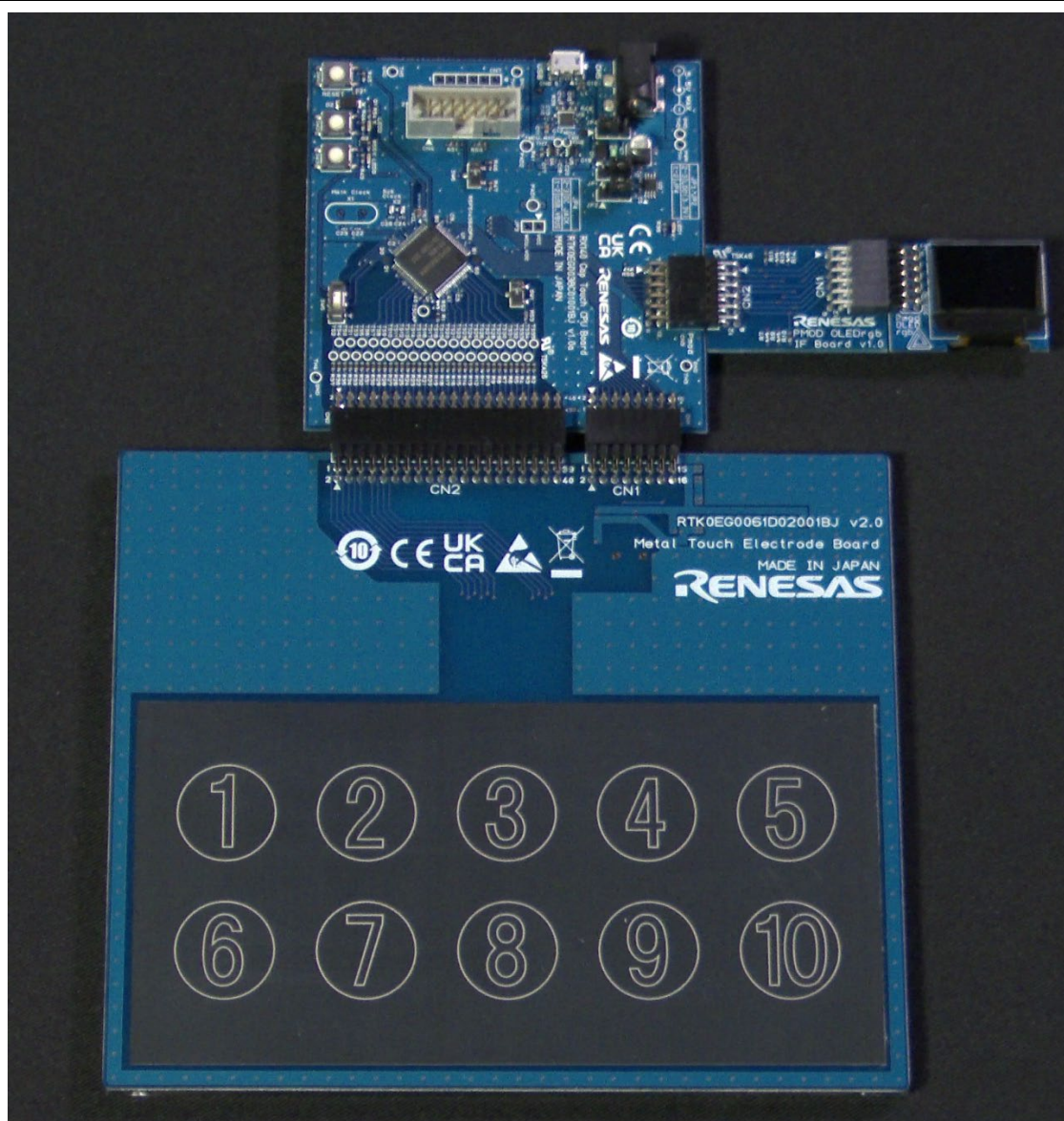


RX140 グループ

金属パネルタッチセンサ例

要旨

本アプリケーションノートは RX140 の静電容量式タッチセンサユニット(Capacitive Touch Sensing Unit: 以下 CTSU)を用いて、タッチセンサの電極のオーバーレイ（タッチセンサの電極を保護や静電容量検出の特性を安定させるために用いるカバー素材）として金属パネルを使用した静電容量式タッチセンサの実現例を示します。本方式では金属パネルを指などで押した際のたわみによる静電容量変化を計測し、タッチの判定を行います。金属パネルによるタッチセンサではたわみを計測するため、押し込み強度の判定により 1 つのボタンで多段階のタッチ判定が行えます。金属筐体をそのまま利用できるため、あらたにタッチのためのオーバーレイを用意するコストの削減が期待でき、また外観上も優れたボタンを搭載できます。



目次

1. 概要	4
2. システム動作	5
3. 動作確認環境	6
4. 関連ドキュメント	6
5. 動作原理	7
5.1 コンデンサの基本モデル	7
5.2 非導電性オーバーレイを使用した静電容量式センサ	8
5.3 導電性オーバーレイを使用した静電容量センサ（金属パネルタッチセンサ）	9
5.4 CTSU の計測原理	10
6. システム構成	11
6.1 基板構成	12
6.1.1 CPU ボード（RX140 RSSK）	12
6.1.2 タッチ電極ボード	13
6.1.3 PMOD OLEDrgb IF ボード	16
6.1.4 PMOD OLEDrgb (ディスプレイ)	17
6.2 機械構成	18
6.2.1 センサ部の構造	18
6.2.2 金属パネル層	19
6.2.3 スペーサ層	21
6.2.4 基板層	22
6.2.5 センサ設計	23
6.2.6 組立方法	27
7. サンプルプログラム	28
7.1 動作概要	28
7.2 使用する MCU 機能と設定	30
7.2.1 静電容量タッチ検出	32
7.2.1.1 使用するモジュールの設定	32
7.2.1.2 QE for Capacitive Touch の設定	34
7.2.2 表示器制御	36
7.2.3 ブザー発音	37
7.2.4 SW3 と LED3	38
7.2.5 その他端子設定	38
7.3 タッチ検出処理	39
7.3.1 タッチ判定しきい値設定	39
7.3.2 タッチ検出処理	40
7.4 ブザー発音処理／停止処理	41
7.5 表示処理	42
7.6 プログラム構成	44
7.6.1 ソースファイル構成	44

7.6.2	マクロ定義.....	45
7.6.3	構造体・共用体・列挙型.....	47
7.6.4	関数一覧.....	48
8.	プロジェクトをインポートする方法.....	53
8.1	e ² studio での手順.....	53
8.2	CS+での手順.....	54
9.	サンプルプログラムを使用した評価結果.....	55
9.1	使用メモリと実行サイクル数.....	55
9.1.1	ビルド条件.....	55
9.1.2	使用メモリ.....	55
9.1.3	実行サイクル数と処理時間.....	56
9.2	評価結果.....	57
9.2.1	条件.....	57
9.2.2	ボタン感度.....	58
9.2.3	EMC 試験結果.....	59
	改訂記録.....	60

1. 概要

本アプリケーションノートは静電容量式タッチセンサのオーバーレイに金属パネルを用いたシステムを開発するシステム設計者、ハードウェア設計者、およびソフトウェア設計者を対象としたガイドです。

金属パネルを用いたタッチセンサは従来のタッチセンサと異なり、金属を押し込んだときのたわみを静電容量変化で捉えることでタッチの状態、強度を検出できます。そのため、非導電性の手袋を使用してもタッチ検出が可能で、水滴、汚れなどの影響も受けません。従来の機械式ボタンと比較して、機械的可動部が必要なく、耐水性が高く、水滴や汚れにも強く幅広い環境条件において、高い信頼性でボタン機能を実装することが可能です。

本書では金属パネルを用いた静電容量式タッチセンサの検出原理、特徴、デモキットの構成や回路、筐体などの設計情報やソフトウェア構成について紹介します。デモキットについては下記へお問い合わせください。

<https://www.renesas.com/buy-sample/locations>

また、本書では MCU 内蔵の静電容量式タッチセンサ名を、CTSU/CTSUa/CTSUB は CTSU1、CTSU2/CTSU2L/CTSU2La/CTSU2SL は CTSU2 と表記します。静電容量式タッチセンサユニットの基本的な扱い方や、ユニット自体の詳細な説明、CTSU1 と CTSU2 の特徴や種類、違いなどについてはアプリケーションノート「静電容量センサマイコン 静電容量タッチ導入ガイド」を参考にしてください。

2. システム動作

本システムは以下 2 種のタッチ検出と表示を行います。

- 操作モード 1：ボタン電極の押下を検出し、ブザー発音と OLED へ押下を表示
- 操作モード 2：ボタン電極の押下強度を 3 段階で検出し、ブザー発音と OLED へ押下強度を表示

表 2-1 と図 2-1 にボードの設定、操作と表示を示します。本システムを使用するときは必ず JP1 と JP2 の 2 番ピンと 3 番ピンを接続してから電源を投入してください。

表 2-1 ボードの設定、操作と表示

分類	項目	設定・操作・表示		
準備	JP	LDO 電源選択	JP1	2-3
			JP2	2-3
		電源選択	JP4	USB: 1-2 DC Jack: 2-3
		TS2,TS4 を使用	SW5	OFF (下側)
	電源	USB または DC Jack から 5V 入力		
操作	SW3	操作モード切替	初期設定：操作モード 2	
表示	LED3	操作モード表示	消灯：操作モード 1	
			点灯：操作モード 2	
	OLED	操作モードとボタンの押下状態の表示	操作モード 1	押下状態: 0, 1
			操作モード 2	押下強度: 0, 1, 2, 3

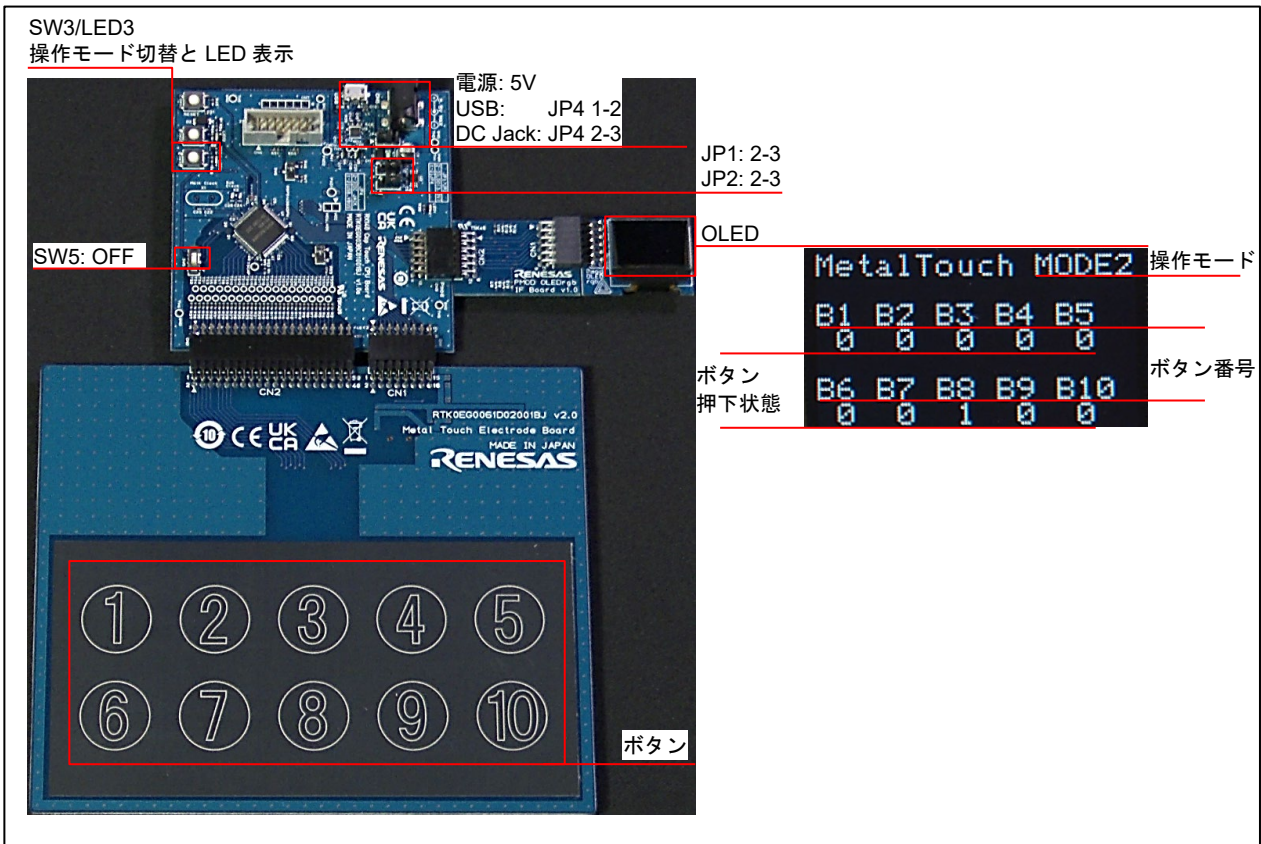


図 2-1 設定、操作と表示

3. 動作確認環境

表 3-1 動作確認環境

項目		名称
CPU ボード		RX140 搭載静電容量タッチ評価システム CPU ボード (RTK0EG0038C01001BJ)
	MCU	RX140 (R5F51406ADFN)
電極ボード		Metal Touch Electrode Board (RTK0EG0061D02001BJ)
表示器接続ボード		PMOD OLEDrgb IF Board
表示器		Digilent Pmod OLEDrgb (410-323)
IDE		e2 studio Version 2025-01 RX Smart Configurator V25.1.0 QE for Capacitive Touch V4.1.0
	FIT Modules	QE TOUCH module V3.1.0 QE CTSU module V3.1.0 DTC module V4.5.0
Tool Chain		CC-RX V3.07.00
エミュレータ		E2 エミュレータ Lite

4. 関連ドキュメント

- [R30AN0424 静電容量センサマイコン 静電容量タッチ導入ガイド](#)
- [R12UZ0102 RX140 グループ 静電容量タッチ評価 システムユーザーズマニュアル](#)
- [R01AN4516 RX ファミリ QE と FIT を使用した静電容量タッチアプリケーションの開発](#)
- [R01UH0905 RX140 グループ ユーザーズマニュアル ハードウェア編](#)

5. 動作原理

5.1 コンデンサの基本モデル

図 5-1 にコンデンサの基本的なモデルである平行板コンデンサの構成を示します。

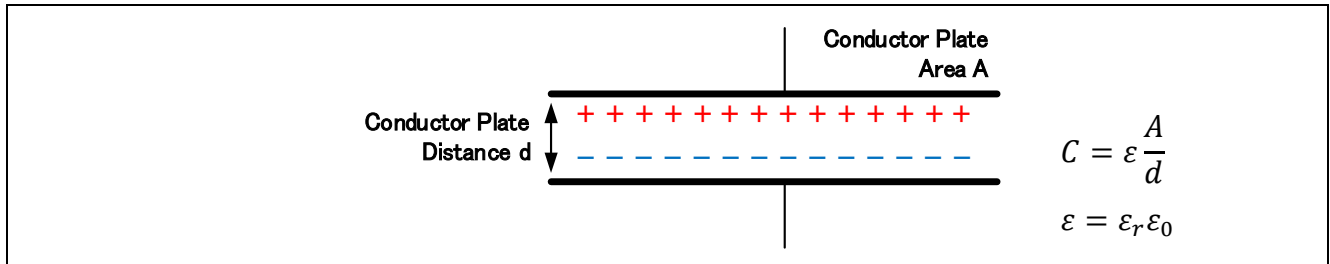


図 5-1 平行板コンデンサ

ここで、

C : 平行板コンデンサの静電容量 [F]

A : 2 枚の板が重なる部分の面積 [m^2]

ϵ_r : 電極の間にある材質の比誘電率(無単位)

ϵ_0 : 電気定数($\epsilon_0 \approx 8.854 \times 10^{-12}$) [F/m]

d : 2 枚の板の距離 [m]

平行板コンデンサは 2 枚の電極で構成され、その静電容量は 2 枚の電極の面積と電極の間にある材質の誘電率に比例し、電極間の距離に反比例します。

5.2 非導電性オーバーレイを使用した静電容量式センサ

図 5-2 に非導電性オーバーレイを使用した一般的な自己容量方式の静電容量式タッチセンサの構造を示します。一般的な静電容量式タッチセンサの場合は基板上に形成した電極と指がコンデンサを構成します。電極は寄生容量 C_p を介して基板の GND に接続されます。電極に指を近づけることで電極と指の間の静電容量 C_s が電極と PCB GND 間の寄生容量 C_p に加わり、この静電容量の増加分を計測することでタッチを検出することが可能です。本方式では一般的にアクリル板やガラス板などの非導電性オーバーレイを電極上に配置します。空気と比較して高い比誘電率を持つ材質のオーバーレイを使用することにより感度の向上が期待できます。さらに、高い絶縁性能を持つ材質を使用することで静電気放電（ESD）による搭載デバイスの破壊や誤動作のリスクを低減することができます。

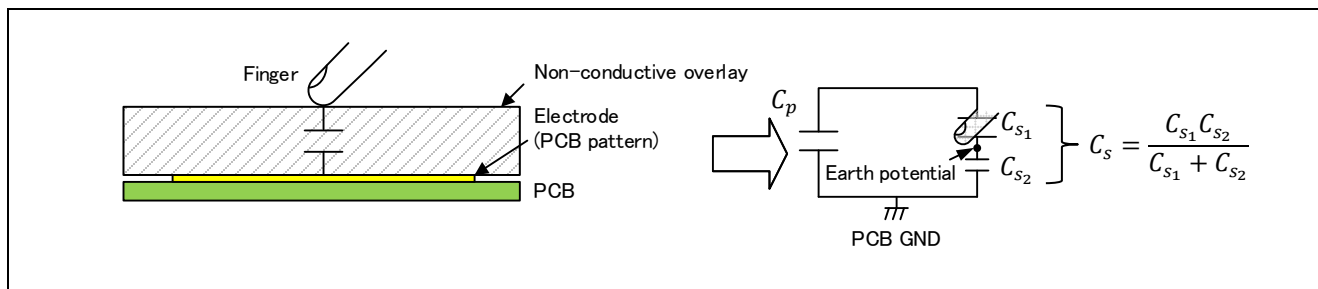


図 5-2 一般的な静電容量式タッチセンサの構成

ここで、

C_p : 電極－PCB GND 間の寄生容量 [F]

