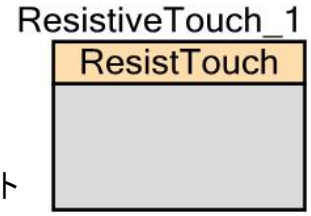
抵抗膜式タッチ(ResistiveTouch)

1.0



特長

- 4 線抵抗膜式タッチスクリーン・インターフェイスをサポート
- PSoC 3 および PSoC 5 デバイスの両方に対応するデルタ・シグマ変換器をサポート
- PSoC 5 デバイス用 ADC 逐次比較型レジスタをサポート

概要説明

この抵抗膜式タッチスクリーン・コンポーネントは 4 線抵抗膜式タッチ スクリーンインターフェイスに使用されます。コンポーネントは emWin グラフィックライブラリを搭載したタッチスクリーンの抵抗膜式タッチ要素を統合および設定する方法を提供します。これはタッチパネルのポーリング時に emWin で与えられるタッチスクリーン・ドライバによって呼び出されるハードウェア依存機能を統合します。

ResistiveTouch の用途

低コストで簡単なインターフェイスの電子技術が必要とされる ResistiveTouch コンポーネントを採用。

入出力接続

このセクションでは、ResistiveTouch のさまざまな入出力接続について説明します。

xm − Digital Input / Output

抵抗膜式タッチパネル X 軸からの x–信号 (active low)。

xp − Analog / Digital Output

抵抗膜式タッチパネル X 軸からの x+信号(active high)。

ym − Digital Input / Output

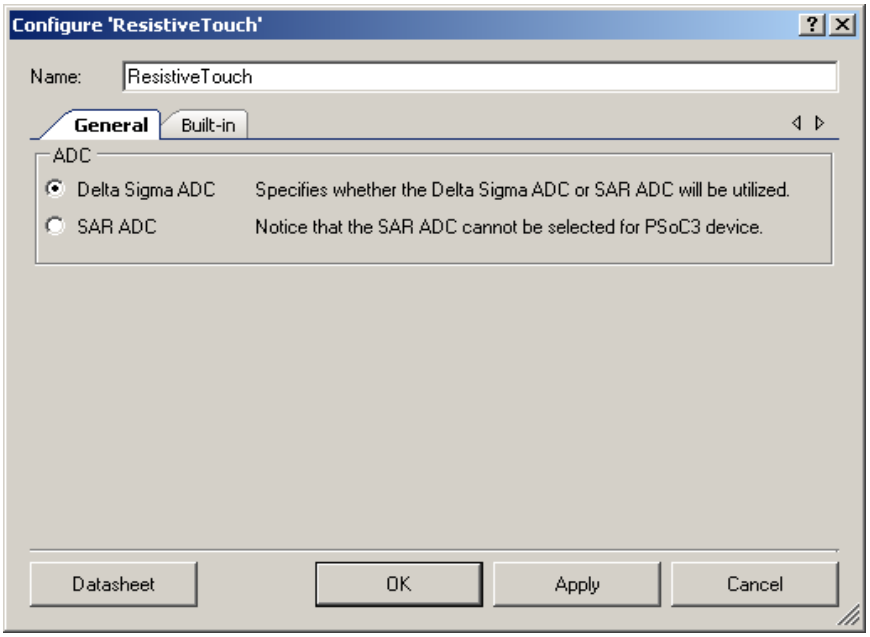
抵抗膜式タッチパネル Y 軸からの y–信号(active low)。

yp – Analog / Digital Output

抵抗膜式タッチパネル Y 軸からの y+信号(active high)。

コンポーネント・パラメータ

ResistiveTouch コンポーネントを回路図上にドラッグし、ダブルクリックして **Configure**（**設定**）ダイアログを開きます。



ResistiveTouch コンポーネントには次のパラメータがあります。

ADC

ADC パラメータはどの ADC を使用するのかを決定します。PSoC 3 デバイスの場合、**Delta Sigma ADC** オプションを選択します。

配置

ResistiveTouch のすべての配置情報は、*cyfitter.h* ファイルを通じて API に提供されます。

リソース

ResistiveTouchコンポーネントは外部抵抗膜式タッチパネル デバイスとの接続に使用されるPSoC 3/PSoC 5 シリーズのアナログ/デジタル ハードウェア専用部品の一部です。

