に設定します。感度制御の節を参照してください。
- 必要な場合のみ EMC をイネーブルします。電磁環境適合性 (EMC) 節を参照してください。
- スキャン期間を増やし、状態タイムアウト期間を減少します。
- センサーのスキャン処理が不要になった時に、デバイスをディープスリープ モードに移行させる、またはパワー マネージャ IC (PMIC) を使ってデバイスへの電源を切ります。
- CY8CMBR3xxx コントローラーへの頻繁な I²C 読み込み／書き込みを回避します。CY8CMBR3xxx がホスト割り込みパルスを送信した後のみにホスト割り込みピンを実装し、²C トランザクションを開始します。
- 近接センサーを実行して、ウェーク オン アプローチ (近づいたときにオンになる) 機能をイネーブルします。
- ブザーON 時間の期間を減少します。
- 必要な場合のみ GPO 上の PWM をイネーブルします。

8. リソース

8.1 ウェブサイト

サイプレスの CapSense コントローラーウェブサイトを訪問すれば、本書で説明したすべての参照資料にアクセスできます。

CY8CMBR3xxx ウェブページで、様々な技術的リソースを見つけてください。

8.2 データシート

CapSense CY8CMBR3xxx デバイスのデータシートは、CY8CMBR3xxx データシードを参照してください。

8.3 レジスタ技術リファレンス マニュアル

レジスタ技術リファレンス マニュアルは、CY8CMBR3xxx レジスタ TRM に掲載されています。

8.4 Design Toolbox (デザイン ツールボックス)

対話式のデザイン ツールボックス により、ユーザーは強固で信頼できる CY8CMBR3xxx CapSense ソリューションを設計することができます。

8.5 EZ-Click™ 2.0

EZ-Click 2.0は、CY8CMBR3xxxデバイスの設定をカスタマイズするのに使用する簡単なGUIベースのソフトウェア ユーティリティです。

8.6 開発キット

CY3280-MBR3評価キット は、CY8CMBR3116 CapSenseコントローラー機能の紹介を目的としています。CY8CMBR3xxxソリューションの機能を迅速に評価するためにこのキットを使用します。

8.7 デザイン サポート

サイプレスには様々なデザイン サポート チャネルがあり、お客様の CapSense ソリューションの成功を確実にしています。

- 知識ベース記事 – 製品ファミリ別の技術情報記事を閲覧したり、CapSense についての様々なトピックスを検索できます。
- CapSense アプリケーション ノート: 本書で紹介した情報に基づいた幅広いアプリケーション ノートを詳細に調べます。
- ホワイト ペーパー: 静電容量タッチ インターフェースの高度なトピックについて学びます。
- サイプレス開発コミュニティ: サイプレス技術コミュニティに参加し、情報交換できます。
- CapSense 製品選択ガイド: すべての CapSense 製品ラインを見ることができます。
- ビデオ ライブラリ: チュートリアル ビデオで素早く学習できます。
- 品質と信頼性: サイプレスはお客様に完全にご満足いただけるよう努力しています。当社の品質ウェブサイトで、信頼性および製品認定報告書をご覧ください。
- 技術サポート: 一流の技術サポートをオンラインで受けられます。

9. 付録

9.1 高レベル API

表 9-1. 高レベルのホスト API

1	**CY8CMBR3xxx_ReadSensorStatus**	
	説明	この API は、ボタンの状態、ラッチされたボタンの状態、近接状態、ラッチされた近接状態、スライダー1 と 2 の位置、CY8CMBR3xxx デバイスからのスライダー1 と 2 のリフトオフ位置を読み出す スライダーの位置とスライダーリフトオフ位置は、CY8CMBR3106S デバイスでのみ有効 対応する状態の変化を理解するために、デバイスはホスト割り込みパルスをアサートするこの API を使用することが可能
	試作	bool CY8CMBR3xxx_ReadSensorStatus(uint8 slaveAddress, CY8CMBR3XXX_SENSORSTATUS *status)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119
		*status CY8CMBR3xxx デバイスから読み込むセンサーの状態が保存されているユーザー定義のバッファへのポインタ。バッファ構造は CY8CMBR3xxx_APIs.h.内に定義
	戻り値	TRUE – バッファが正しく更新された場合 FALSE – バッファが更新されていない場合
2	**CY8CMBR3xxx_WriteData**	
	説明	この API は、CY8CMBR3xxx デバイス内の連続するレジスタ位置にデータの 1 つまたは複数のバイトを書き込みます。最初のレジスタ位置は、ユーザーに指定される
	試作	bool CY8CMBR3xxx_WriteData(uint8 slaveAddress, uint8 *writeBuffer, uint8 numberOfBytes)
	パラメーター	SlaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119