C_s : 指－PCB GND 間の静電容量 [F]

C_{s1} : 指－アース電位間の寄生容量 [F]

C_{s2} : アース電位－PCB GND 間の寄生容量 [F]

5.3 導電性オーバーレイを使用した静電容量センサ（金属パネルタッチセンサ）

図 5-3 に導電性オーバーレイを使用した金属パネルタッチセンサの構成を示します。金属パネルはスペーサにより基板との距離が固定されています。また、金属パネルは基板の GND に接続されています。金属パネルに力を加えるとわずかなひずみが生じ、金属パネルと基板上に形成した電極との距離が変化することで金属パネルと電極間の静電容量 C_s が変化します。本方式は金属パネルと電極との距離を静電容量で計測するセンサとして動作するため、静電容量の変化度合いによって金属パネルを押し込む力の強度を計測できます。

金属パネルは GND 電位で安定しているため人体の寄生容量は計測に影響せず、ユーザが非導電性の手袋をしていてもタッチの検出ができます。また、水滴が金属パネル表面についても影響はありません。

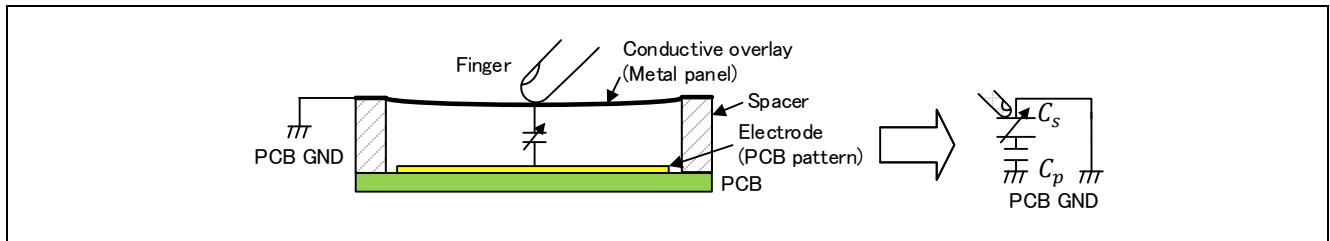


図 5-3 金属パネルタッチセンサの構成

ここで、

C_p : 電極—PCB GND 間の寄生容量 [F]

C_s : 金属パネル—電極間の静電容量 [F]

5.4 CTSU の計測原理

図 5-4 に CTSU の計測部回路を示します。

CTSU は TS 端子よりセンサドライブパルスを出力して、その充放電電流を計測することで静電容量値を計測します。電極側の電流 I 、センサドライブパルス周波数 F 、寄生容量を C_p 、検出対象の静電容量を C_s 、センサドライブパルス電圧 V とすると次の式が成立します。

$$I = F(C_s + C_p)V$$

ここで、電流 I は計測用 VDC (Voltage Down Converter) から供給される電流 I_1 とオフセット DAC から供給される電流 I_2 を加算した電流です。

計測用 VDC から供給した電流 I_1 は、カレントミラー回路を介して電流制御発振器 (Current Controlled Oscillator: 以下 CCO) に比例した電流 I_{OUT} が印加されます。CCO は I_{OUT} に比例した周波数のパルスをセンサカウンタに出力します。センサカウンタはセンサドライブパルスを一定時間計測して、センサカウンタレジスタに電流 I_{OUT} の計測値を格納します。

寄生容量 C_p だけの場合と指などの接触により C_s が印加された場合では電流量 (計測値) が異なることを利用して電流計測値の変化から静電容量の変化を検出します。

TSCAP 端子に接続しているコンデンサは内部電圧を安定させる役割があります。

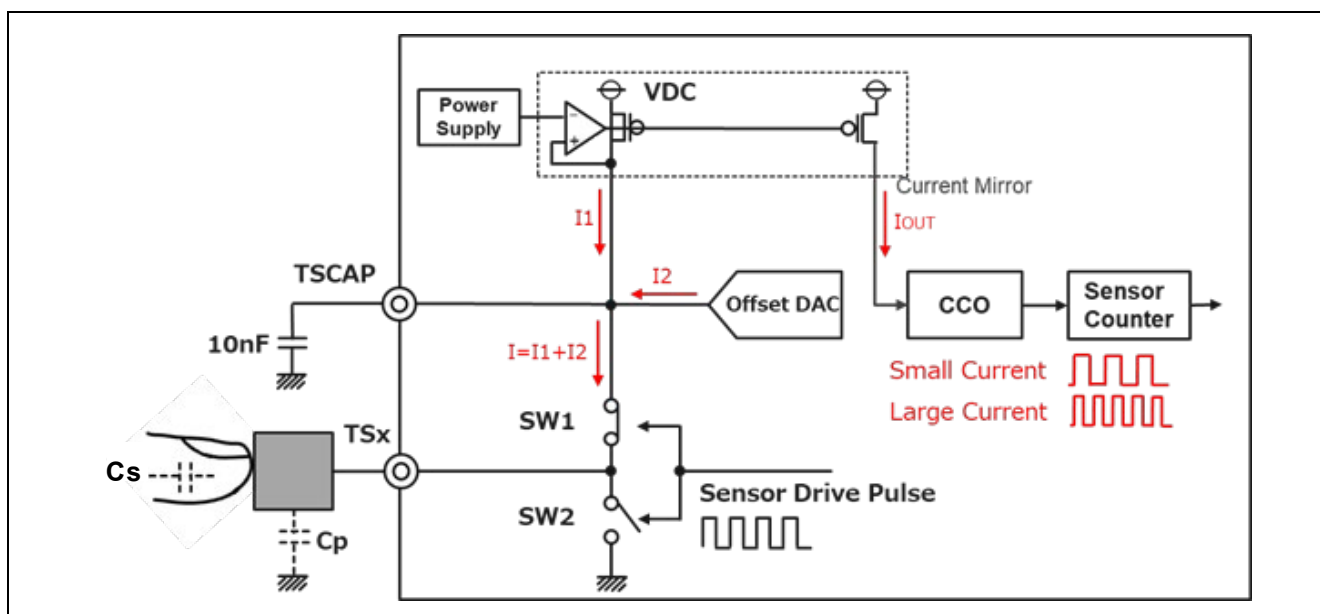


図 5-4 CTSU の計測部回路

6. システム構成

デモキットのシステム構成図を図 6-1 に示します。本デモキットは CPU ボード、タッチ電極ボード、PMOD OLEDrgb IF ボード、PMOD OLEDrgb (ディスプレイ)、の 4 枚の基板で構成されます。

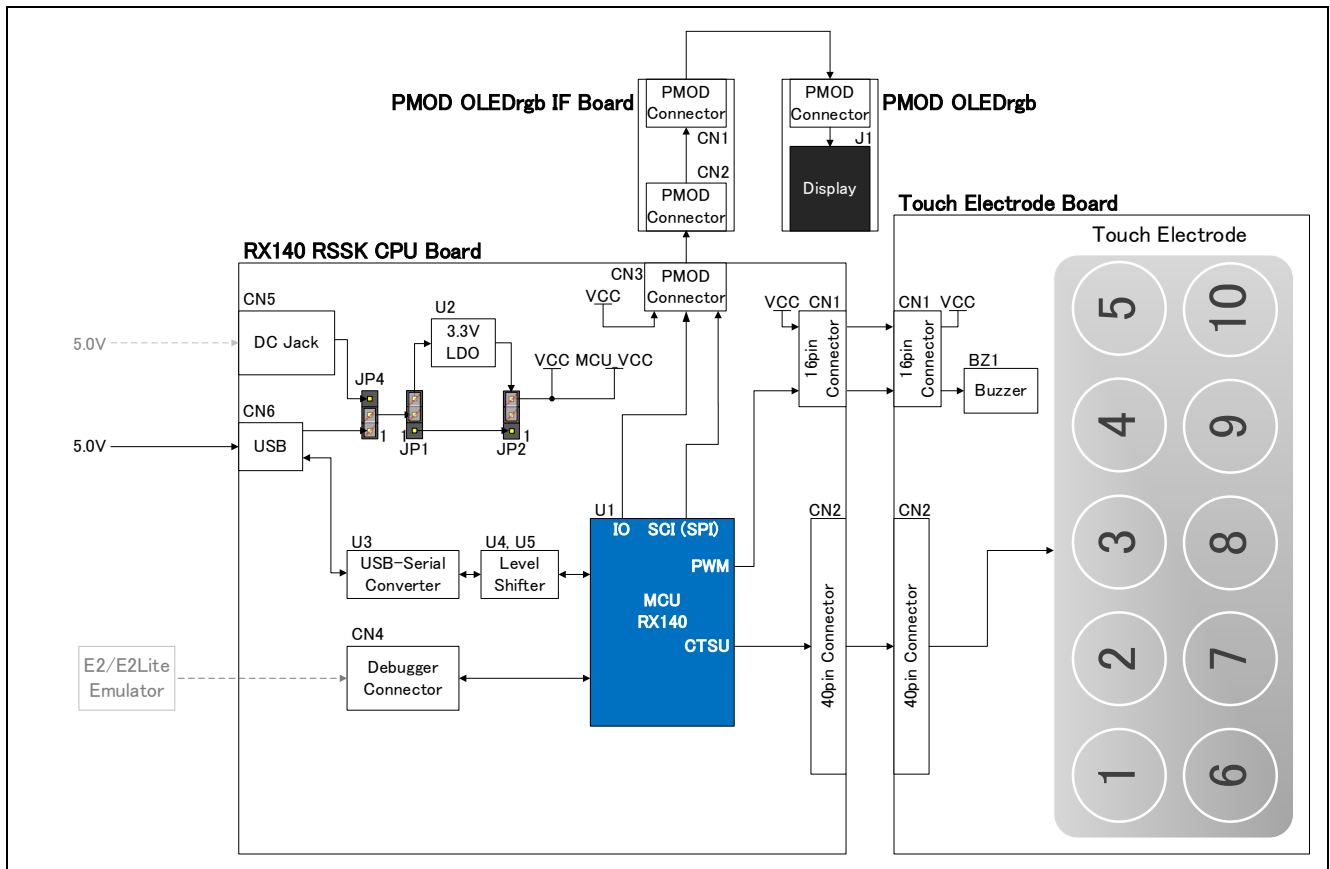


図 6-1 デモキット システム構成図

6.1 基板構成

6.1.1 CPU ボード (RX140 RSSK)

図 6-2 に CPU ボード (RX140 RSSK) の外観を示します。CPU ボードは、RX140 が搭載されており、各ボードへの電源供給、PC との通信、タッチ電極ボードの容量計測とタッチ判定、PMOD OLEDrgb との通信を行います。PMOD OLEDrgb の最大定格は 3.5V のため JP1 と JP2 の 2 番ピンと 3 番ピンをジャンパソケットでクローズし U2 の 3.3V LDO を使用するように設定してください。(表 2-1 参照)

CPU ボードの詳細は「RX140 グループ 静電容量タッチ評価 システムユーザズマニュアル」を参照してください。

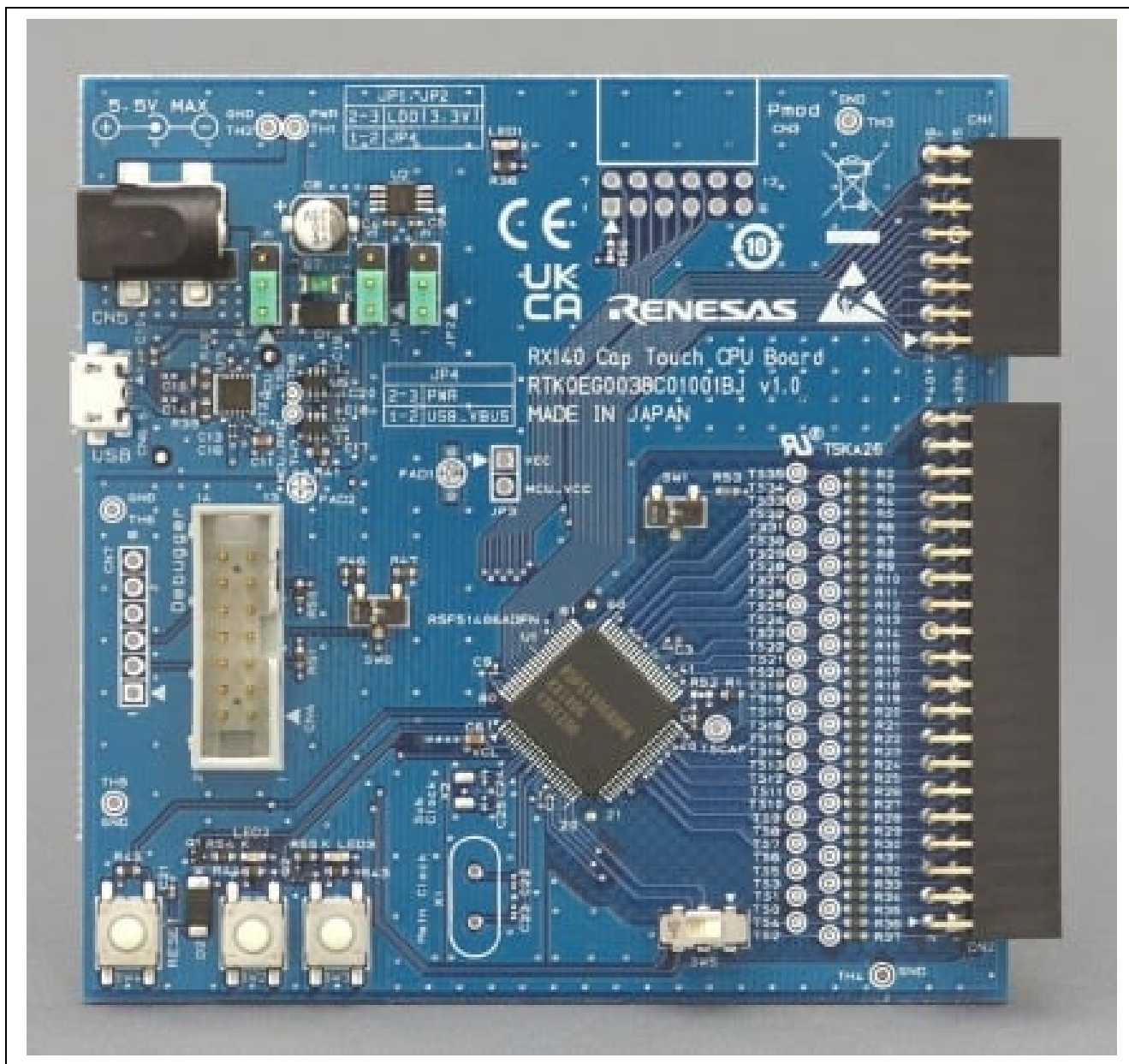


図 6-2 CPU ボード(RTK0EG0038C01001BJ)の外観

6.1.2 タッチ電極ボード

図 6-3 にタッチ電極の外観を示します。タッチ電極ボードは 10 個のタッチ電極と、タッチ判定時に発音するブザーを搭載しています。図 6-4、図 6-5、図 6-6 にタッチ電極ボードの図面を示します。タッチ電極ボードは両面基板で、部品面側にΦ20mm の円形の電極を、電極の中心から 30mm の間隔を空けて 10 個配置しています。半田面には電極をノイズから保護するためのシールドを配置しています。シールドは電極と GND 間の寄生容量の増加を防ぐためにメッシュ形状にしています。金属パネル貼り付け位置の四隅には金属パネルと基板 GND の導通を確実にするためにスプリングローデッドピン(TP3~TP6)を配置しています。