リソース	リソースのタイプ					API メモリ（バイト）		ピン（外部入出力ごと）
	デシメータ	SAR	DSM	PLD	割り込み	フラッシュ	RAM	
ResistiveTouch_DelSig	TBD	TBD	TBD	TBD	TBD	TBD	TBD	TBD
ResistiveTouch_SAR	TBD	TBD	TBD	TBD	TBD	TBD	TBD	TBD

アプリケーション プログラミング インタフェース

アプリケーション・プログラミング・インターフェース (API) ルーチンにより、ソフトウェアを使用してコンポーネントを設定できます。次の表は、各関数へのインターフェースとその説明を示しています。続くセクションでは、各関数について詳しく説明します。

初期設定では、PSoC Creator は、最初に配置したピンコンポーネントのインスタンス名として“ResistiveTouch_1”を割り当てます。コンポーネントのインスタンス名称は、識別子の文法ルールに従って固有の名前に変更できます。インスタンス名は、すべてのグローバル関数名、変数名、定数名のプリフィックスになります。理解しやすいように、次の表では、インスタンス名「ResistiveTouch」を使用しています。

関数	説明
ResistiveTouch_Start()	ResistiveTouch_Init() および ResistiveTouch_Enable() API を呼び出します。
ResistiveTouch_Stop()	DelSig ADC または SAR ADC を停止し、AMux コンポーネントを停止します。
ResistiveTouch_Init()	DelSig ADC または SAR ADC の Init 関数と AMux コンポーネントを呼び出します。
ResistiveTouch_Enable()	DelSig ADC またはSAR ADC をイネーブルにし、AMux コンポーネントを有効にします。
ResistiveTouch_ActivateY()	Y 軸の測定用ピンを設定します。
ResistiveTouch_ActivateX()	X 軸の測定用ピンを設定します。
ResistiveTouch_TouchDetect()	スクリーンのタッチを検出します。
ResistiveTouch_Measure()	A/D 変換器の結果を返します。
ResistiveTouch_SaveConfig()	DelSig ADC または SAR ADC の設定を保存します。
ResistiveTouch_Sleep()	SaveConfigおよびStop関数を呼び出すことで、低消費電力モードのDelSig ADCまたはSAR ADCを準備します。
ResistiveTouch_RestoreConfig()	DelSig ADまたはSAR ADCの設定を復元します。
ResistiveTouch_Wakeup()	低消費電力モードから起動後、DelSig ADCまたはSAR ADCを設定を復元します。

グローバル変数

変数	説明
ResistiveTouch_initVar	ResistiveTouchが初期化されたかどうかを示します。変数は、0に初期化され、ResistiveTouch_Start()が初めて呼び出されたときに 1 に設定されます。これにより、コンポーネントは ResistiveTouch_Start()ルーチンへの最初の呼び出し後、再初期化せずに再起動できます。 コンポーネントの再初期化が必要な場合、ResistiveTouch_Init()やResistiveTouch_Enable()関数の前に、ResistiveTouch_Init()関数が呼び出されます。
ResistiveTouch_enableVar	この変数はコンポーネント状態のイネーブル／ディスエーブルを表示するのに使用されます。

void ResistiveTouch_Start(void)

説明: ResistiveTouch_Init()および ResistiveTouch_Enable() APIを呼び出します。

パラメータ: なし

返り値: なし

副作用: なし

void ResistiveTouch_Init(void)