		*writeBuffer そこからデータがデバイスに書き込まれるバッファ 最初の要素には、書き込むためのデバイスのレジスタの場所を常に含める必要がある。この値は 0～134 のいずれかである 連続する各要素は、そのレジスタと連続したレジスタに書き込まれるデータを含む必要がある。これらの要素は、0～255 の範囲の値を持てます。データ バイト数は、1 と (128 と 135 – レジスタ位置) の最小値の間である
		numberOfBytes 書き込まれるバイト数は、バッファ内の要素の数に等しい 値の範囲: 1 – 最小(129, 136 – レジスタ位置)
	戻り値	TRUE – 書き込みが正常であった場合 FALSE – 書き込みが正常でなかった場合
3	**CY8CMBR3xxx_ReadData**	
	説明	この API は、CY8CMBR3xxx デバイス内の連続するレジスタ位置にデータの 1 つまたは複数のバイトを読み出す。最初のレジスタ位置は、ユーザーに指定される
	試作	bool CY8CMBR3xxx_ReadData(uint8 slaveAddress, uint8 registerAddress, uint8 *readBuffer, uint8 numberOfBytes)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119
		registerAddress 以下から読み出されるレジスタ位置この引数に渡された値は、ファイル CY8CMBR3xxx_Registers.h 内で定義されたレジスタ オフセットのマクロのいずれかである CY8CMBR3xxx_DEVICE マクロで定義されているデバイスによってサポートされている場合のみ、レジスタ用のマクロが定義される
		*readBuffer デバイスから読み出されたデータで更新されるバッファ 連続したレジスタからのデータ読み出しを含む連続する各要素これらの要素は、0～255 のいずれかの値である
		numberOfBytes 読み出されるバイト数は、バッファ内の要素の数に等しい 有効範囲: 1～252
	戻り値	TRUE – 読み出しが正常であった場合 FALSE – 読み出しが正常でなかった場合

4	**CY8CMBR3xxx_WriteDualByte**	
	説明	この API は CY8CMBR3xxx デバイス内の 2 バイトのレジスタに書き込みます。レジスタ位置はユーザーに指定される
	試作	bool CY8CMBR3xxx_WriteDualByte(uint8 slaveAddress, uint8 registerAddress, uint16 writeData)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119
		registerAddress 以下から書き込まれるレジスタ位置この引数に渡された値は、ファイル CY8CMBR3xxx_Registers.h 内で定義されたレジスタ オフセットのマクロのいずれかである CY8CMBR3xxx_DEVICE マクロで定義されているデバイスによってサポートされている場合のみ、レジスタ用のマクロが定義される
		writeData 2 バイトの値をレジスタに書き込みます。LSB が最初のバイトを形成し、MSB は 2 番目のバイトを形成する 有効範囲: 0～65535
	戻り値	TRUE – 書き込みが正常であった場合 FALSE – 書き込みが正常でなかった場合
5	**CY8CMBR3xxx_ReadDualByte**	
	説明	この API は、CY8CMBR3xxx デバイス内の 2 バイトのレジスタから読み出し、読み出したバッファに連結された値を更新します。レジスタ位置はユーザーに指定される
	試作	bool CY8CMBR3xxx_ReadDualByte(uint8 slaveAddress, uint8 registerAddress, uint16 *readData)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119
		registerAddress 以下から読み出される 2 バイトのレジスタデータはこのアドレスとその次のアドレスから読み出される この引数に渡された値は、ファイル CY8CMBR3xxx_Registers.h 内で定義されたレジスタ オフセットのマクロのいずれかでなければなりません。CY8CMBR3xxx_DEVICE マクロで定義されているデバイスによってサポートされている場合のみ、レジスタ用のマクロが定義される
		*readData デバイスから読み出された値で更新される 2 バイトのバッファ API は、デバイスからの最初のバイト読み出しで LSB を埋め、2 番目のバイトで MSB を埋めます。値は、0～65535 である