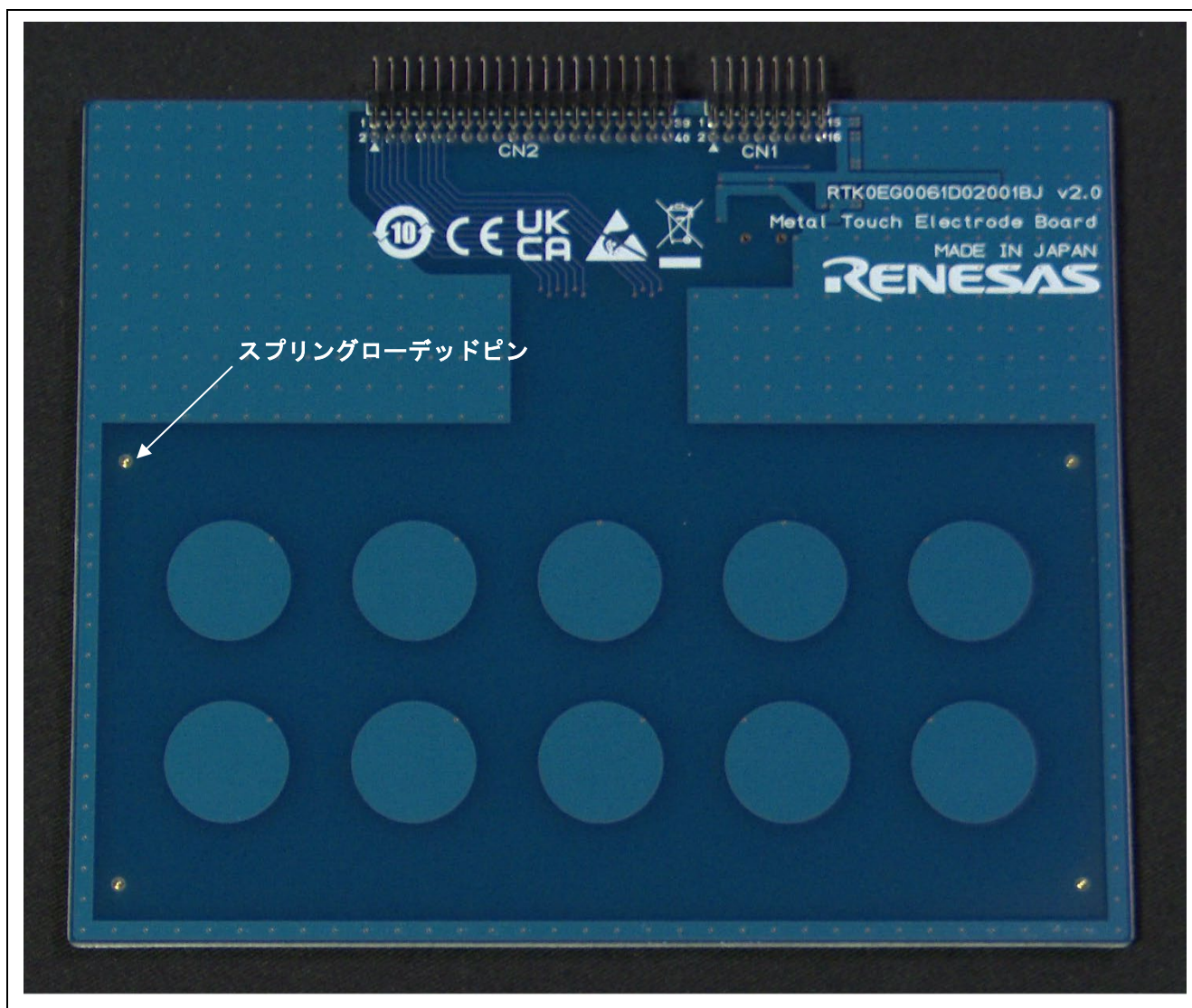


図 6-3 タッチ電極ボードの外観（金属板貼り付け前）

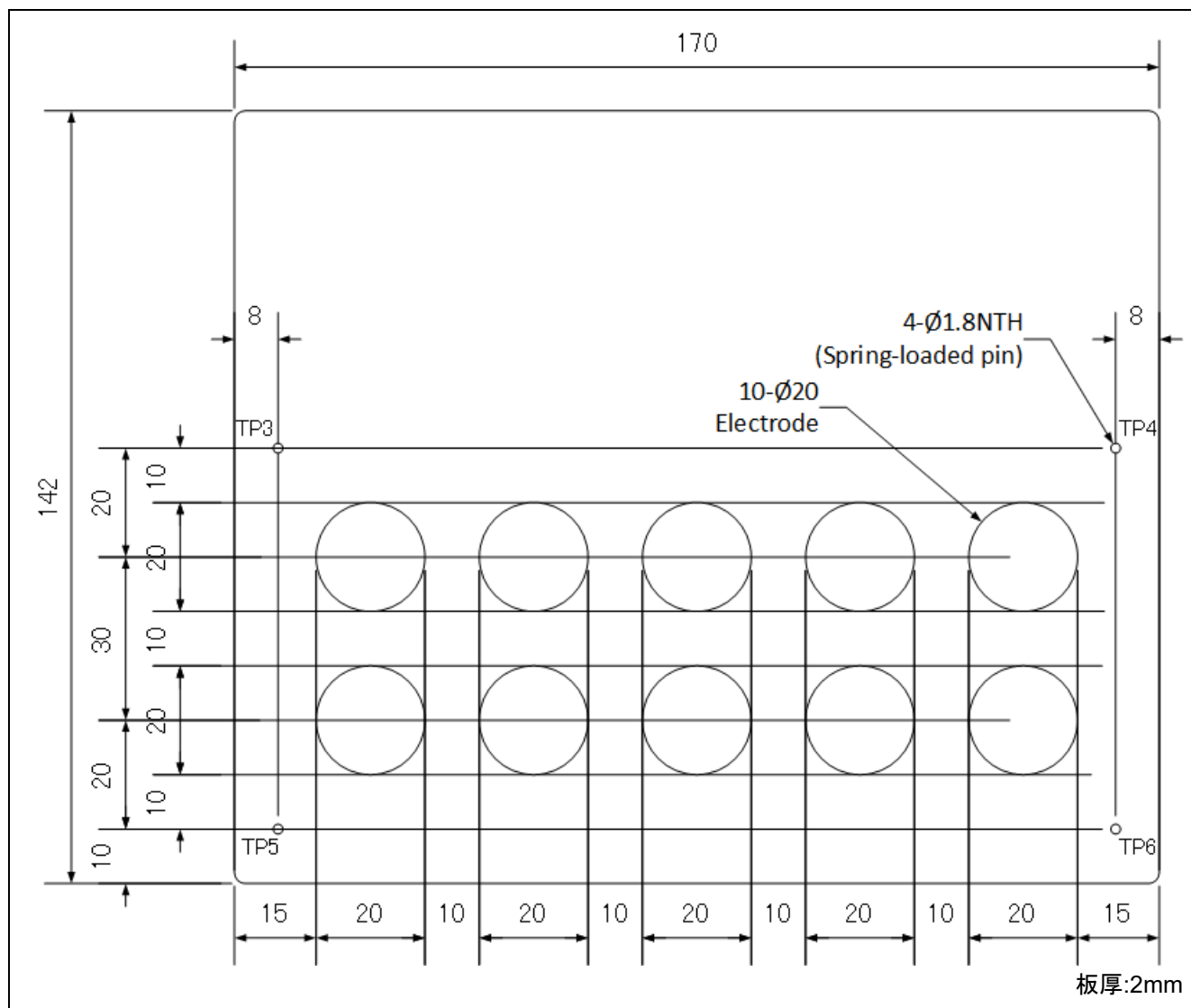


図 6-4 タッチ電極ボード 電極部の寸法 (単位 : mm)

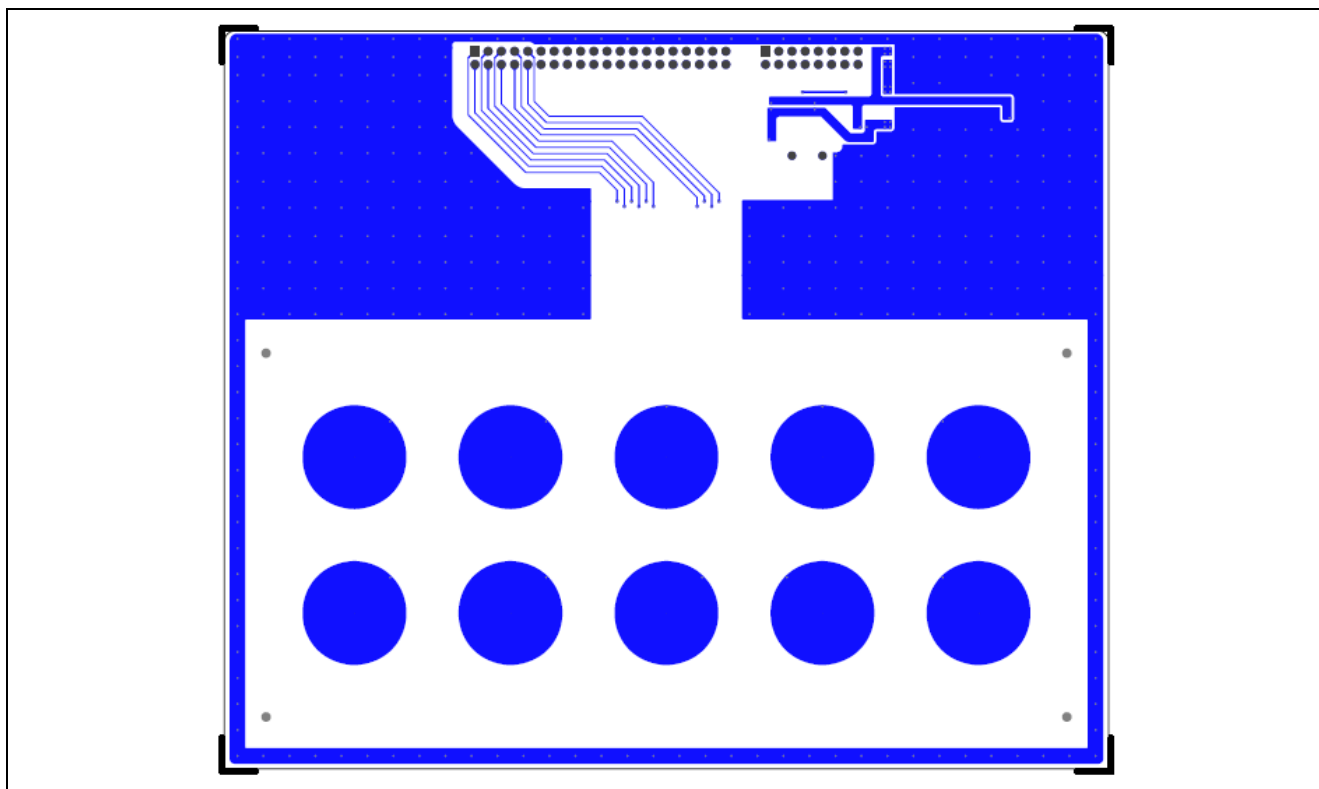


図 6-5 タッチ電極ボード 部品面パターン

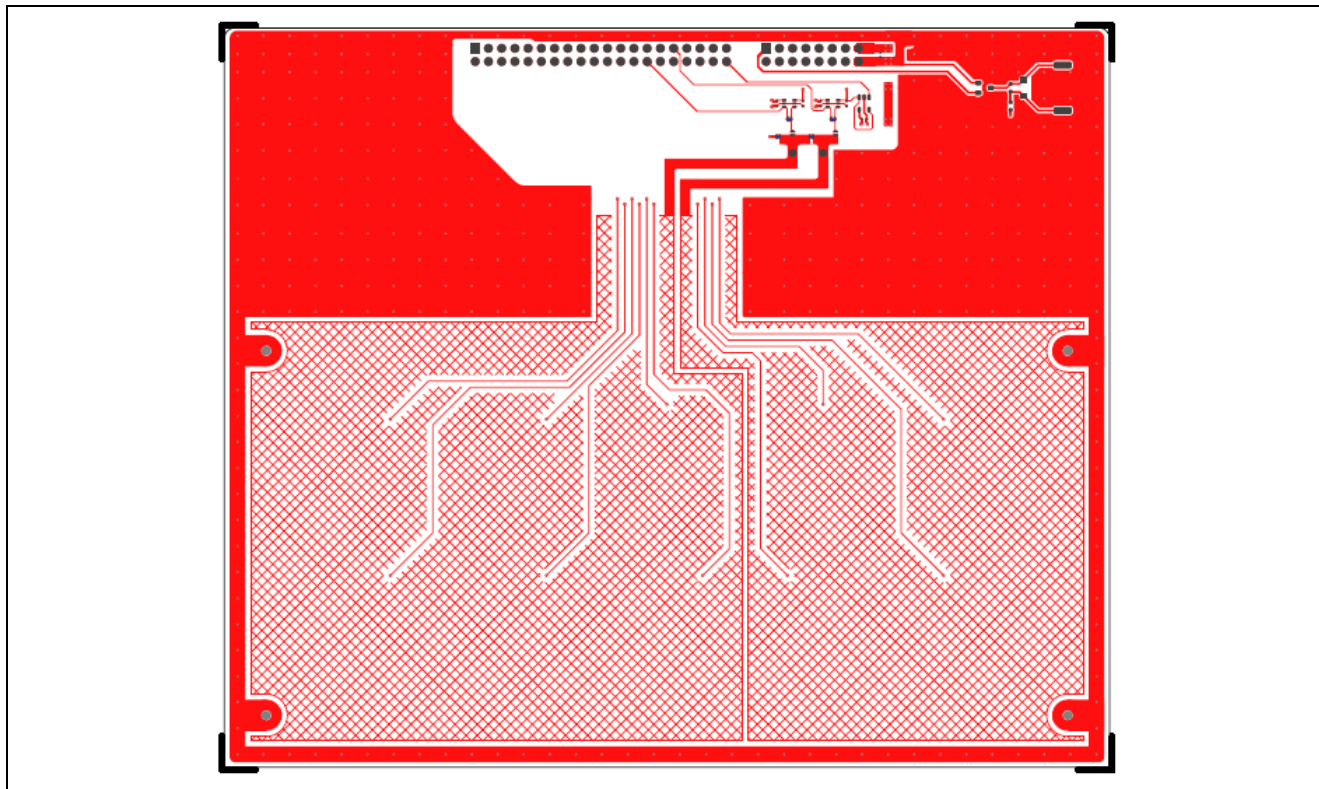


図 6-6 タッチ電極ボード 半田面パターン（部品面視）

6.1.3 PMOD OLEDrgb IF ボード

PMOD OLEDrgb IF ボードは CPU ボードと PMOD OLEDrgb を接続するためのボードです。

CPU ボードは CN3 に PMOD タイプ 6A (拡張 I2C) 対応の PMOD コネクタを搭載しています。PMOD OLEDrgb は PMOD タイプ 2A (拡張 SPI) 対応のため、本ボードで PMOD OLEDrgb と CPU の SPI 対応端子を接続しています。

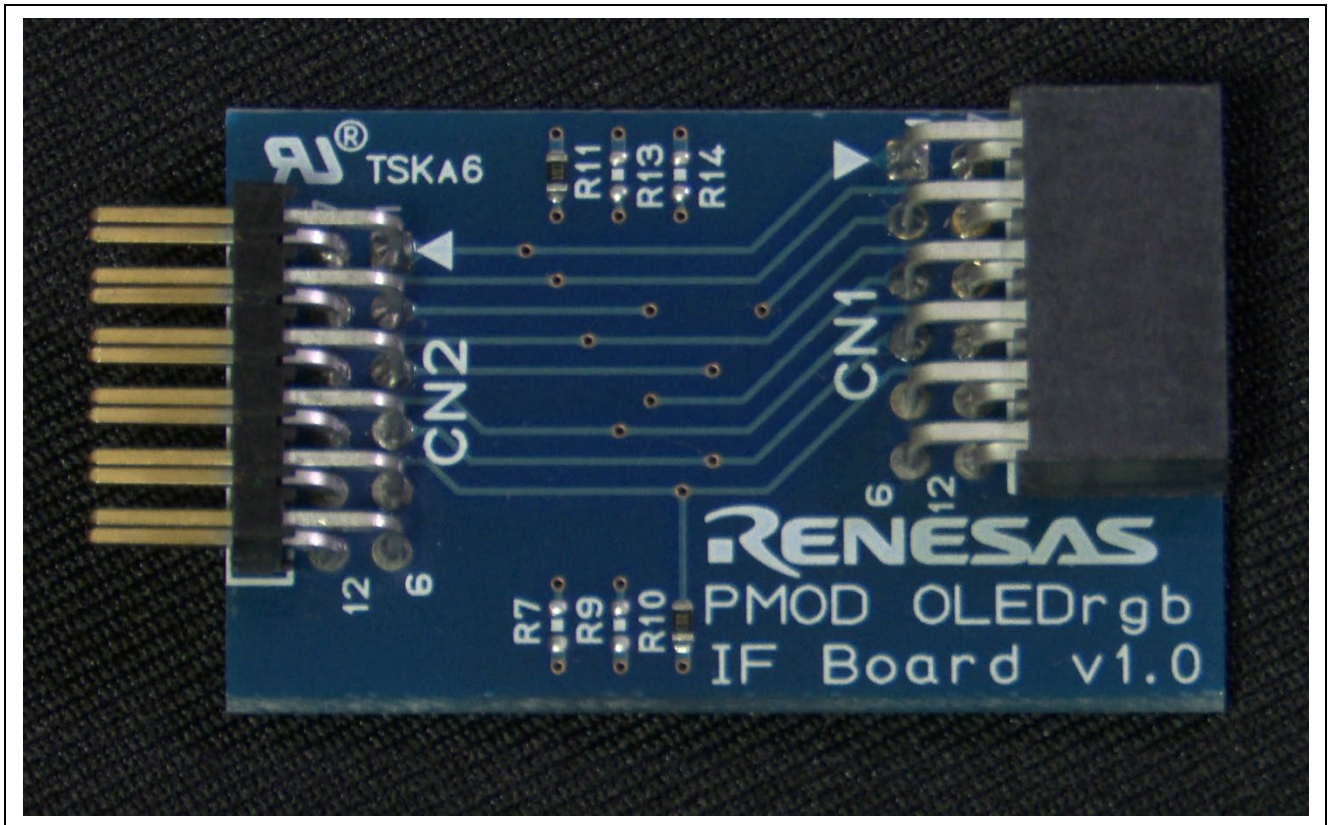


図 6-7 PMOD OLEDrgb IF ボードの外観

6.1.4 PMOD OLEDrgb (ディスプレイ)

PMOD OLEDrgb は Digilent 社製の PMOD タイプ 6A (拡張 I2C)対応のディスプレイモジュールです。本システムでは、押下されているボタンとその強度を表示します。

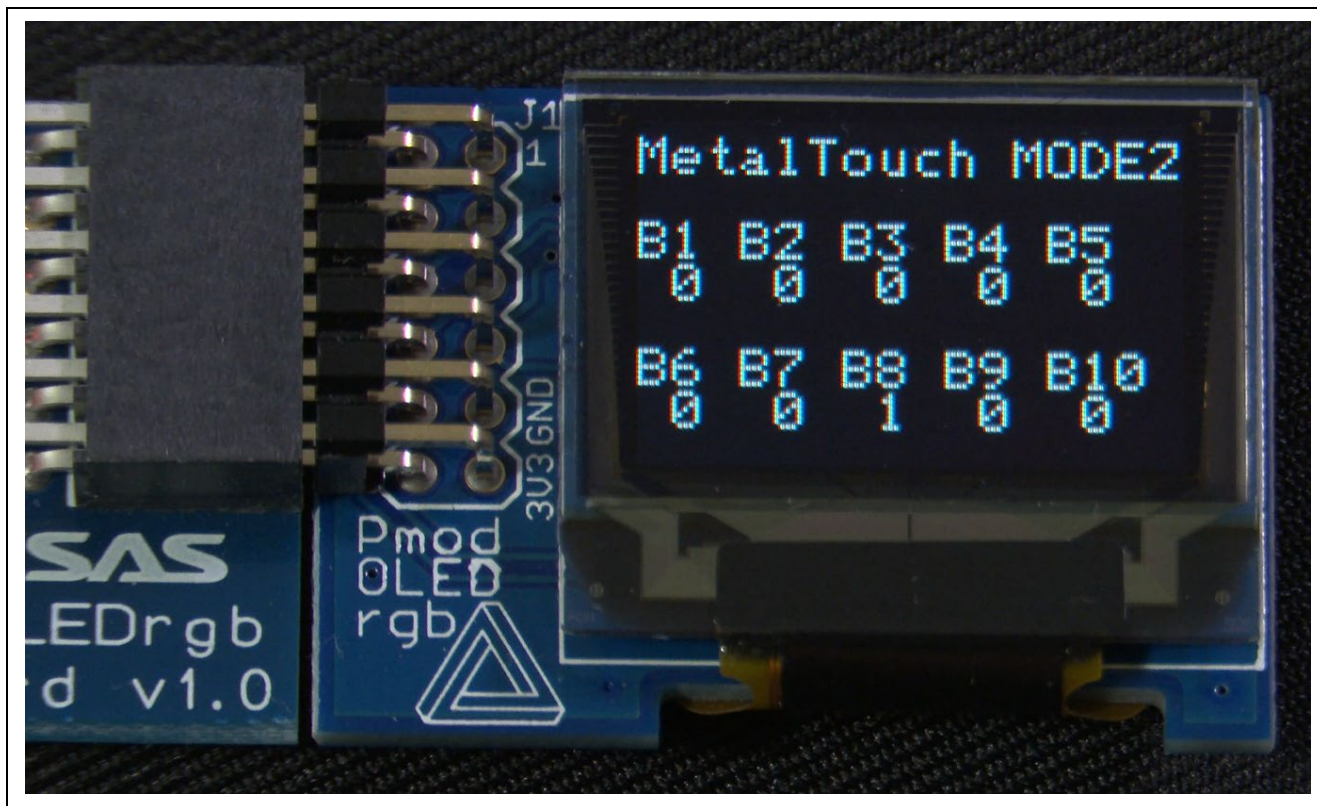


図 6-8 PMOD OLEDrgb の外観と表示例

6.2 機械構成

6.2.1 センサ部の構造

図 6-9 に金属パネルタッチセンサのセンサ部の構成を示します。金属パネルタッチセンサのセンサ部は主に3つの層で構成されます。第一層は導電性オーバーレイ（金属パネル）層で、一般的に金属板を使用します。第二層はスペーサ層で、第一層と第三層の距離を固定する役割を持ちます。第三層は基板層で、基板上に形成した電極と第一層の金属パネルとの間でコンデンサを形成します。