説明: DelSig ADCまたは SAR ADC、および AMuxコンポーネントの Init関数を呼び出します。

パラメータ: なし

返り値: なし

副作用: なし

void ResistiveTouch_Enable(void)

説明: DelSig ADCまたはSAR ADCをイネーブルにし、AMuxコンポーネントを有効にします。

パラメータ: なし

返り値: なし

副作用: なし

void ResistiveTouch_Stop(void)

説明: DelSig ADCまたは SAR ADCを停止し、AMuxコンポーネントを停止します。

パラメータ: なし

返り値: なし

副作用: なし

void ResistiveTouch_ActivateX(void)

説明: X 軸の測定用ピンを設定します。

パラメータ: なし

返り値: なし

副作用: なし

void ResistiveTouch_ActivateY(void)

説明: Y 軸の測定用ピンを設定します。

パラメータ: なし

返り値: なし

副作用: なし

int16 ResistiveTouch_Measure(void)

説明: A/D変換器の結果を返します。

パラメータ: なし

返り値: int16: ADC変換の結果

副作用: なし

uint8 ResistiveTouch_TouchDetect(void)

説明: スクリーンのタッチを検出します。

パラメータ: なし

返り値: uint8: タッチ状態
0 – 非タッチ
1 – タッチ

副作用: なし

void ResistiveTouch_SaveConfig(void)

説明: DelSig ADCまたは SAR ADCの設定を保存します。

パラメータ: なし

返り値: なし

副作用: なし

void ResistiveTouch_RestoreConfig(void)

説明: DelSig ADCまたはSAR ADCの設定を復元します。

パラメータ: なし

返り値: なし

副作用: なし

void ResistiveTouch_Sleep(void)

説明: SaveConfigおよびStop関数を呼び出すことで、低消費電力モードの DelSig ADCまたは SAR ADCを準備します。

パラメータ: なし

返り値: なし

副作用: なし

void ResistiveTouch Wakeup(void)

説明: 低消費電力モードから起動後、DelSig ADCまたはSAR ADCの設定を復元します。

パラメータ: なし

返り値: なし

副作用: なし

ファームウェア・ソースコードのサンプル

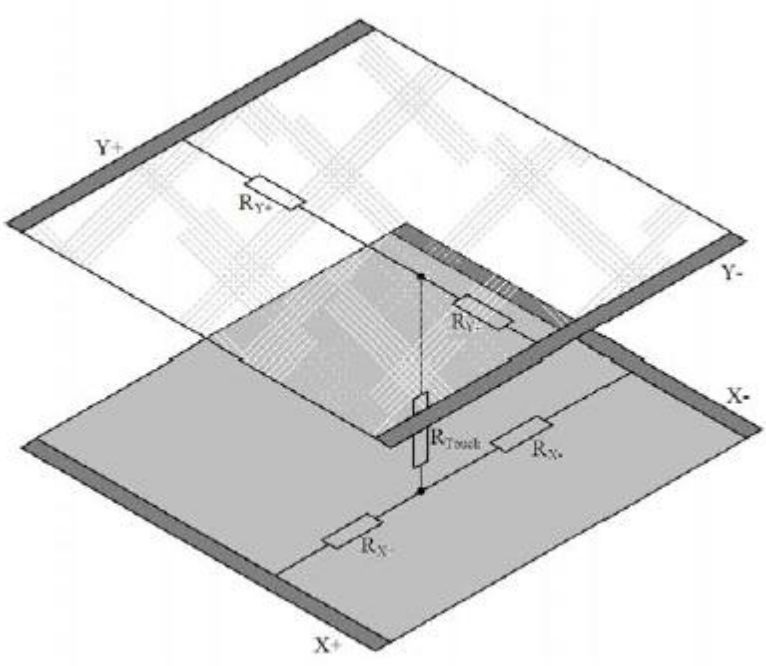
PSoC Creator は、[Find Example Project（サンプルプロジェクトを検索）] ダイアログに多数のサンプルプロジェクトを提供しており、そこには回路図およびサンプル コードが含まれています。コンポーネント固有の例を見るには、[Component Catalog] または回路図に置いたコンポーネント インスタンスからダイアログを開きます。一般例については、[Start Page] または **[File（ファイル）]** メニューからダイアログを開きます。必要に応じてダイアログにある **Filter Options** を使用し、選択できるプロジェクトのリストを絞り込みます。