	戻り値	TRUE – 読み出しが正常であった場合 FALSE – 読み出しが正常でなかった場合
6	**CY8CMBR3xxx_SendCommand**	
	説明	この API は、CY8CMBR3xxx デバイスに新しいコマンドを発行します。この API を呼び出す前に、デバイスが CY8CMBR3xxx_CheckCommandStatus () を呼び出して新しいコマンドを受け入れる準備ができているかを確認する必要がある
	試作	bool CY8CMBR3xxx_SendCommand(uint8 slaveAddress, uint8 command)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119
		コマンド コマンドのオペコードが発行この引数に渡された値は、ファイル CY8CMBR3xxx_CommandsAndConfig.h 内で定義されたオペコードのマクロのいずれかである
	戻り値	TRUE – コマンドが正常に発行された場合 FALSE – コマンドが正常に発行されなかった場合
7	**CY8CMBR3xxx_CheckCommandStatus**	
	説明	この API は、CY8CMBR3xxx デバイスが新しいコマンドを受け入れる準備ができているかどうか、そして以前に発行されたコマンドが正常に実行されたかを検証する
	試作	uint8 CY8CMRB3xxx_CheckCommandStatus(uint8 slaveAddress, uint8 *errorCode)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119
		*errorCode 最後のコマンドの実行からされたエラーコードへのポインタ最後のコマンドが失敗した場合、API は失敗の原因を示すためにこのデータを更新する。有効範囲は 0～255 である
	戻り値	0 – 最後のコマンドがまだ実行されている場合 1 – 最後のコマンドが正常に発行された場合 2 – 最後のコマンドが正常に実行した場合、または実行しなかった場合、I^2C 通信に失敗した

8	**CY8CMBR3xxx_Configure**	
	説明	この API は、ユーザー指定のコンフィギュレーションを CY8CMBR3xxx デバイスに書き込み、コンフィギュレーションを保存し、デバイスをリセット デバイスをコンフィギュレーションするには、まず、EZ-Click を使用してコンフィギュレーションを作成する必要があります。128 バイトのコンフィギュレーション データ (EZ-Click プロジェクト内のプロジェクト固有の.h ファイル内に生成) は、この API に応じて直接使用する必要がある この API は、コンフィギュレーションの保存コマンド実行中と CY8CMBR3xxx デバイスのリセット中に CPU の実行をブロックする
	試作	bool CY8CMBR3xxx_Configure(uint8 slaveAddress, const unsigned char *configuration)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119
		*コンフィギュレーション デバイスの全コンフィギュレーションを保存する 128 バイトのバッファこのデータは、EZ-Click 生成されたコンフィギュレーション データから直接取得する必要があります。これは 126 バイトのコンフィギュレーション レジスタのデータと 2 バイトの CRC が含まれている
	戻り値	TRUE – デバイスが正常にコンフィギュレーションされた場合 FALSE – デバイスが正常にコンフィギュレーションされなかった場合
9	**CY8CMBR3xxx_CalculateCrc**	
	説明	この API は、CY8CMBR3xxx デバイス用にコンフィギュレーションする特定のコンフィギュレーショの CRC チェックサムを計算
	試作	uint16 CY8CMBR3xxx_CalculateCrc(uint8 *configuration)
	パラメーター	*コンフィギュレーション この 126 バイトのバッファは、デバイス用にコンフィギュレーションする全コンフィギュレーションを保持
	戻り値	計算された CRC 用の 2 バイトの値 (MSB である上位バイト)
10	**CY8CMBR3xxx_VerifyDeviceOnBus**	
	説明	この API は、I^2C バス上のデバイスが有効な CY8CMBR3xxx デバイスであることを検証します。これは、デバイスを確認するために、デバイス ID、ファミリーID、およびデバイスのリビジョンをチェック
	試作	bool CY8CMBR3xxx_VerifyDeviceOnBus(uint8 slaveAddress)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119