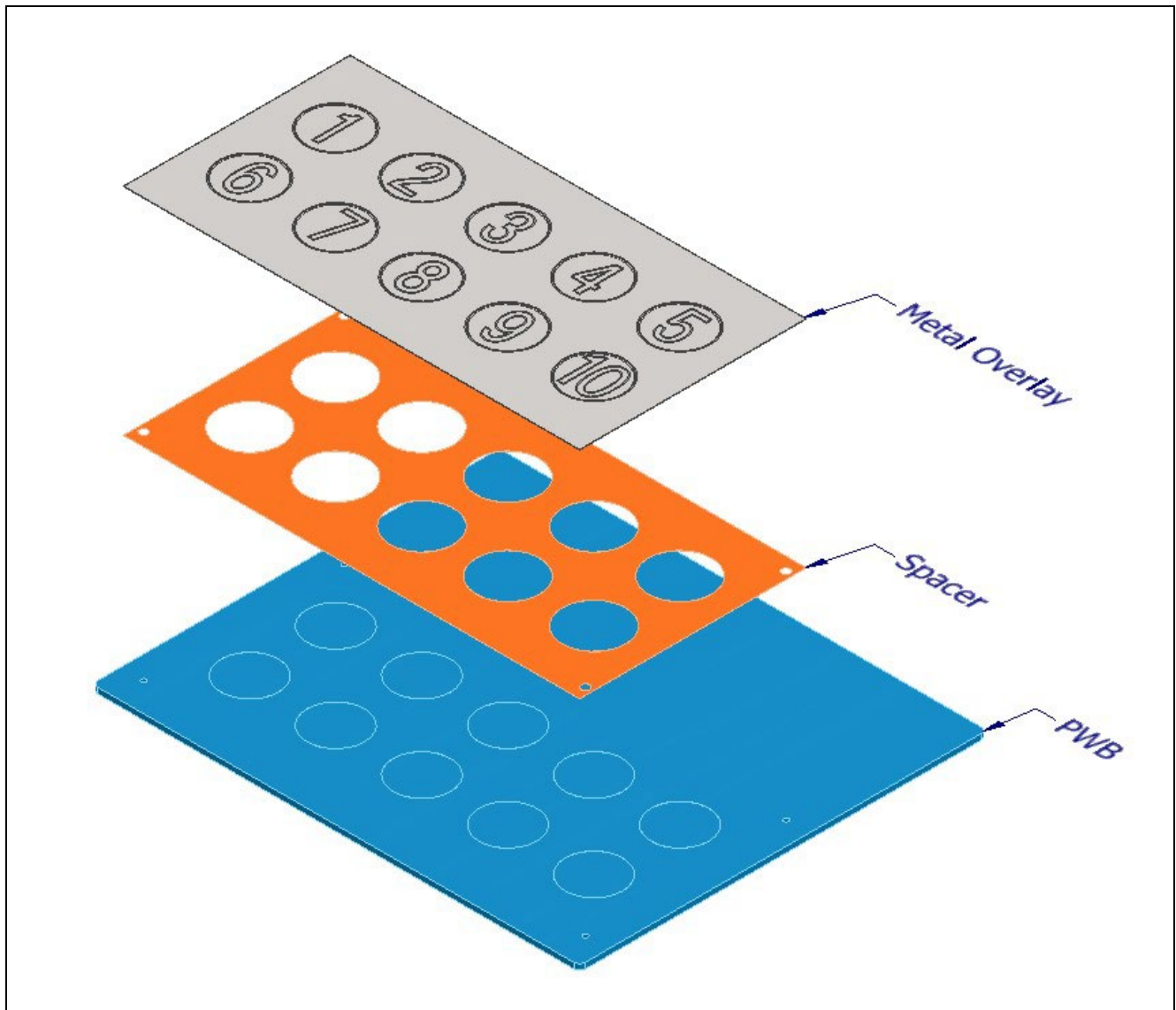


図 6-9 金属パネルタッチセンサ センサ部の構成

6.2.2 金属パネル層

金属パネルの材質はステンレスやアルミなどを使用します。金属パネル表面にはセンサの位置を示すためにレーザー加工、エッチング、印刷などの方法で印字を行います。金属パネルは基板 GND などの固定電位に接続してください。基板層の電極との電位が安定していない場合、タッチ判定の信頼性とノイズ耐性が大幅に低下します。本システムでは金属パネルの表側にエッチングによる印字を行っています。また、裏面では電極位置にあわせてエッチングで円状に掘り込みを行っています。エッチングにより以下の効果が期待できます。

- ・ 押下したボタン以外のボタンやボタン周辺に圧力がかかったときのたわみを抑え誤検知を抑制する
- ・ ボタン部の金属パネルの厚みを薄くすることによりパネルがたわみやすくなり感度が向上する

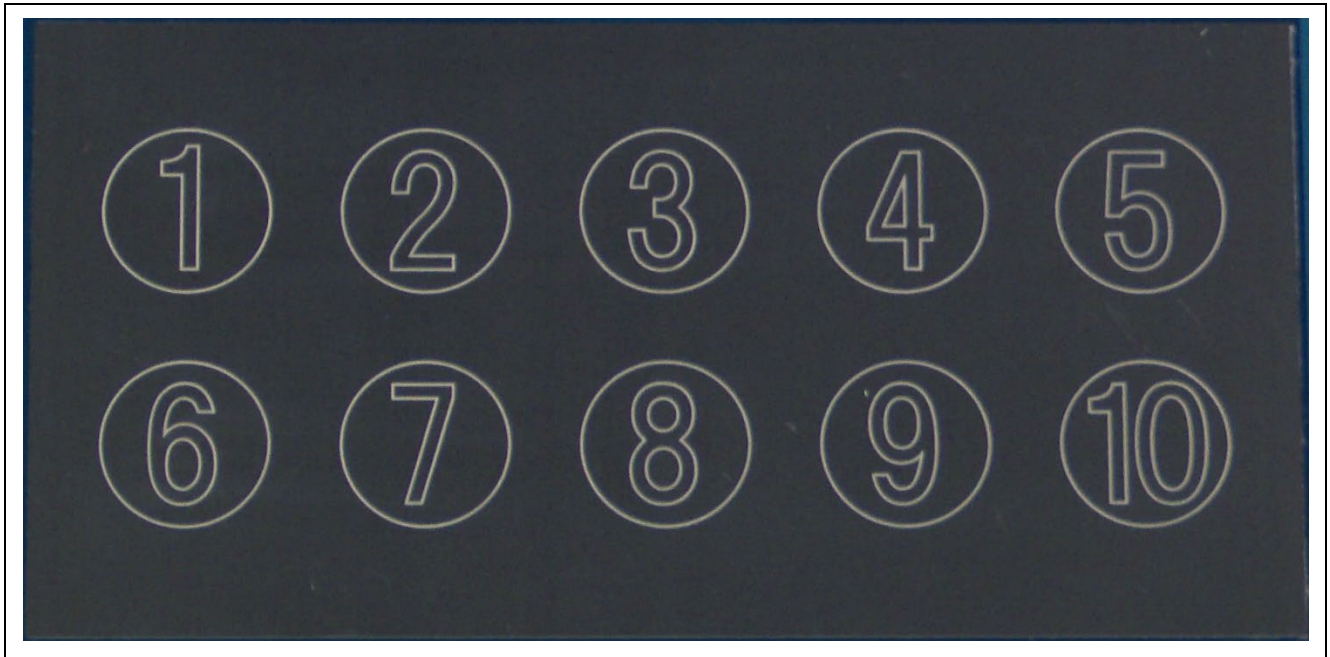


図 6-10 金属パネル層 外観（エッチングによる印字）

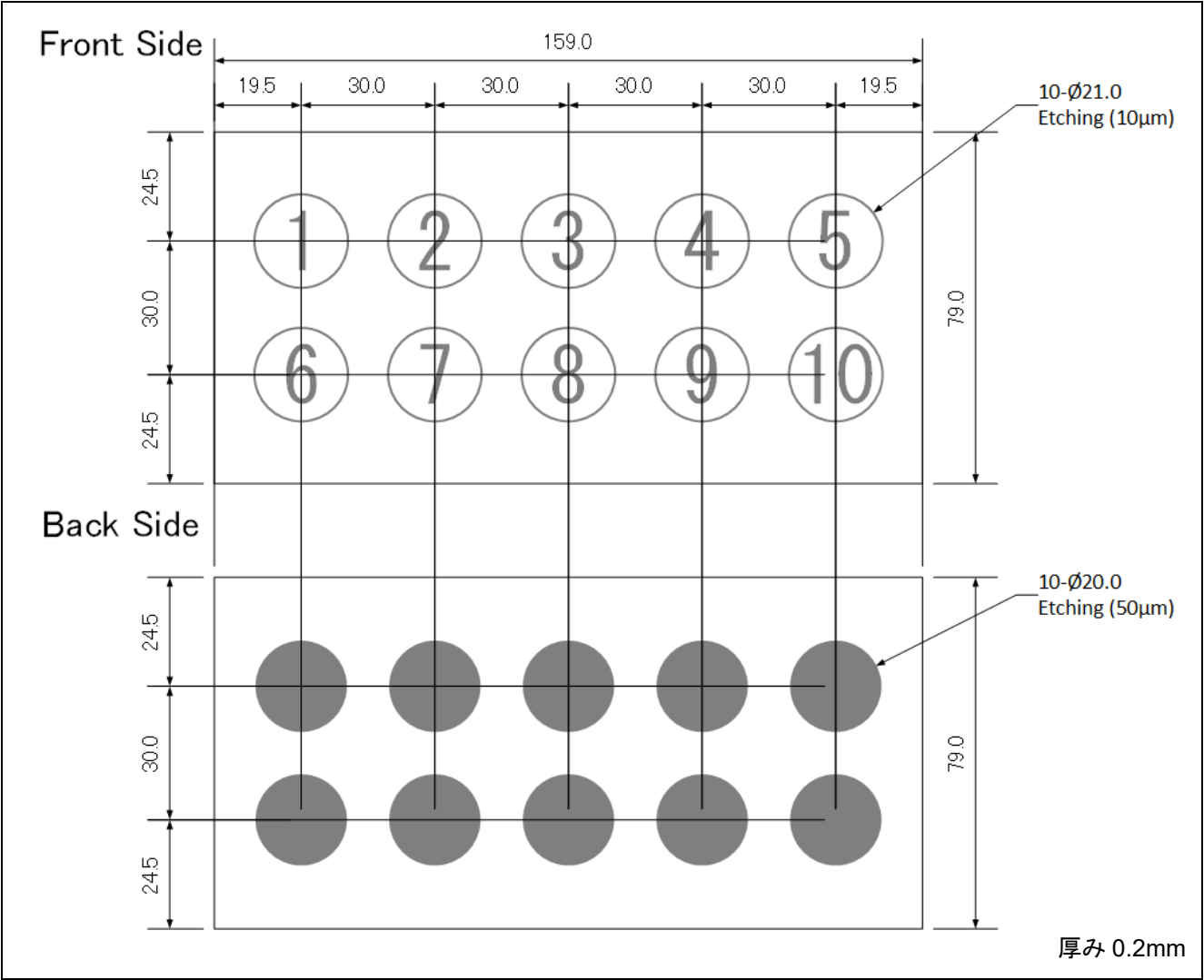


図 6-11 金属パネル層 寸法（単位：mm，特記部除く）

6.2.3 スペーサ層

スペーサ層は金属パネル層と基板層の間の距離を固定するために使用します。スペーサは一般に絶縁体のシートを使用します。スペーサはセンサの位置に合わせて穴を開けています。スペーサは金属パネル層を押したときに変形しない、堅牢な素材を使用してください。スペーサに使用する素材は、OCA（光学透明粘着）フィルム、ガラスエポキシ材（FR4）、アクリル樹脂等を推奨します。