詳しくは、PSoC Creator ヘルプの「Find Example Project（サンプルプロジェクトを検索）」を参照してください。

機能説明

このコンポーネントは 4 線抵抗膜式タッチスクリーン・インターフェイスを備え、タッチスクリーン座標を読み取りスクリーン抵抗を測定できます。このコンポーネントは、抵抗を画面座標に変換する SEGGER emWin グラフィック・ライブラリのタッチスクリーン機能へのアクセスを提供します。

下の図は圧力が印加された時の 4 線抵抗膜式タッチスクリーンの回路図を示しています。



接点は、2 つの抵抗（図を参照）を有する直列の抵抗回路の各レイヤを「分割」します。この接点は 2 つのレイヤ間の抵抗を接続します。この点の電圧を測定することで、電圧勾配に直角の接点位置情報が得られます。座標一式を得るには、一度、電圧勾配を垂直方向にしてから水平方向にして印加する必要があります。最初に印加する供給電圧を 1 つのレイヤに加え、他のレイヤ全体の電圧を測定し、次に供給電圧を他のレイヤに接続して反対側のレイヤ電圧を測定する必要があります。タッチ モードの場合、ラインの 1 つがタッチ動作を検出するために接続されています。次の表では座標またはタッチを測定中のピン構成を定義しています。

	XP	XM	YP	YM
タッチ	Res Pullup	Digital Hi-Z	Analog Hi-Z	Strong (ストロング) ドライブ
X 座標	Strong (ストロング) ドライブ	Strong (ストロング) ドライブ	Analog Hi-Z	Analog Hi-Z
Y 座標	Analog Hi-Z	Analog Hi-Z	Strong (ストロング) ドライブ	Strong (ストロング) ドライブ

ResistiveTouch コンポーネントがプロジェクト回路図に配置されている場合、I/O ポートの指定を xm、xp、ym、yp ピンに設定することが重要です。それらのピンの指定はシンボルや回路図ではなく、「Design-Wide Resources（設計全体リソース）」ウィンドウの「Pins（ピン）」タブで実施します。

DC 電気的特性と AC 電気的特性

TBD

コンポーネントの変更

バージョン 1.0 は ResistiveTouch コンポーネントの最初のリリースです。

Copyright © 2005-2012 Cypress Semiconductor Corporation 本文書に記載される情報は、予告なく変更される場合があります。Cypress Semiconductor Corporationは、サイプレス製品に組み込まれた回路以外のいかなる回路を使用することに対しても一切の責任を負いません。特許又はその他の権限下で、ライセンスを譲渡又は暗示することもありません。サイプレス製品は、サイプレスとの書面による合意に基づくものでない限り、医療、生命維持、救命、重要な管理、又は安全の用途のために仕様することを保証するものではなく、また使用することを意図したものでもありません。さらにサイプレスは、誤動作や故障によって使用者に重大な傷害をもたらすことを合理的に予想される、生命維持システムの重要なコンポーネンツとしてサイプレス製品を使用することを許可していません。生命維持システムの用途にサイプレス製品を供することは、製造者がそのような使用におけるあらゆるリスクを負うことを意味し、その結果サイプレスはあらゆる責任を免除されることを意味します。

PSoC Designer™及びProgrammable System-on-Chip™は、Cypress Semiconductor Corp.の商標、PSoC®は同社の登録商標です。本文書で言及するその他全ての商標又は登録商標は各社の所有物です。

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

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

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