	戻り値	TRUE – デバイスが有効である場合 FALSE – デバイスが有効でない場合
11	**CY8CMBR3xxx_SetDebugDataSensorId**	
	説明	この API は、CY8CMBR3xxx デバイスから対応するデバッグ データを読み出すためのセンサー ID をコンフィギュレーションする デバイスからデバッグ データを読み出すためには、まず、この API を呼び出し、次にデバイスのリフレッシュ間隔に等しい待ち時間を持つ必要があります。次に、デバイスからのデバッグ データをリード バックするために CY8CMBR3xxx_ReadSensorDebugData () を呼び出す必要がある
	試作	bool CY8CMBR3xxx_SetDebugDataSensorId(uint8 slaveAddress, uint8 sensorId)
	パラメーター	SlaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119
		sensorId データが要求されているセンサー番号。この番号は、デバイスによってサポートされているセンサーの合計数より 1 つ少ない最大の数である
	戻り値	TRUE – センサーID 書き込み動作が正常な場合 FALSE – センサーID 書き込み動作が正常でなかった場合
12	**CY8CMBR3xxx_ReadSensorDebugData**	
	説明	この API は、C_P、差のカウント、ベースライン、Raw のカウント、およびセンサーの平均 Raw のカウントを読み出し、バッファにデータを更新する デバッグ データを読み出すためには、まず、CY8CMBR3xxx_SetDebugDataSensorId を () 呼び出して、その後で 1 回のリフレッシュ間隔を待つ必要があります。この時間後に、デバイスからデータを読み出すために、この API を呼び出す必要がある API は、デバイスの同期カウンタがマッチした場合にのみ、バッファを更新し、そうでない場合、API はデータを再度読み出そうとします。読み出し試行の最大回数は、CY8CMBR3xxx_SYNC_COUNTER_MATCH_RETRY で定義される。CY8CMBR3xxx_APIs.h ファイル内でこの値をコンフィギュレーションできる
	試作	bool CY8CMBR3xxx_ReadSensorDebugData(uint8 slaveAddress, CY8CMBR3XXX_SENSORDATA *debugData)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I^2C アドレス 有効範囲: 8～119
		*debugData CY8CMBR3xxx デバイスから読み込出すデータが保存されているユーザー定義のバッファへのポインタ。バッファ構造は CY8CMBR3xxx_APIs.h.内に定義される

	戻り値	TRUE – バッファが正しく更新された場合 FALSE – バッファが更新されていなかった場合
13	**CY8CMBR3xxx_ReadDiffCounts**	
	説明	この API は、CY8CMBR3xxx デバイスの全センサーの差のカウントを読み出し、バッファにデータを更新する API は、デバイスの同期カウンタがマッチした場合にのみ、バッファを更新します。そうでない場合には、API はデータを再度読み出そうとします。読み出し試行の最大回数は、CY8CMBR3xxx_SYNC_COUNTER_MATCH_RETRY で定義される。CY8CMBR3xxx_APIs.h ファイル内でこの値をコンフィギュレーションできる
	試作	bool CY8CMBR3xxx_ReadDiffCounts(uint8 slaveAddress, uin16 *differenceCounts)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I²C アドレス 有効範囲: 8～119
		*differenceCounts すべてのセンサーの差分カウントを格納するバッファバッファサイズは、デバイス内のセンサーの数に等しいでなければならない
	戻り値	TRUE – バッファが正しく更新された場合 FALSE – バッファが更新されていなかった場合

9.2 低レベル API

表 9-2. 低レベルのホスト API

1	**Host_LOWLevelWrite**	
	説明	この API は、I²C 通信プロトコルを使用して CY8CMBR3xxx デバイスのレジスタ マップに書き込みます。実施はホスト プロセッサに依存し、特定のホスト プロセッサと合わせて API コードを変更する必要がある デバイスが I²C のトランザクションを初めて NACK 応答するケースでは、この API が、次の 340 ミリ秒以内に 2 回再試行することを確認
	試作	bool Host_LOWLevelWrite(uint8 slaveAddress, uint8 *writeBuffer, uint8 numberOfBytes)
	パラメーター	SlaveAddress CY8CMBR3xxx デバイスの I²C アドレス 有効範囲: 8～119