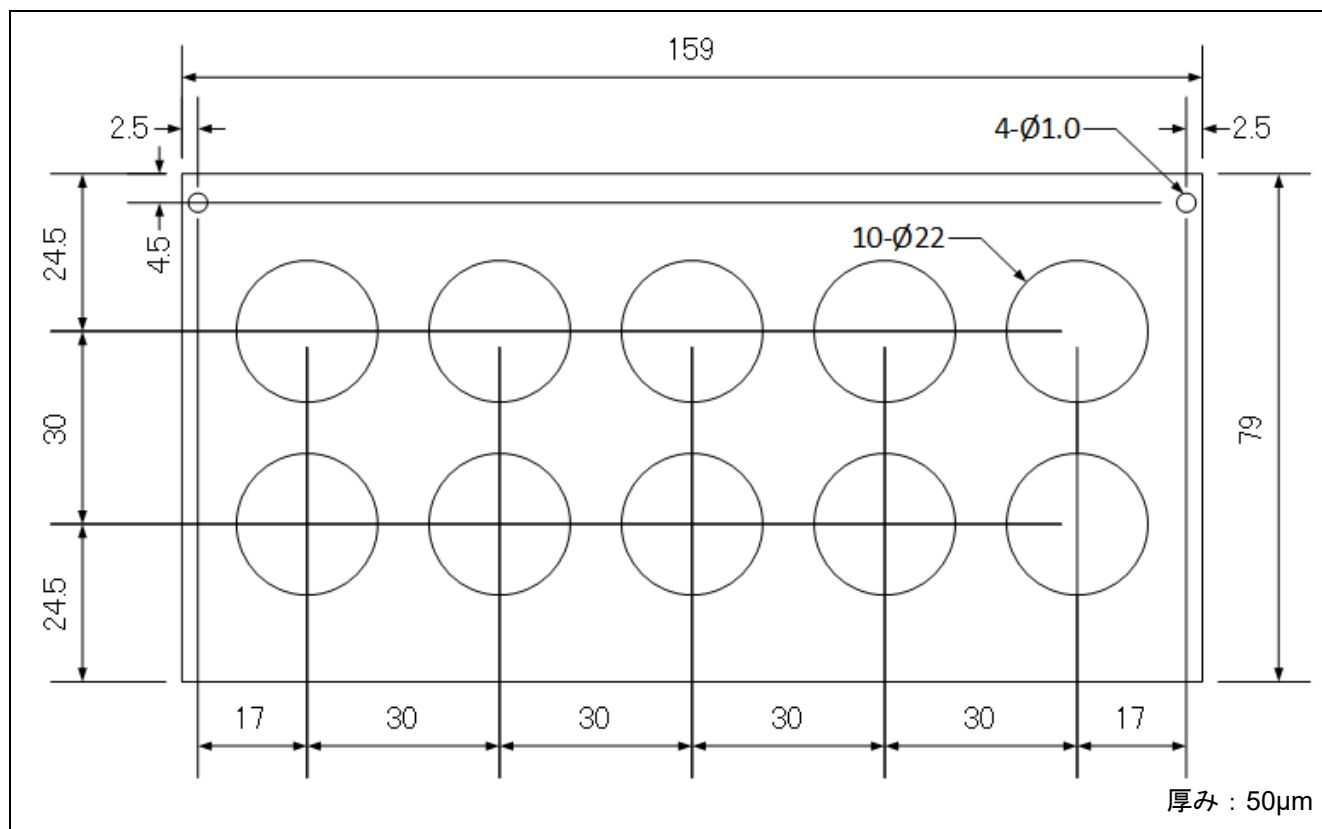


図 6-12 スペーサ層（単位：mm）

6.2.4 基板層

基板層はパターンにより形成した電極を持つ基板で、金属パネル層との間にコンデンサを形成します。

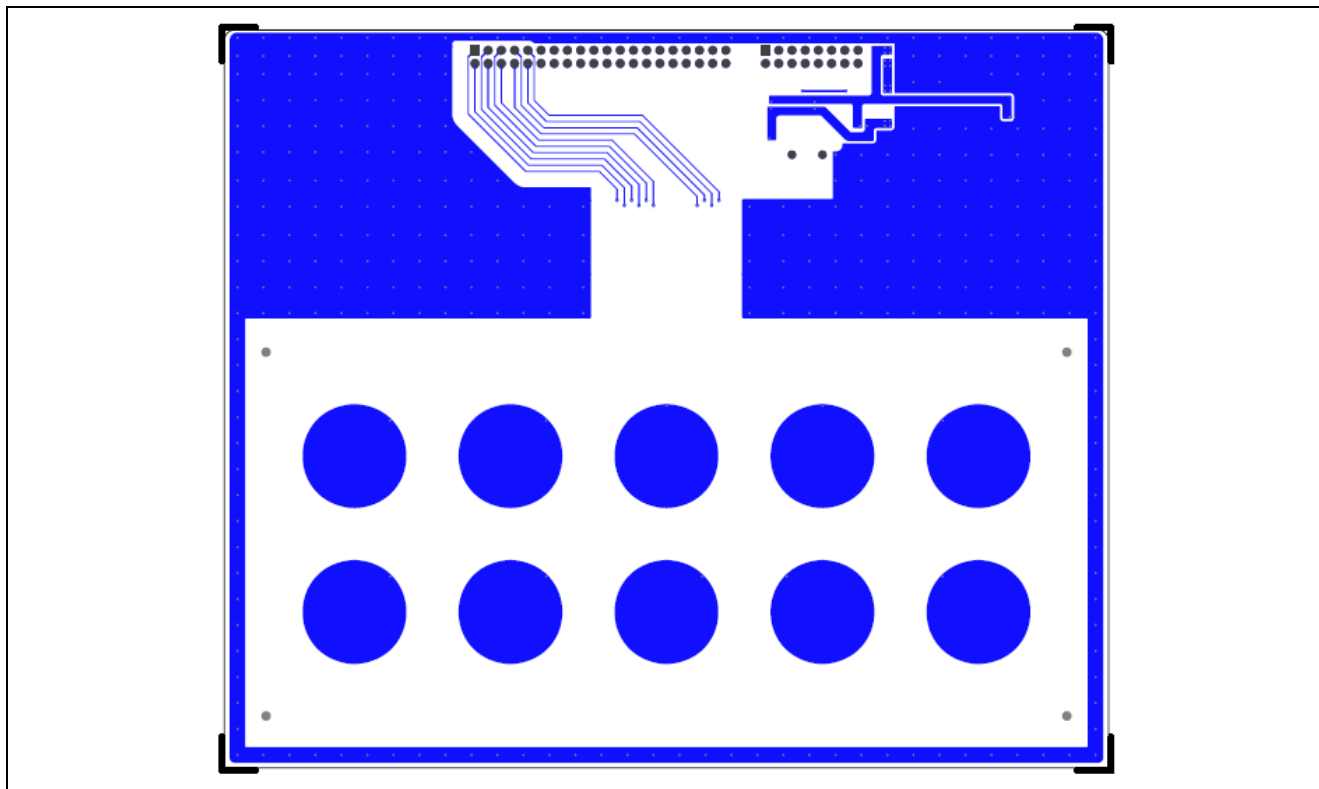


図 6-13 基板層

6.2.5 センサ設計

金属パネルタッチセンサの設計上の注意点を記載します。図 6-14 に金属パネルの構造を示します。

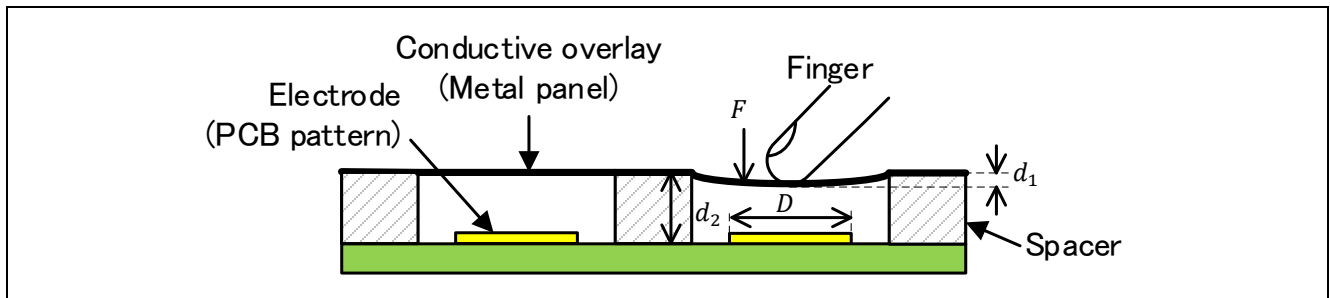


図 6-14 金属パネルタッチセンサの構造

ここで

F : 導電性オーバーレイを押し込む力 [N]

D : 電極の直径 [m]

d_1 : 応力による導電性オーバーレイたわみ[m]

d_2 : スペーサ厚 [m]

設計では検出したい最小の力と不可逆的な変形が起こらない最大の力 F を制約条件として、導電性オーバーレイとスペーサの材質を選択します。導電性オーバーレイの材質により、力 F に対するたわみ d_1 が決まります。次に、力に応じた静電容量変化を適切にするため、電極の直径 D とスペーサ厚 d_2 を決定します。

(1) 導電性オーバーレイの選択

金属パネルはユーザの押し込みによってたわむような十分な柔らかさを持つ必要があります。同時に、押し込みにより金属パネルに不可逆的な変化が生じないよう十分な強度を持つ必要があります。柔らかさを示す指標として、ひずみと応力の比例定数であるヤング率という特性が存在します。また強度を表す指標として、降伏強度という特性があります。一般的な金属材料のヤング率と降伏強度を表 6-1 に示します。

民生、産業用で使用される最も一般的な金属素材はステンレスとアルミニウムです。十分な降伏強度を有していれば、ヤング率が大きいほど押し込む力に対する金属のたわみが大きくなり、静電容量の変化が大きくなります。

表 6-1 金属材料のヤング率と降伏強度（参考値）

材質	ヤング率 [GPa]	降伏強度 [MPa]
ステンレス (SUS303, SUS304)	193	210
アルミニウム (A5052)	70	215
黄銅 (C2600)	110	130
チタン (TP340)	106	215

(2) スペーサの選択

スペーサの厚み d_2 により、金属パネルと基板層の電極との距離が決定します。これにより、力を加えていないときの静電容量 C_{sdef} [F]が決まります。 C_{sdef} は図 6-14 のパラメータを用いて以下の式で求めることができます。

$$C_{sdef} = \varepsilon_r \varepsilon_0 \frac{\pi D^2}{4d_2} \quad (1)$$

ε_r : 電極の間にある材質の比誘電率（空気として、 $\varepsilon_r \cong 1$ ）

ε_0 : 電気定数 ($\varepsilon_0 \approx 8.854 \times 10^{-12}$) [F/m]

(1)式より、比誘電率 $\varepsilon_r \cong 1$ 、直径 $D=20$ [mm]の円形の電極のときのスペーサ厚 d_2 と力を加えていないときの静電容量 C_{sdef} の関係は図 6-15 のようになります（参考値）。図 6-15 より、スペーサ厚が小さいほど、静電容量の変化量が大きくなるため、高い感度を得るためにはスペーサ厚を小さくする必要があります。ただし、変化が大きすぎる場合強度判定のしきい値の設定が難しくなるため、強度判定機能が必要な場合は極端にスペーサ厚を小さくしすぎないようにする必要があります。

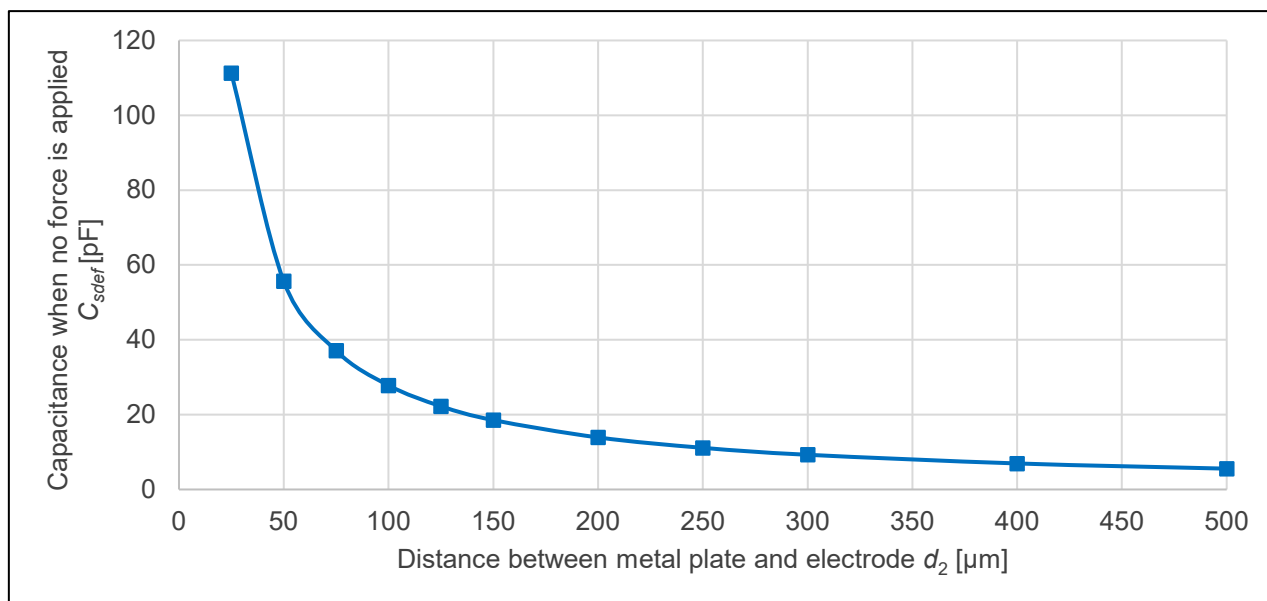


図 6-15 スペーサ厚 d_2 と静電容量 C_{sdef} の関係（参考値）

ボタン部を押し込んだときの静電容量変化率 ΔC_s [%]は以下の式で表すことができます。

$$\Delta C_s = \frac{C_{s_{push}} - C_{s_{def}}}{C_{s_{def}}} \times 100 \quad (2)$$

ここで、ボタン部を押し込んだときの静電容量 $C_{s_{push}}$ [F]は

$$C_{s_{push}} = \varepsilon_r \varepsilon_0 \frac{\pi D^2}{4(d_2 - d_1)} \quad (3)$$

で求めることができます。

(2)式より、比誘電率 $\varepsilon_r \cong 1$ 、電極直径 $D=20$ [mm]、スペーサ厚 $d_2 = 0.05$ mm のときのボタン部の押し込んだときのたわみ量 d_1 と静電容量変化率 ΔC_s との関係は図 6-16 のようになります（参考値）。

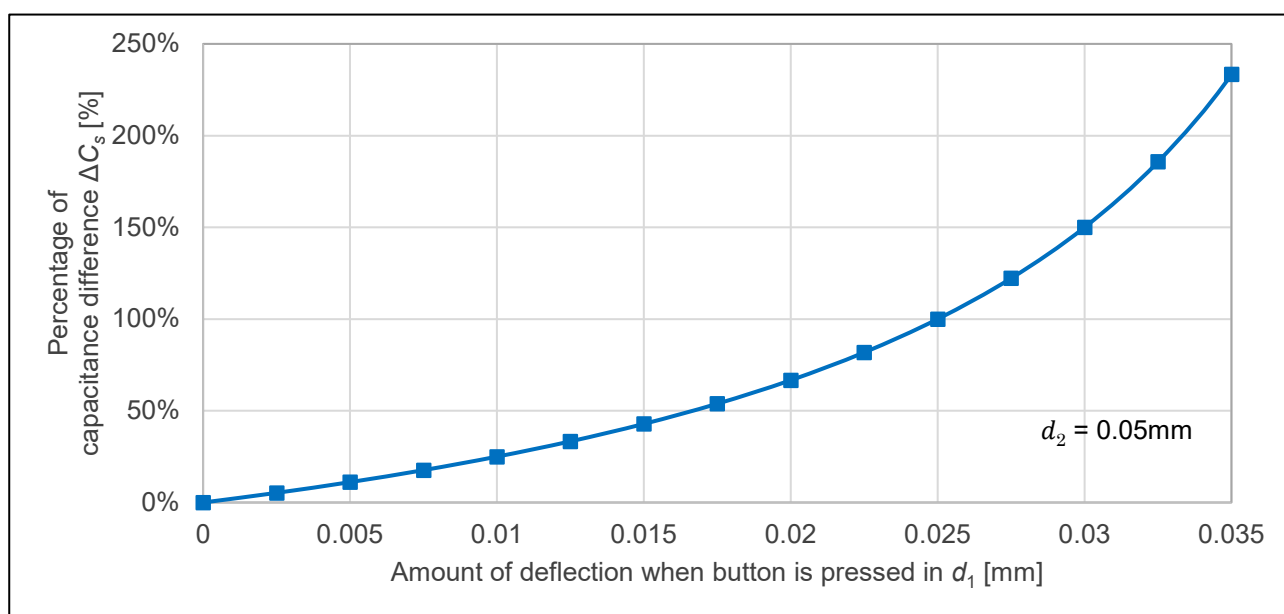


図 6-16 静電容量変化 ΔC_s と応力による金属パネルのたわみ d_1 の関係（参考値）

6.2.6 組立方法

金属パネル層、スペーサ層、基板層をそれぞれ重ね合わせ、固定する必要があります。各層の固定は一般的に接着材を使用します。接着剤は強固で弾力がないものを使用する必要があります。組立が適切でない場合、ボタン間のクロストークや、ボタン以外の箇所を押し込んだときに誤検知が発生する場合があります。以下に、適切な組立を行うための設計上の注意点を記載します。

(1) 接着剤やスペーサ層の剛性が不十分な場合、または貼り付けが不均一な場合

金属パネルと比較して接着剤やスペーサ層の剛性が不十分な場合や、貼り付けが不均一な場合、センサ部を押し込んだときにセンサ以外の箇所にも変形が生じます。これにより、誤検知が発生する可能性があります。これらの影響を避けるために適切な金属パネル、接着剤、スペーサ層の材質の選定を行ってください。また、貼り付けを均一に行うために基板のパターンの厚みを考慮し、基板の平坦な箇所に貼り付けることを推奨します。

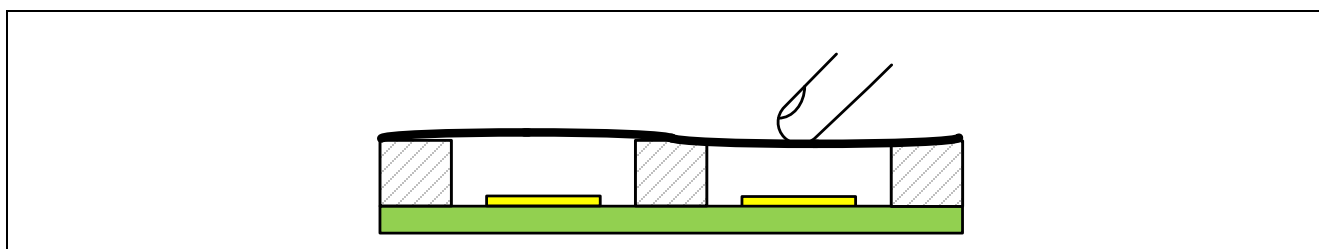


図 6-17 接着剤やスペーサ層の剛性が不十分、または貼り付けが不均一な場合

(2) 機構の剛性が不十分な場合

機構の剛性が不十分な場合、押し込みにより全体がたわみます。これにより、センサ部を押し込んだときにセンサ以外の箇所にも変形が生じ、誤検知が発生する可能性があります。この対策として、基板層の厚みを大きくするか、基板の半田側に基板の剛性を強化するための補強材を取り付けることを推奨します。寄生容量の増加を避けるため、補強材にはガラスエポキシ材やアクリル樹脂などの非導電性の材料を使用してください。

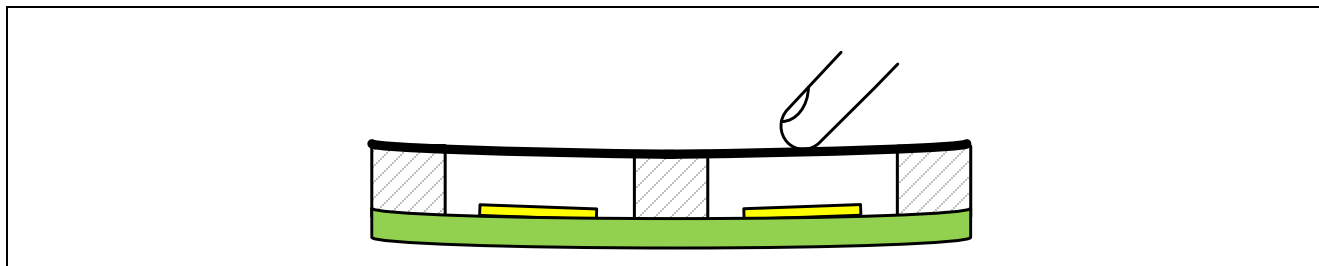


図 6-18 機構の剛性が不十分な場合

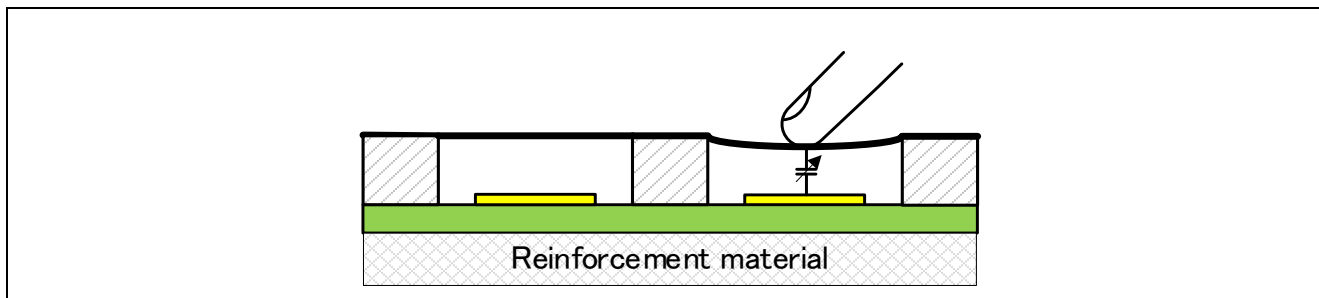


図 6-19 補強材の追加

7. サンプルプログラム

7.1 動作概要

図 7-1 に本サンプルプログラムの処理フローを示します。

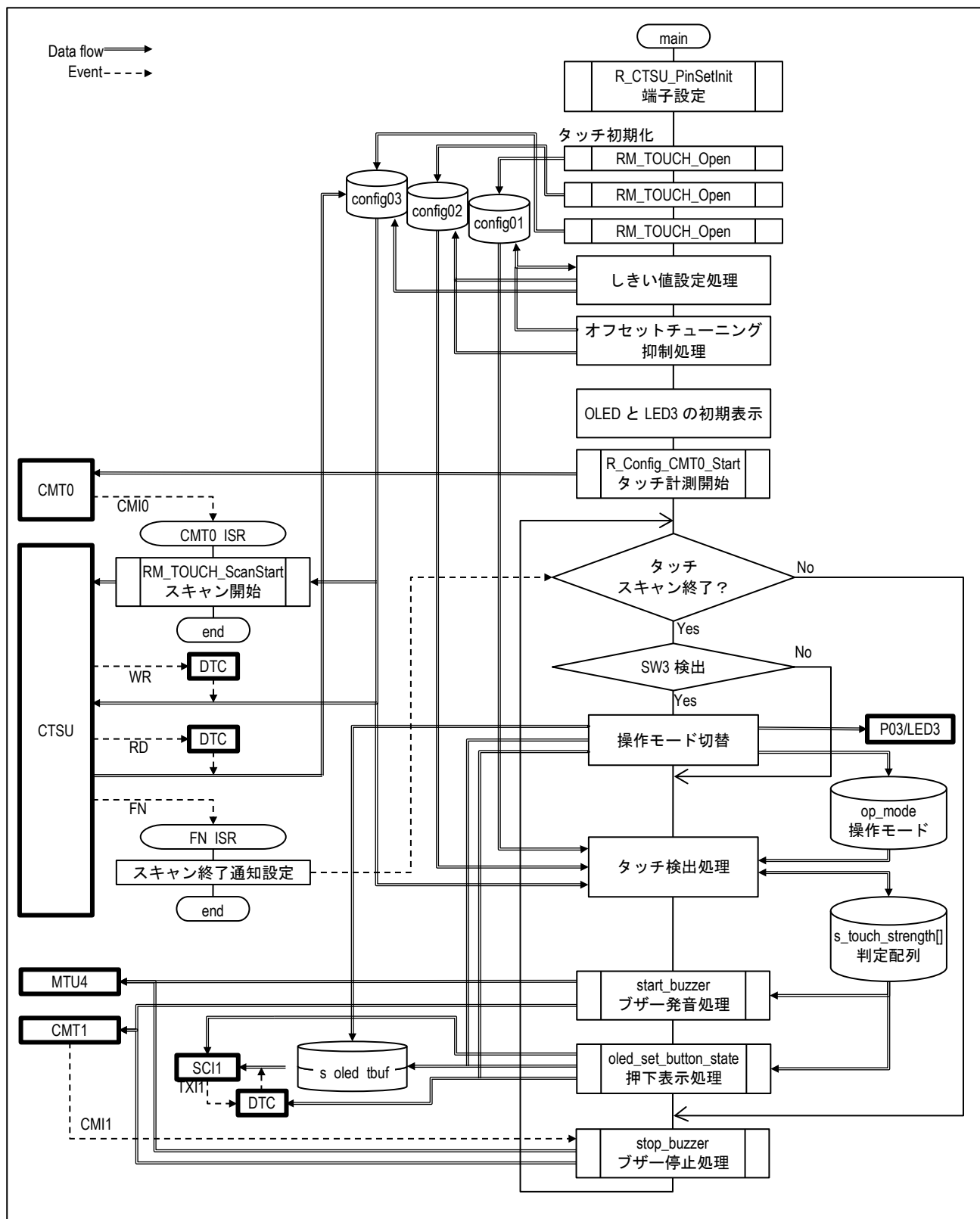


図 7-1 処理フロー概略

図 7-1 の各処理の概略は次の通りです。またタッチ計測処理と周辺機能動作のタイミングチャートを図 7-2 に示します。

初期設定

各機能について初期設定を行います。

- RM_TOUCH_Open を使用して「7.2.1.2 QE for Capacitive Touch の設定」で作成した config01,02,03 を初期設定
- 「表 7-19 usr_config.h 定義一覧」のしきい値定義に基づき config01,02,03 のタッチ検出しきい値を設定
詳細は「7.3.1 タッチ判定しきい値設定」参照
- config01,02 のオフセットチューニング抑制設定
- Pmod OLEDrgb の初期表示設定
- タッチ計測周期タイマのカウント開始

SW3 検出

SW3 の押下状態をポート状態の 3 回連続一致で確定し、押下検出時は操作モードを切り替え、LED3 と Pmod OLEDrgb の操作モード表示を更新します。

タッチ検出処理

タッチのスキャン終了を検知すると、操作モードによりタッチ判定結果を取得します。

- 操作モード 1: 押下判定、config03 の計測結果から各ボタンの押下判定
- 操作モード 2: 押下強度判定、config03 の計測結果を config01,02,03 の判定条件で各ボタンの押下強度を判定

各操作モードでの押下判定の詳細は「7.3.1 タッチ検出処理」を参照ください。

ブザー発音処理、ブザー停止処理

ブザー発音処理は、押下を検出ボタンに割り当てられた音階を MTU4 に設定して発音を行います。複数のボタン押下を同時検出した場合は番号の大きいボタンに対応する発音を行います。また、発音中に次のボタン押下を検出した場合は、改めて新たなボタンに対応するブザー発音を行います。発音設定時に発音時間を規定するタイマ CMT1 のカウント開始を行います

ブザー停止処理は CMT1 のコンペアマッチ CMI1 により発音時間経過を検出し、MTU4 と CMT1 の停止を行います。

詳細は「7.4 ブザー発音処理／停止処理」を参照ください。

押下表示処理

判定配列 s_data の各ボタンの押下状態を Pmod OLEDrgb に表示します。

詳細は「7.5 表示処理」を参照ください。

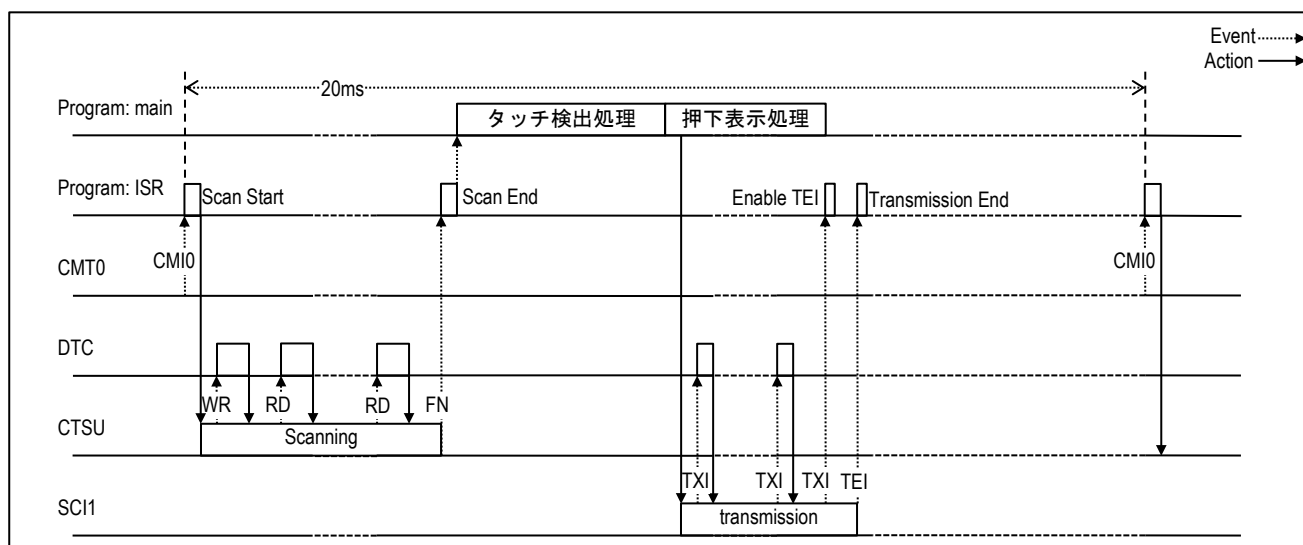


図 7-2 周辺機能タイミングチャート

7.2 使用する MCU 機能と設定

本例で使用する周辺機能一覧を表 7-1 に、使用端子設定一覧を表 7-2 に示します。また、クロック設定を表 7-3 に示します。未使用端子は出力 Low に設定しています。

周辺機能の設定は Smart Configurator の FIT モジュールまたはコード生成機能を用いて行います。各周辺機能の設定は以降に示します。

表 7-1 使用する周辺機能一覧

周辺機能	用途
CTSU	静電容量タッチ検出
CMT0	静電容量タッチ検出周期
DTC	CTSU 制御 SCI1 を使用した表示器通信
SCI1	表示器制御
PORT	表示器・SW3・LED3 制御
MTU4	ブザー発音
CMT1	ブザー発音制御

表 7-2 使用端子一覧

No.	端子名	I/O	用途
2	P03	O	LED3
14	P35	I	SW2 の検出（本プログラムでは未使用）
15	P34	I	SW3 の押下検出
16	TS0	-	電極ボード電極 Button02 の計測
17	TS1	-	電極ボード電極 Button07 の計測
18	TS2	-	電極ボード電極 Button01 の計測
19	TS3	-	電極ボード電極 Button03 の計測
20	TS4	-	電極ボード電極 Button06 の計測
21	P21	-	PMOD 8 ピン, Pmod OLEDrgb RES#
22	P20	I	PMOD 2 ピン
23	SCK1	O	PMOD 4 ピン, Pmod OLEDrgb SPI クロック出力
24	SMOSI1	O	PMOD 3 ピン, Pmod OLEDrgb SPI データ出力
25	TS5	-	電極ボード電極 Button08 の計測
26	TS6	-	電極ボード電極 Button04 の計測
27	P13	O	PMOD 1 ピン, Pmod OLEDrgb CS#
28	P12	O	PMOD 7 ピン, Pmod OLEDrgb D/C#
29	TS7	-	電極ボード電極 Button09 の計測
30	TS8	-	電極ボード電極 Button05 の計測
31	TS9	-	電極ボード電極 Button10 の計測
38	TSCAP	-	CTSU TSCAP 端子
62	MTIOC4C	O	電極ボードブザー
65	PD1	I	USB シリアル IC（本プログラムでは未使用）
66	PD0	O	
78	P07	O	PMOD9, PmodOLEDrgb VCCEN
80	P05	O	PMOD10, PmodOLEDrgb PMODEN

【注】 未使用端子は出力 0 に設定

表 7-3 クロック設定

項目	設定
使用クロック	HOCO クロック (48MHz)
SCKCR (FCLK)	x1 (48MHz)
SCKCR (ICLK)	x1 (48MHz)
SCKCR (PCLKB)	x1/2 (24MHz)
SCKCR (PCLKD)	x1 (48MHz)

7.2.1 静電容量タッチ検出

静電容量タッチ検出には以下を使用します。

- FIT QE Touch モジュール : rm_touch_qe
- FIT QE CTSU モジュール : r_ctsu_qe
- FIT DTC モジュール : r_dtc_rx

また、タッチ検出周期に CMT0 を使用します。

使用するモジュールと周辺機能の設定と QE for Capacitive Touch での設定を以降に示します。

7.2.1.1 使用するモジュールの設定

各モジュールの Smart Configurator での設定を以下に示します。

表 7-4 FIT QE Touch モジュール : rm_touch_qe の設定

項目	設定
Configuration	
Parameter check	Use system default
Support for QE monitoring using UART	Sensor monitor not used
Support for QE monitoring using UART	Serial tuning not used
Type of chattering suppression	TypeA: Counter if exceeded threshold is hold within hysteresis range

表 7-5 QE CTSU モジュール : r_ctsu_qe の設定

項目	設定
Configuration	
Parameter check	Use system default
Data transfer of INTCTSUWR and INTCTSURD	DTC
Select Automatic Judgement code	Disable
Interrupt level for INTCTSUWR	Level 2
Interrupt level for INTCTSURD	Level 2
Interrupt level for INTCTSUFN	Level 2
リソース : CTSU	
TSCAP 端子	使用する
TS0 端子	使用する
TS1 端子	
TS2 端子	
TS3 端子	
TS4 端子	
TS5 端子	
TS6 端子	
TS7 端子	
TS8 端子	
TS9 端子	
その他 TS 端子	使用しない

表 7-6 FIT DTC モジュール : r_dtc_rx の設定

項目	設定
Configuration	
Parameter check	Use system default
DTCER control	Clear all DTCER registers in R_DTC_Open()
Address mode	Full address mode
Transfer Data Read Skip	Enable transfer data read skip
DMAC FIT check	DMAC FIT module is not used with DTC FIT module
Sequence transfer	Sequence transfer not used

表 7-7 CMT0 の設定

項目		設定
クロック設定		PCLK/32
コンペアマッチ 設定	インターバル時間	20ms
	コンペアマッチ割り込み許可(CMI0)	許可
	複数の割込み(CMI0)を許可	許可しない
	優先順位	レベル 15 (最高)

7.2.1.2 QE for Capacitive Touch の設定

表 7-1 から表 7-3 に示す MCU 動作条件、使用する周辺機能、端子設定を Smart Configurator で行った後、QE for Capacitive Touch のワークフローに従ってタッチ計測の設定を行います。

(1) タッチインターフェース構成

電極ボードに合わせてタッチインターフェース構成を作成します。本例では図 7-3 に示すように電極を構成し、構成（メソッド）を同一電極群に対して 3 種作成します。

config03 はタッチ計測に使用し、操作モード 1 では押下判定に適用します。config01 と config02 は config03 と合わせて操作モード 2 で押下強度判定に使用します。

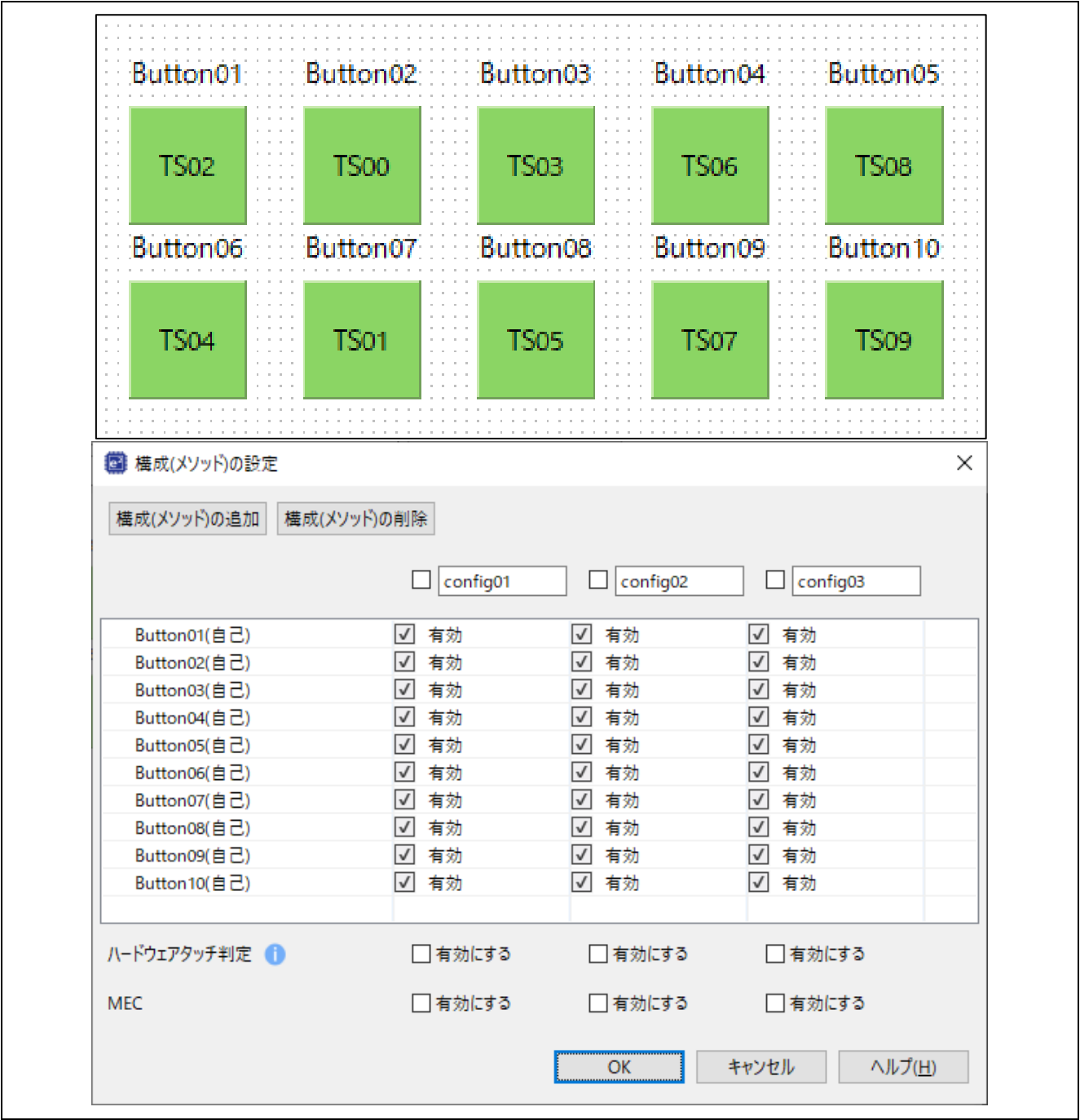


図 7-3 タッチインターフェース構成の電極構成

(2) タッチセンサの調整

各メソッドへの押下強度の割り当てを表 7-8 に示します。config01 に割り当てられた押下強度でタッチセンサの調整を行います。config02, config03 に対するタッチセンサの調整はプログラムで再設定するため押下強度によらず進めます。

タッチセンサの調整後、調整結果のファイルを「表 7-17 ソースファイル構成」のフォルダ「qe_gen」に出力し、プロジェクトを再ビルドします。

表 7-8 各メソッドの操作モード 2 の押下強度割り当て

メソッド	押下強度
config01	強
config02	中
config03	弱

7.2.2 表示器制御

Pmod OLEDrgb の制御には SCI1 を SPI クロック同期式モードのマスタ送信で、端子制御に PORT0,1,2 を使用し、SCI1 の送信データ設定には DTC の FIT モジュールを使用します。

SCI1 の設定を表 7-9 に、表 7-2 使用端子一覧に基づく PORT 設定を表 7-10 に示します。FIT DTC モジュールの設定は表 7-6 を参照下さい。