		*writeBuffer そこからデータがデバイスに書き込まれるバッファ 最初の要素には、書き込むためのデバイスのレジスタの場所を常に含める必要がある。この値は 0～251 のいずれかである 連続する各要素は、そのレジスタと連続したレジスタに書き込まれるデータを含む必要がある。これらの要素は、0～255 の範囲の値を持てます。データ バイト数は、0 と 128 の間とすることができる。ゼロ バイトのデータを書くことは、デバイス レジスタ マップ情報を読み出すためのレジスタ位置を更新していることを示す
		numberOfBytes 書き込まれるバイト数は、バッファ内の要素の数に等しい (すなわち、レジスタ位置に対しデータ バイトの数に 1 を加算)
	戻り値	TRUE – 書き込み処理が正常であった場合 FALSE – 書き込み処理が正常でなかった場合
2	**Host_LOWLevelRead**	
	説明	この API は、I²C 通信プロトコルを使用して CY8CMBR3xxx デバイスのレジスタ マップから読み出す。実施はホスト プロセッサに依存し、特定のホスト プロセッサに合わせて API コードの変更が必要 デバイスが I2C のトランザクションを初めて NACK 応答するケースでは、この API が、次の 340 ミリ秒以内に 2 回再試行することを確認
	試作	bool Host_LOWLevelRead(uint8 slaveAddress, uint8 *readBuffer, uint8 numberOfBytes)
	パラメーター	slaveAddress CY8CMBR3xxx デバイスの I²C アドレス 有効範囲: 8～119
		*readBuffer デバイスから読み出されたデータで更新されるバッファ 読み出すレジスタ位置は、この API を呼び出す前に Host_LOWLevelWrite API を使用してコンフィギュレーションする必要がある。そのレジスタと連続したレジスタから読み出されたデータが含まれている連続する各要素これらの要素は、0～255 のいずれかの値を持つ
		numberOfBytes バッファ サイズに等しく、読み出されるデータバイト数 有効範囲: 1～252
	戻り値	TRUE – 読み出し処理が正常であった場合 FALSE – 読み出し処理が正常でなかった場合
3	**Host_LOWLevelDelay**	
	説明	この API は、高レベルの API によって使用される時間遅延機能を実現します。遅延周期はミリ秒単位です。この遅延は、必要量の時間に応じてコードが実行されるブロックで成される 実施はホスト プロセッサに依存し、ユーザーのホストと合わせて API コードを変更する必要がある

	試作	void Host_LOWLevelDelay(uint16 milliseconds)
	パラメーター	ミリ秒 待機が必要とされる時間はミリ秒単位である 有効範囲: 0～65535
	戻り値	なし

用語集

AMUXBUS
入出力ピンを複数の内部アナログ信号に接続する PSoC 内にあるアナログ マルチプレクサ バスです。

SmartSense™ 自動チューニング
設計フェーズの後で最適性能のために、センス パラメーターを自動的にセットし、システム、製造および環境変化に対し連続的に補正する CapSense アルゴリズムです。

ベースライン
センサーに人の指で触らない時の raw カウントの傾向を推定しファームウェア アルゴリズムから得られる値です。ベースラインは raw カウントの突然の変化に敏感性が低く、差分カウントを計算するためのリファレンス点を提供します。

ボタンまたはボタン ウィジェット
関連したセンサーを持って、センサーのアクティブ状態または非アクティブ状態 (すなわち、2 つだけの状態) を報告するウィジェットです。例えば、センサー上の指の「タッチ有り」または「タッチ無し」の状態を検出できます。