表 7-9 SCI1 の設定

SPI クロック同期式モード
マスタ送信

項目		設定
データ転送方向設定		MSB ファースト
データ反転設定		標準
転送速度設定	転送クロック	内部クロック
	ビットレート	6000 kbps
	モジュレーション機能を有効	無効
クロック設定		使用しない
データ処理設定	送信データ処理	DTC で処理する
割り込み設定	TXI1、TEI1 優先順位	レベル 15 (最高)
コールバック機能設定		使用しない

表 7-10 Pmod OLEDrgb 制御の PORT 設定

項目	設定					
ポート選択	PORT0		PORT1		PORT2	
使用ポート	P05	P07	P12	P13	P20	P21
	出力		出力	出力 1 を出力	入力 内蔵プルアップ	出力 1 を出力

7.2.3 ブザー発音

ブザー発音には MTU4 を PWM モード 1 で使用し、PWM 波形を PE1 から出力します。発音時間設定には CMT1 を使用します。

MTU4 の設定を表 7-11 に、発音時の音階設定を表 7-12 に、CMT1 の時間設定を表 7-13 に示します。

表 7-11 PWM モードタイマ 1(MTU4)の設定

項目		設定
同期動作設定		使用しない
TCNT4 カウンタ設定	カウンタクリア要因	TGR4 コンペアマッチ (TGRC4 を周期レジスタとして使用)
	カウンタクロックの選択	PCLK
ジェネラルレジスタの設定	TGRC4	アウトプットコンペアレジスタ
	TGRD4	アウトプットコンペアレジスタ
出力端子の設定	MTIOC4A 端子	端子出力は無効
	MTIOC4C 端子	端子初期出力は 0、コンペアマッチでトグル出力
	TGRD コンペアマッチ一致時の動作	MTIOC4C 端子からトグル出力
PWM 出力設定	PWM 周期	表 7-12 参照 初期値に音階 C5 を適用
	TGRA 初期値	
	TGRB 初期値	
	TGRC 初期値	
	TGRD 初期値	
A/D 変換トリガ設定		使用しない
割り込み設定		使用しない

表 7-12 MTU4 音階設定

index	音階	周波数 [Hz]	設定		
			PWM 周期 [ms]	TGRA, TGRC	TGRB, TGRD
0	A4	880	1.136363636	27272	13636
1	B4	987.7666025	1.012384907	24297	12148
2	C5	1046.502261	0.955564108	22933	11466
3	D5	1174.659072	0.851310839	20431	10215
4	E5	1318.510228	0.758431735	18202	9101
5	F5	1396.912926	0.715864233	17180	8590
6	G5	1567.981744	0.637762527	15306	7653
7	A5	1760	0.568181818	13636	6818
8	B5	1975.533205	0.506192453	12148	6074
9	C6	2093.004522	0.477782054	11466	5733

表 7-13 CMT1 の設定

項目		設定
クロック設定		PCLK/512
コンペアマッチ設定	インターバル時間	100ms
	コンペアマッチ割り込み許可(CMI1)	許可
	複数の割り込み(CMI1)を許可	許可しない
	優先順位	レベル 0 (割り込み禁止)

7.2.4 SW3 と LED3

操作モードの切り替えと表示に SW3 と LED3 を使用します。「表 7-2 使用端子一覧」に基づく PORT 設定を表 7-14 に示します。

表 7-14 SW3 と LED3 の PORT 設定

項目	設定	
ポート選択	PORT0	PORT3
使用ポート	P03	P34
	出力 1 を出力	入力

7.2.5 その他端子設定

本プログラムでは使用しない SW2 と USB シリアル IC の PORT 設定を表 7-15 に示します。

また、未使用端子は出力 Low に設定します。

表 7-15 SW2 と USB シリアル IC 接続の PORT 設定

項目	設定		
ポート選択	PORTD		PORT3
使用ポート	PD0	PD1	P35
	出力 CMOS 出力 1 を出力	入力	入力

7.3 タッチ検出処理

タッチの計測は config03 で行い、計測終了を検知するとタッチ検出処理を行います。タッチ判定はしきい値設定に基づきます。

7.3.1 タッチ判定しきい値設定

各押下強度のタッチ判定しきい値の設定は初期化処理で行います。QE for Capacitive Touch で行った config01 のタッチセンサの調整結果から、「表 7-19 usr_config.h 定義一覧」のタッチ判定しきい値割合に基づいてボタンごとに次のように設定します。

(1) 計測値変化のダイナミックレンジとヒステリシス割合の算出

タッチセンサの調整では、タッチ判定しきい値を調整時の計測値変化の 60% に設定しています。また、タッチ判定しきい値に対して 5% 下がった値をヒステリシス値としています。

タッチ判定しきい値とヒステリシス値は、config01 の場合は以下に格納されています。

```
g_ge_touch_instance_config01.p_ctrl->binfo.p_threshold[] // タッチ判定しきい値
g_ge_touch_instance_config01.p_ctrl->binfo.p_husteresis[] // ヒステリシス値
```

上記から、ボタン番号を n ($=0\sim9$) として、計測値のダイナミックレンジは次式で定義します。さらに、整数演算のために 2^{SCALE} のスケールを行います。演算結果を 32bit に収めるために SCALE を 9 とします。

$$\begin{aligned} \text{DR}[n] &= \frac{\text{p_threshold}[n]}{60\%} \\ \text{DR}[n] \cdot 2^{\text{SCALE}} &= \frac{100 \cdot \text{p_threshold}[n] \cdot 2^{\text{SCALE}}}{(100 \cdot 60\%)} \end{aligned}$$

また、(3)のヒステリシス算出へ向け、config01 の先頭ボタンのタッチ判定しきい値とヒステリシス値を取り出しておきます。

```
thr = g_ge_touch_instance_config01.p_ctrl->binfo.p_threshold[0];
hys = g_ge_touch_instance_config01.p_ctrl->binfo.p_hysteresis[0];
```

(2) タッチ判定しきい値の算出

求めた各ボタンの計測値変化のダイナミックレンジに対して、「表 7-19 usr_config.h 定義一覧」のタッチ判定しきい値割合に基づき、各 config のタッチしきい値を次式で算出します。m は config 番号を示します。

$$\begin{aligned} \text{config}_{0m}.\text{p_threshold}[n] &= \text{DR}[n] \cdot \text{しきい値割合}[m] \\ &= \left(\frac{(\text{DR}[n] \cdot 2^{\text{SCALE}}) \cdot \text{D_CFG_TOUCH_THRESHOLD}_m}{100} \right) 2^{-\text{SCALE}} \end{aligned}$$

(3) ヒステリシスの算出

(2)で求めたタッチ判定しきい値から、次式でヒステリシス値を求めます。

$$\begin{aligned} \text{config}_{0m}.\text{p_hysteresis}[n] &= \text{config}_{0m}.\text{p_threshold}[n] \frac{\text{hys}}{\text{thr}} \\ &= \frac{\text{config}_{0m}.\text{p_threshold}[n] \cdot \text{hys}}{\text{thr}} \end{aligned}$$

7.3.2 タッチ検出処理

タッチ検出は操作モードにより処理が異なります。

操作モード 1: 押下判定

config03 の計測結果からボタンごとに押下を判定し、押下の場合 1、そうでない場合は 0 を押下判定結果に格納します。

操作モード 2: 押下強度判定

config03 の計測結果をボタンごとに config03,02,01 の判定条件で押下強度を判定し、押下判定結果に格納します。各 config の押下強度の割り当ては表 7-8 を参照ください。

また、タッチ判定に同一のベースラインを適用するため、処理後に config03 のベースラインを config01,02 に適用します。

タッチ検出処理フローを図 7-4 に示します。

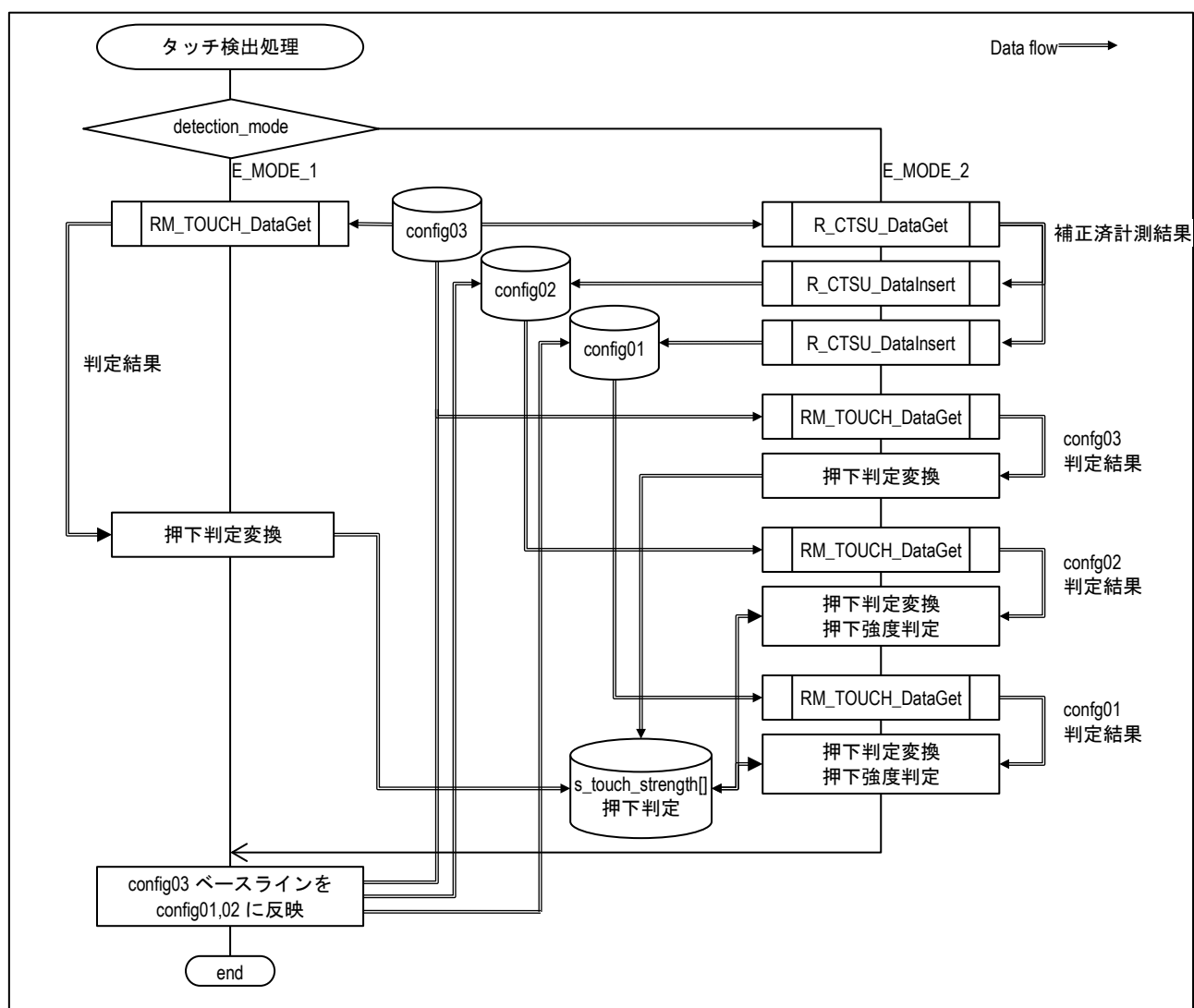


図 7-4 タッチ検出処理フローチャート

7.4 ブザー発音処理／停止処理

タッチ押下検出時に、検出したボタンに割り当てられた音階でブザー発音を行い、所定の発音時間経過後に発音を停止します。

複数のボタン押下を同時検出した場合は番号の大きいボタンに対応する発音を行います。また、発音中に次のボタン押下を検出した場合は、改めて新たなボタンに対応するブザー発音を行います。発音設定時はCMT1 をクリアしカウントをスタートします。

ブザー停止は、ブザー発音設定時に設定した CMT1 のコンペアマッチを検出した場合に行います。

図 7-5 に発音開始と停止のフローチャートを示します。

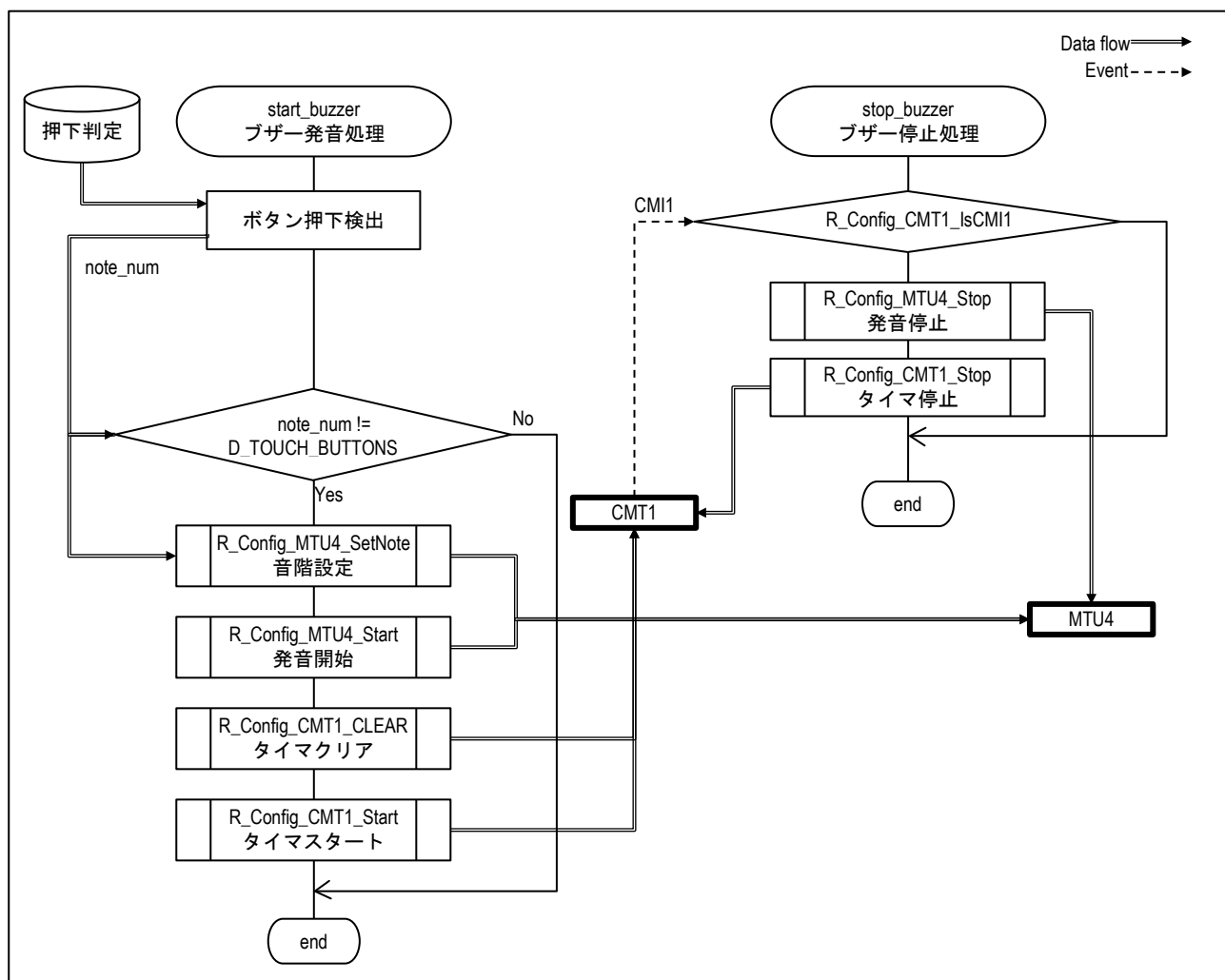


図 7-5 発音開始処理・停止処理フローチャート

7.5 表示処理

Pmod OLEDrgb への表示処理を行います。制御についての詳細は下記 Digilent 社サイトを参照ください。

[Pmod OLEDrgb Reference Manual - Digilent Reference](#)

本プログラムでは、図 7-6 に示すように、8 x 6 dot のフォントを使用して、表示器を 8 行 16 桁のキャラクタ表示で扱います。

押下状態表示のデータ送信タイミングチャートを図 7-7 に、本システムの Pmod OLEDrgb 初期設定を表 7-16 に示します。

		COL																
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
Connector Side	ROW 0	M	e	t	a	i	T	o	u	c	h		M	O	D	E	2	操作モード
	1																	
	2	B	1		B	2		B	3		B	4		B	5			ボタン番号
	3		0			0			0			0			0			押下状態
	4																	
	5	B	6		B	7		B	8		B	9		B	1	0		
	6		0			0			0			0			0			
	7																	