差分カウント
raw カウントとベースラインの差分です。差分が負数であるか、またはノイズ閾値未満である場合、差分カウントは常に 0 に設定されます。

静電容量センサー
静電容量の変化によってタッチまたは近づいている物体に反応する導電体および基板 (プリント回路基板 (PCB) 上の銅ボタンなど) です。

CapSense®
サイプレスのタッチ センシング ユーザー インターフェース ソリューションです。業界 2 位に対して、4 倍の販売実績がある業界 No.1 ソリューション。

CapSense メカニカル ボタン リプレースメント (MBR)
メカニカル ボタンを静電容量ボタンにアップグレードするサイプレスの構成可能なソリューションであり、センサー パラメーターの設定に必要な設計工数を最小限に抑え、ファームウェアの開発も不要とします。これらのデバイスは CY8CMBR3XXX および CY8CMBR2XXX のファミリを含んでいます。

重心または重心位置
スライダー分解能の指定した範囲内のスライダー上の指の位置を示す数です。この数は CapSense 重心計算アルゴリズムにより算出されます。

補正 IDAC
過剰なセンサーC_P を補正するために CSD により使用されるプログラム可能な定電流源です。この IDAC は、変調 IDAC と違って、CSD ブロックでシグマ-デルタ変調器によって制御されません。

CSD
CapSense シグマ デルタ (CSD) は、静電容量センスのアプリケーション用に自己容量を測定するサイプレスの特許取得済み方法です。

CSD モードでは、センス システムは電極の自己容量を測定し、指の有無を識別するために自己容量の変化が検出されます。

デバウンス

有効なタッチとなるためにタッチがある必要な連続スキャン サンプル数を定義するパラメーターです。このパラメーターは怪しいタッチ信号を除去するために役立ちます。

指のタッチは、差分カウントがスキャン サンプルの連続デバウンス数で「指閾値 + ヒステリシス」を超える場合にのみ報告されます。

駆動シールド

シールド電極がセンサー スイッチング信号と同じ位相および振幅を持つ信号によって駆動され、耐液性を有効にするために CSD によって使用される技術です。

電極

プリント基板、ITO または FPCB 上のパッドまたは層などの導電材料です。電極は CapSense デバイス上のポート ピンに接続し、CapSense センサーとして使用されるか、または CapSense 機能に関する特定の信号を駆動するために使用されます。

指の閾値

センサーの状態を確定するためにヒステリシスと一緒に使用されるパラメーターです。センサーの状態は、差分カウントが指閾値 + ヒステリシスを上回る場合、オンとして報告され、差分カウントが指閾値 – ヒステリシスを下回る場合、オフとして報告されます。

連結センサー

複数のセンサーを連動して、単一のセンサーとしてスキャンする方法です。近接センス用のセンサーの領域を増やし、電力消費量を減少するために用いられます。

システムが低消費電力モードにある場合の電力を削減するために、センサーは個別にスキャンされずに、これらのすべてを連動して単一のセンサーとしてスキャンされ、時間を短縮させます。ユーザーがセンサーに触る場合、システムはアクティブ モードに移行し、全てのセンサーを個別にスキャンしてアクティブになったセンサーを検出します。

PSoC はファームウェアでセンサー連動をサポートし、これにより、複数のセンサーはスキャンのために AMUXBUS と同時に接続できます。

ジェスチャー

ジェスチャーはスワイプやピンチズームなどのユーザーの行動です。CapSense は事前に定義されたタッチ パターンに基づいて異なるジェスチャーを識別するジェスチャー検出機能を備えています。CapSense コンポーネントでは、ジェスチャー機能はタッチパッド ウィジェットのみによりサポートされます。

ガード センサー

ボタン センサーと同様に PCB 上の全てのセンサーを取り囲み、液体流を検出するために使用される銅配線です。ガードセンサーがトリガーされると、ファームウェアは誤ったタッチを防ぐために、すべての他のセンサーのスキャンを無効にすることができます。

ハッチ フィル／ハッチ グランド／ハッチド グランド

静電容量センスの PCB を設計する時、良好なノイズ耐性のために接地した銅面をセンサーの周りに配置する必要があります。しかし、ベタ グランドはセンサーの期待されない寄生静電容量を増加させます。そのため、グランドは特別なハッチ パターンで充填する必要があります。ハッチ パターンはメッシュのように密接に配置され、交差されるラインがあり、ラインの幅および 2 本のライン間の間隔は充填率を指定します。耐液性の場合、シールド電極 として呼ばれるこのハッチ フィルはグランドの代わりにシールド信号で駆動されます。

ヒステリシス

システム ノイズに起因してセンサー状態の出力がランダムにトグルすることを回避し、センサーの状態を決定するために指の閾値と一緒に使用されるパラメーターです。指の閾値を参照してください。

IDAC (電流出力デジタル-アナログ変換器)

CapSense および ADC 動作用の PSoC 内のプログラマブルな定電流源です。

耐液性

水滴、液体流や霧が存在する環境でも確実に動作する静電容量センス システムの能力です。

リニア スライダー

指の物理的な位置 (単一の軸で) を検出するために特定の直線状で配置された複数のセンサーを含むウィジェットです。

低ベースライン リセット

raw カウントが負のノイズ閾値を異常的に下回るスキャン サンプルの最大数を表すパラメーターです。低ベースライン リセットの値を超える場合、ベースラインは現時点の raw カウントにリセットされます。

手動チューニング

CapSense パラメーターの手動設定 (または手動チューニング) プロセスです。

マトリクス ボタン

マトリクス状で配置された 2 つ以上のセンサーを含んで、垂直方向および水平方向に配置されるセンサーの交差部に人の指 (タッチ) の有無を検出するために使用されるウィジェットです。

M を水平軸上のセンサーの数と、N を垂直軸上のセンサーの数とすれば、マトリクス ボタン ウィジェットは M + N 本のポート ピンだけを使用して合計で M x N 個の交差部を監視することができます。

CSD センス方式 (自己容量) を使用する場合、このウィジェットは同時に 1 点のみの交差位置で有効なタッチを検出できます。

変調コンデンサー (CMOD)

自己容量センス モードでの CSD ブロックの動作のために必要な外部コンデンサーです。

変調器クロック

センサーのスキャン間に CSD ブロックから変調器出力をサンプリングするために使用されるクロック ソースです。このクロックが raw カウント カウンターにも供給されます。スキャン時間 (事前および事後処理時間を除く) は「$(2^N – 1)$ /変調器クロック周波数」(N はスキャンの分解能) により計算されます 。

変調 IDAC

変調 IDAC はプログラマブルな定電流源であり、この出力は V_{REF} の AMUXBUS 電圧を維持するために、CSD ブロックのシグマ デルタ変調器の出力によって制御 (オン/オフ) されます。この IDAC によって供給される平均電流は、センサー コンデンサーが引き出した平均電流に等しいです。

相互静電容量

ある電極 (例えば、TX) と他の電極 (例えば RX) 間の静電容量は相互容量として知られています。

負のノイズ閾値

負の方向に出るスプリアス信号から通常のノイズを識別するために使用される閾値です。このパラメーターは低ベースライン リセット パラメーターと併用されます。

raw カウントが負のノイズ閾値を超えない (すなわち、ベースラインと raw カウントの差 (ベースライン – raw カウント) が負のノイズ閾値未満である) 限り、ベースラインは raw カウントの変化を追跡するために更新されます。

負の方向でスプリアス信号をトリガーする可能性があるシナリオは次の通りです: 電源投入時にセンサーに指が触れる時、センサーの近く配置される金属の物体を除去する時、耐液性のある CapSense 製品の水分を除去する時、および他の急激な環境変化がある時です。

ノイズ (CapSense ノイズ)

センサーがオフ状態 (タッチなし) 時にピーク ツー ピーク カウントとして測定される raw カウントの変化です。

ノイズ閾値

センサー用にノイズから信号を識別するために使用されるパラメーターです。「raw カウント - ベースライン」の差分がノイズ閾値を超える場合、おそらく有効な信号を示します。差分がノイズ閾値に下回る場合、raw カウントはノイズのみを含みます。

オーバーレイ

静電容量センサーをカバーしタッチ面として機能するプラスチックやガラスなどの非導電材料です。センサーがある PCB はオーバーレイの下に直接配置されるか、またはスプリングを介して接続されます。製品の筐体は常にオーバーレイになります。

寄生静電容量 (C_P)

寄生容量は PCB 配線、センサー パッド、ビアおよびエア ギャップによって与えられるセンサー電極の固有容量です。寄生容量は CSD の感度を減らすため、望ましくないものです。

近接センサー
物理的な接触無しで近接オブジェクトの存在性を検知できるセンサー。

ラジアル スライダー
指の物理位置を検出する特定の円形に配置される 1 つ以上のセンサーが含むウィジェット。

Raw カウント
センサーの物理的静電容量を示す CapSense ハードウェア ブロックの未処理デジタル カウント出力です。

リフレッシュ間隔
センサーの 2 回の連続スキャンの間の時間です。

スキャン分解能
CSD ブロックによって生成される raw カウントの分解能 (ビット数) です。

スキャン時間
センサーのスキャンを完了する必要な時間です。

自己静電容量
回路のグランドと電極間の静電容量です。

感度
センサー静電容量の変化に応じる raw カウントの変化であり、カウント/pF の単位で表します。センサーの感度は基板レイアウト、オーバーレイ特性、センス方式およびチューニング パラメーターに依存します。

センス クロック
CSD センス方式用のスイッチト コンデンサー回路のフロント エンドを実装するために使用されるクロック ソースです。

センサー
静電容量センサーを参照してください。

センサー自動リセット
センサーがシステム故障によって誤ったタッチ状態の報告から防ぎ、または金属物体がセンサーの近くに連続的に存在する場合の設定。

常に、センサーの自動リセットが有効化されている場合、差分カウントがノイズ閾値を超えてもベースライン値が更新されます。このように、センサーが無期限のオン状態を報告しないようにします。センサーの自動リセットが無効化されていると、ベースライン値は差分カウントがノイズ閾値を下回った場合にのみ更新されます。

センサー連結
連結センサーを参照してください。

シールド電極
センサーの周囲を覆う銅トレースで水分による誤タッチを防止。シールド電極は CSD ブロックからのシールド信号出力によって駆動されます。駆動シールドを参照してください。

シールド タンク コンデンサー (C_{SH})
高い寄生容量を持つ広いシールド層がある場合、CSD シールドの駆動能力を強化するために使用されるオプションの外部コンデンサー (C_{SH} タンク コンデンサー) です。

信号 (CapSense 信号)
差分カウントも信号と呼ばれる。差分カウントを参照してください。

信号対雑音比 (SNR)
タッチした時のセンサーの信号とタッチしない時のセンサーのノイズ信号との比率。

スライダー分解能
スライダーが分解された指の位置の総数を示すパラメーターです。

タッチパッド
特定の水平と垂直な様式で配置された複数のセンサーからなり、タッチの X および Y 位置を検出するウィジェットです。

トラックパッド

タッチパッドを参照してください。

調整

CapSense の動作に必要な様々なハードウェアおよびソフトウェアまたは閾値パラメーターの最適値を決定するプロセスです。

VREF

CapSense および ADC 動作用の PSoC 内のプログラマブルな電圧リファレンス ブロック。

ウィジェット

単一センサーまたは同様のセンサー グループで構成される CapSense コンポーネントのユーザー インターフェース要素です。ボタン、近接センサー、リニア スライダー、ラジアル スライダー、マトリクス ボタンおよびタッチパッドはサポートしたウィジェットです。

改訂履歴

改訂履歴

文書名: AN90071 - CY8CMBR3xxx CapSense® 設計ガイド

文書番号: 001-91966

版	ECN 番号	発行日	変更者	変更内容
**	4335530	02/19/2014	DCHE	これは英語版 001-90071 Rev. **を翻訳した日本語版 001-91966 Rev. **です。
*A	4741593	04/13/2015	HZEN	これは英語版 001-90071 Rev. *E を翻訳した日本語版 001-91966 Rev. *A です。
*B	6588849	06/05/2019	SSAS	これは英語版 001-90071 Rev. *H を翻訳した日本語版 001-91966 Rev. *B です。