図 7-6 Pmod OLEDrgb 表示レイアウト

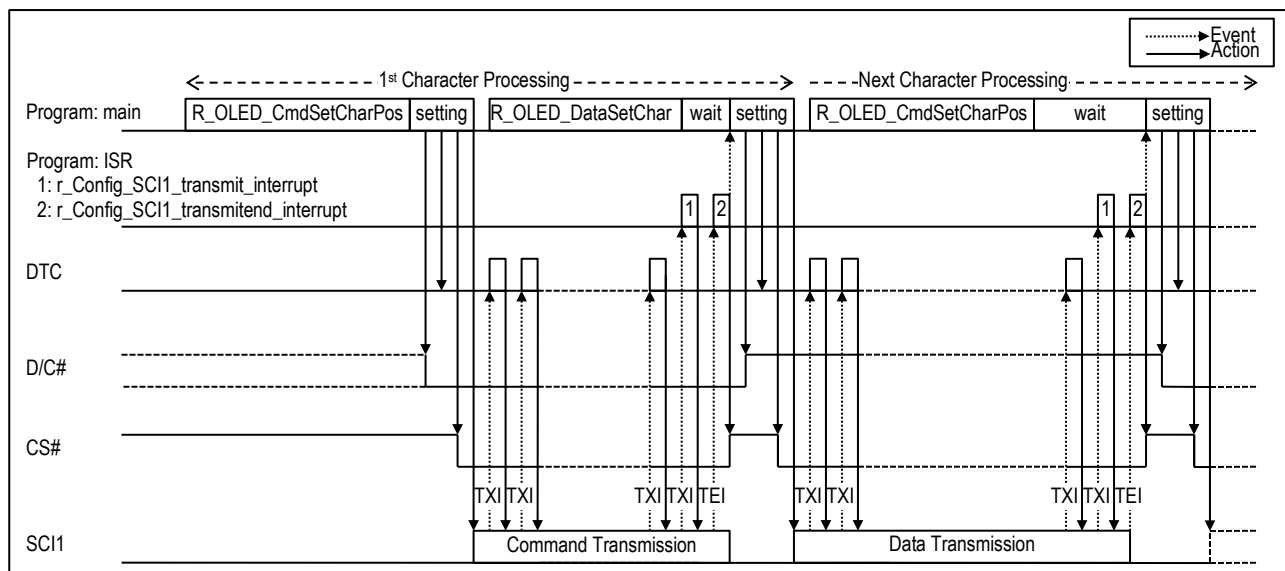


図 7-7 表示データ送信タイミングチャート

表 7-16 Pmod OLEDrgb 初期設定

Item	Setting	Command/Option
Remap & Color Depth setting	[0] Vertical address increment: 1 [1] RAM Column 0 to 95 maps to Pin Seg (SA,SB,SC) 95 to 0 :1 [2] Normal order SA,SB,SC (e.g. RGB): 0 [3] Disable left-right swapping on COM: 0 [4] Scan from COM [N-1] to COM0: 1 [5] Enable COM Split Odd Even: 1 [7:6] 256 color format: 0	A0h, 33h
Display Start Line by Row	0	A1h, 00h
Display Offset	0	A2h, 00h
Display Mode	Normal Display	A4h
Multiplex Ratio	62	A8h, 3Eh
Master Configuration	External V _{CC} supply	ADh, 8Eh
Power Save Mode	Disable Power save mode	B0h, 0Bh
Phase 1 and 2 period adjustment	[3:0] Phase 1 period in N DCLK: 1 [7:4] Phase 2 period in N DCLK: 3	B1h, 31h
Display Clock Divider / Oscillator Frequency	Define the divide ratio (D) of the display clocks (DCLK) [3:0]: 0 Fosc frequency increases as setting value increases [7:4]: 15	B3h, F0h
Second Pre-charge Speed for "A"	Second Pre-charge Speed Ranges.	8Ah, 64h
Second Pre-charge Speed for "B"		8Bh, 78h
Second Pre-charge Speed for "C"		8Ch, 64h
Pre-charge level	pre-charge voltage level [5:1]	BBh, 3Ah
V _{COMH}	0.83 x V _{CC}	BEh, 3Eh
Master Current Control	Master current attenuation factor [3:0]: 7/16	87h, 06h
Contrast for "A"	Contrast for all color in each segment.	81h, 91h
Contrast for "B"		82h, 50h
Contrast for "C"		83h, 7Dh
Deactivate scrolling	Deactivates the scrolling action.	2Eh
Clear Window	Column Address of Start: 0 Row Address of Start: 0 Column Address of End: 95 Row Address of End: 63	25h, 00h, 00h ,5Fh ,3Fh

7.6 プログラム構成

7.6.1 ソースファイル構成

表 7-17 ソースファイル構成

フォルダ名、ファイル名	説明
src	
└ main.c	メイン関数とサブルーチン
└ r_oled_api.h	Pmod OLEDrgb 送信コマンド/データ作成 API
└ r_oled_api.c	
└ usr_config.h	押下強度判定しきい値設定
└ smc_gen	Smart Configurator 生成
└ Config_CMT0	
└ Config_CMT1	
└ Config_MTU4	
└ Config_PORT	
└ Config_SCI1	
└ general	
└ rm_touch_qe	
└ r_bsp	
└ r_config	
└ r_ctsu_qe	
└ r_dtc_rx	
└ r_pincfg	
qe_gen	QE for Capacitive Touch 生成
└ qe_touch_config.c	
└ qe_touch_config.h	
└ qe_touch_define.h	

7.6.2 マクロ定義

表 7-18 main.c 定義一覧

定義名	初期値	説明
タッチ設定		
D_TOUCH_BUTTONS	10	ボタン数
D_TOUCH_STRENGTH_MAX	CTS_CFG_NUM_SELF_ELEMENTS / D_TOUCH_BUTTONS	ボタン押下強度数
D_TOUCH_THRESHOLD_QE	60	QE のタッチ判定しきい値割合 [%]
D_TOUCH_HYSTERESIS	5	QE のヒステリシス値割合 [%]
OLED 表示設定		
D_DISP_BUFFERBYTES	D_OLED_CHR_WIDTH * D_OLED_CHR_HEIGHT * D_OLED_COLS	Pmod OLEDrgb 送信データ バッファバイト数
D_DISP_TITLE_STR	"MetalTouch MODE"	タイトル文字列
D_DISP_TITLE_ROW	0	タイトルの先頭位置
D_DISP_TITLE_COL	0	
D_DISP_KEYNO_STR	"B1 "	ボタン番号文字列
D_DISP_KEYNO_ROW1	2	ボタン番号表示行 1
D_DISP_KEYNO_ROW2	5	ボタン番号表示行 2
D_DISP_KEYNO_COL	0	ボタン番号表示先頭列
D_DISP_KEYNO_LEN	3	ボタン番号文字列長
D_DISP_MODE_ROW	0	操作モード表示位置
D_DISP_MODE_COL	15	
D_DISP_KEY_ROW1	D_DISP_KEYNO_ROW1 + 1	検出結果表示行 1
D_DISP_KEY_ROW2	D_DISP_KEYNO_ROW2 + 1	検出結果表示行 2
D_DISP_KEY_COL	1	検出結果表示の先頭列
D_DISP_KEY_LEN	3	検出結果表示列長

表 7-19 usr_config.h 定義一覧

定義名	初期値	説明
D_CFG_TOUCH_THRESHOLD1	75	各押下強度のタッチ判定しきい値割合 [%] 番号は config 番号に対応
D_CFG_TOUCH_THRESHOLD2	50	
D_CFG_TOUCH_THRESHOLD3	25	
D_CFG_TOUCH_SCAN_INDEX	2	main.c で宣言した gp_touch_instance[] のスキャン対象とする要素のインデックス

表 7-20 r_oled_api.h 定義一覧

定義名	初期値	説明
D_OLED_COLORS	256	色深度（固定）
D_OLED_DOT_WIDTH	96	Pmod OLEDrgb 横ドット数
D_OLED_DOT_HEIGHT	64	Pmod OLEDrgb 縦ドット数
D_OLED_CHR_WIDTH	6	キャラクタ 横ドット数
D_OLED_CHR_HEIGHT	8	キャラクタ 縦ドット数
D_OLED_ROWS	D_OLED_DOT_HEIGHT / D_OLED_CHR_HEIGHT	表示行数
D_OLED_COLS	D_OLED_DOT_WIDTH / D_OLED_CHR_WIDTH	表示列数

7.6.3 構造体・共用体・列挙型

表 7-21 main.c 一覧

列挙型名	e_mode_t		
説明	操作モード		
メンバ	名称	値	説明
	E_MODE_1	0	操作モード 1
	E_MODE_2	1	操作モード 2

表 7-22 r_oled_api.h

共用体型名	u_oled_rgb_t		
説明	表示するキャラクタ・文字列の 8bit 色指定		
メンバ	型	名称	説明
	uint8_t	color	8bit 色指定
	struct	rgb	RGB 指定
	uint8_t:2	r	赤
	uint8_t:3	g	緑
	uint8_t:3	b	青

7.6.4 関数一覧

表 7-23 main.c

関数名	main			
説明	main 関数			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void		--	
関数名	static oled_init			
説明	Pmod OLEDrgb の初期設定			
引数	I/O	型	名称	説明
	I/O	uint8_t *	pbuf	送信するコマンド群の格納先ポインタ
戻り値	uint8_t *		使用可能な送信データ格納先ポインタ	
関数名	static oled_set_operation_mode			
説明	Pmod OLEDrgb に操作モードを表示			
引数	I/O	型	名称	説明
	I	e_mode_t	mode	操作モード
	I/O	uint8_t *	pbuf	送信するコマンド群の格納先ポインタ
戻り値	uint8_t *		使用可能な送信データ格納先ポインタ	
関数名	static oled_set_button_state			
説明	Pmod OLEDrgb にタッチ押下状態を表示			
引数	I/O	型	名称	説明
	I	uint8_t	data[]	押下状態配列
	I/O	uint8_t *	pbuf	送信するコマンド群の格納先ポインタ
戻り値	uint8_t *		使用可能な送信データ格納先ポインタ	
関数名	static start_buzzer			
説明	押下を検出したボタンのブザー発音処理			
引数	I/O	型	名称	説明
	I	uint8_t	data[]	ボタン押下強度配列
戻り値	void		-	
関数名	static stop_buzzer			
説明	ブザー発音停止処理			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void		-	
関数名	static set_transmissiondata_dtc_tx1			
説明	OLED へ送信する DTC 設定			
引数	I/O	型	名称	説明
	I	const void *	src_addr	送信データを格納したバッファのアドレス
	I	uint32_t	count	送信データのバイト数
戻り値	void		-	

表 7-24 r_oled_api

関数名	R_OLED_CmdInit			
説明	PmodOLEDRgb の初期設定を行うコマンド群を作成			
引数	I/O	型	名称	説明
	I	uint8_t *	buf	送信するコマンド群の格納先ポインタ
戻り値		int32_t	buf に格納したコマンド群のバイト数	
関数名	R_OLED_CmdDisplay			
説明	PmodOLEDRgb の Display On/Off コマンドを作成			
引数	I/O	型	名称	説明
	I	bool	on	true: Display On false: Display Off
	I	uint8_t *	buf	送信するコマンドの格納先ポインタ
戻り値		int32_t	buf に格納したコマンドのバイト数	
関数名	R_OLED_CmdSetCharPos			
説明	キャラクタの表示開始位置コマンドの作成			
引数	I/O	型	名称	説明
	I	uint8_t	row	行指定 : 0～15
	I	uint8_t	col	列指定 : 0～7
	I	uint8_t *	buf	送信するコマンドの格納先ポインタ
戻り値		int32_t	buf に格納したコマンドのバイト数	
関数名	R_OLED_DataSetChar			
説明	PmodOLEDRgb に表示する文字データの作成			
引数	I/O	型	名称	説明
	I	uint8_t	code	表示するキャラクタの ASCII Code
	I	u_oled_rgb_t	color	色指定
	I	uint8_t *	buf	送信するデータの格納先ポインタ
戻り値		int32_t	buf に格納したデータのバイト数	
関数名	R_OLED_DataSetStr			
説明	PmodOLEDRgb に表示する文字列データの作成			
引数	I/O	型	名称	説明
	I	const uint8_t *	code	表示する ASCII 文字列へのポインタ
	I	uint8_t	len	文字列長 (NULL 終端除く)
	I	u_oled_rgb_t	color	色指定
	I	uint8_t *	buf	送信するデータの格納先ポインタ
戻り値		int32_t	buf に格納したデータのバイト数	

表 7-25 Config_MTU4 ユーザ定義関数

関数名	R_Config_MTU4_SetNote			
説明	ブザー発音の音階設定			
引数	I/O	型	名称	説明
	-	uint32_t	note	音階を示す index (表 7-12 参照)
戻り値		void	-	

表 7-26 Config_CMT0_user.c 割り込み処理関数

関数名	r_Config_CMT0_cmi0_interrupt			
説明	CMI0 を受けて、タッチスキャン開始			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void			

表 7-27 Config_CMT1 ユーザ定義関数

関数名	R_Config_CMT1_CLEAR			
説明	カウント値のクリア（マクロ関数）			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void			
関数名	R_Config_CMT1_IsCMI1			
説明	CMT1 コンペアマッチの取得			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	bool true: コンペアマッチ検出 false: 非検出			

表 7-28 Config_PORT ユーザ定義関数 (1/2)

関数名	R_Config_PORT_GetSwitchState			
説明	SW3 の押下検出			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	bool true: 押下 false: その他			
関数名	R_Config_PORT_OLED_SET_CS			
説明	Pmod OLEDrgb CS#のアサート（マクロ関数）			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void			
関数名	R_Config_PORT_OLED_CLEAR_CS			
説明	Pmod OLEDrgb CS#のネゲート（マクロ関数）			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void			
関数名	R_Config_PORT_OLED_SET_DC_DATA			
説明	Pmod OLEDrgb D/C#のデータ送信選択（マクロ関数）			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void			

表 7-29 Config_PORT ユーザ定義関数 (2/2)

関数名	R_Config_PORT_OLED_SET_DC_CMD			
説明	Pmod OLEDrgb D/C#のコマンド送信選択（マクロ関数）			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void		-	
関数名	R_Config_PORT_OLED_SET_RES			
説明	Pmod OLEDrgb RST#のアサート（マクロ関数）			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void		-	
関数名	R_Config_PORT_OLED_CLEAR_RES			
説明	Pmod OLEDrgb RST#のネゲート（マクロ関数）			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void		-	
関数名	R_Config_PORT_OLED_SET_VCCEN			
説明	Pmod OLEDrgb VCCN のアサート（マクロ関数）			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void		-	
関数名	R_Config_PORT_OLED_SET_PMODEN			
説明	Pmod OLEDrgb PMODEN のアサート（マクロ関数）			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void		-	
関数名	R_PORT_OLED_CS			
説明	Pmod OLEDrgb CS#の状態取得（マクロ関数）			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	bool		false: アサート true: ネゲート	
関数名	R_Config_PORT_SET_LED3			
説明	LED3 の設定（マクロ関数）			
引数	I/O	型	名称	説明
	-	int32_t	a	0: 消灯 1: 点灯
戻り値	void		-	

表 7-30 Config_SCI1_user.c 割り込み処理関数

関数名	r_Config_SCI1_transmit_interrupt			
説明	TXI1 を受けて、TXI1 割り込みを禁止、TEI1 割り込みを有効に設定			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void		-	

関数名	r_Config_SCI1_transmitend_interrupt			
説明	TEI1 を受けて、Pmod OLEDogb CS#をネゲート、TEI1 割り込みを禁止、送信停止に設定			
引数	I/O	型	名称	説明
	-	void	-	-
戻り値	void		-	

8. プロジェクトをインポートする方法

サンプルコードは e² studio のプロジェクト形式で提供しています。本章では、e² studio および CS+へプロジェクトをインポートする方法を示します。インポート完了後、ビルドおよびデバッグの設定を確認してください。

8.1 e² studio での手順

e² studio でご使用になる際は、下記の手順で e² studio にインポートしてください。

(使用する e² studio のバージョンによっては画面が異なる場合があります。)

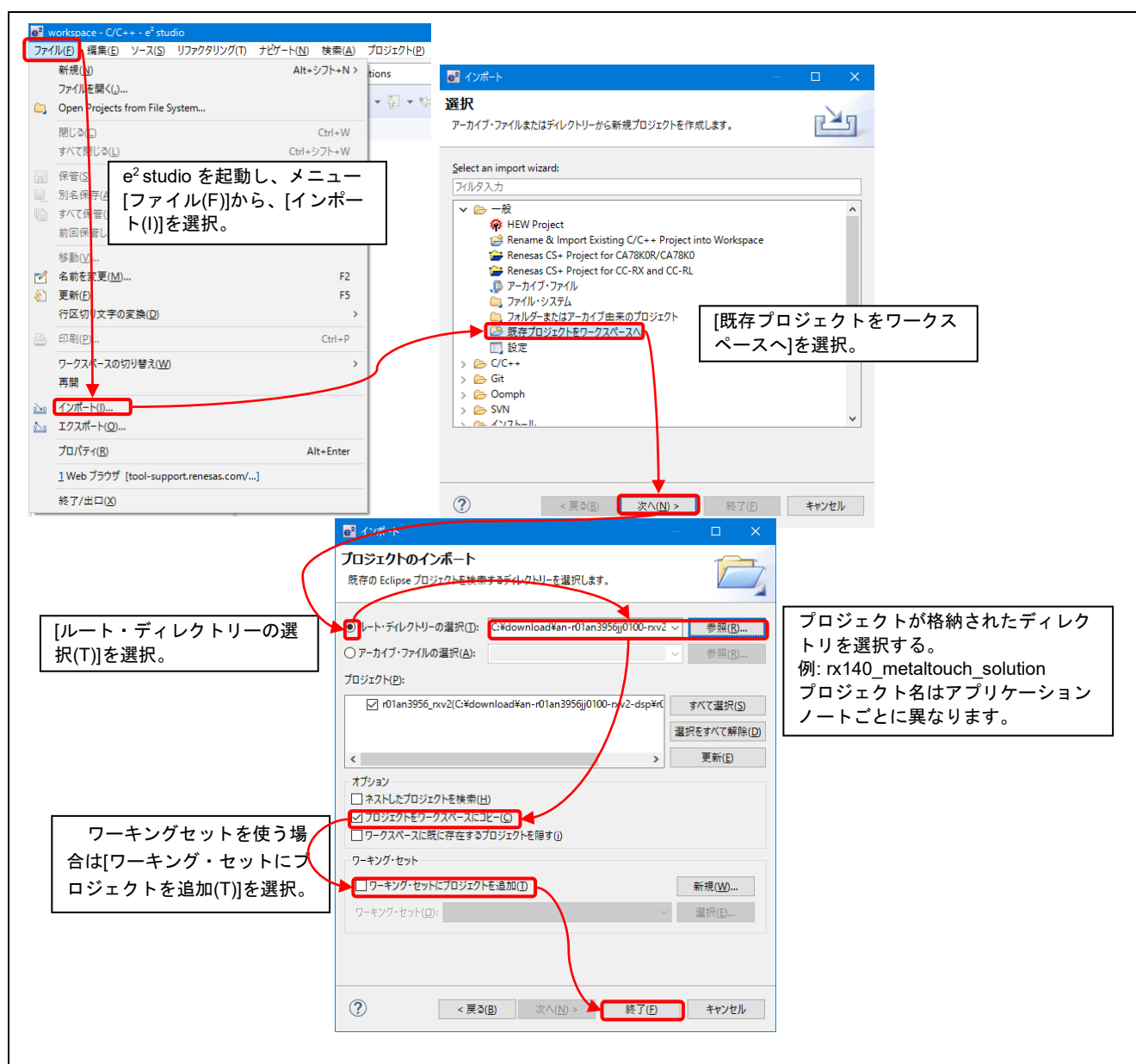


図 8-1 プロジェクトを e² studio にインポートする方法

8.2 CS+での手順

CS+でご使用になる際は、下記の手順でCS+にインポートしてください。

(使用するCS+のバージョンによっては画面が異なる場合があります。)

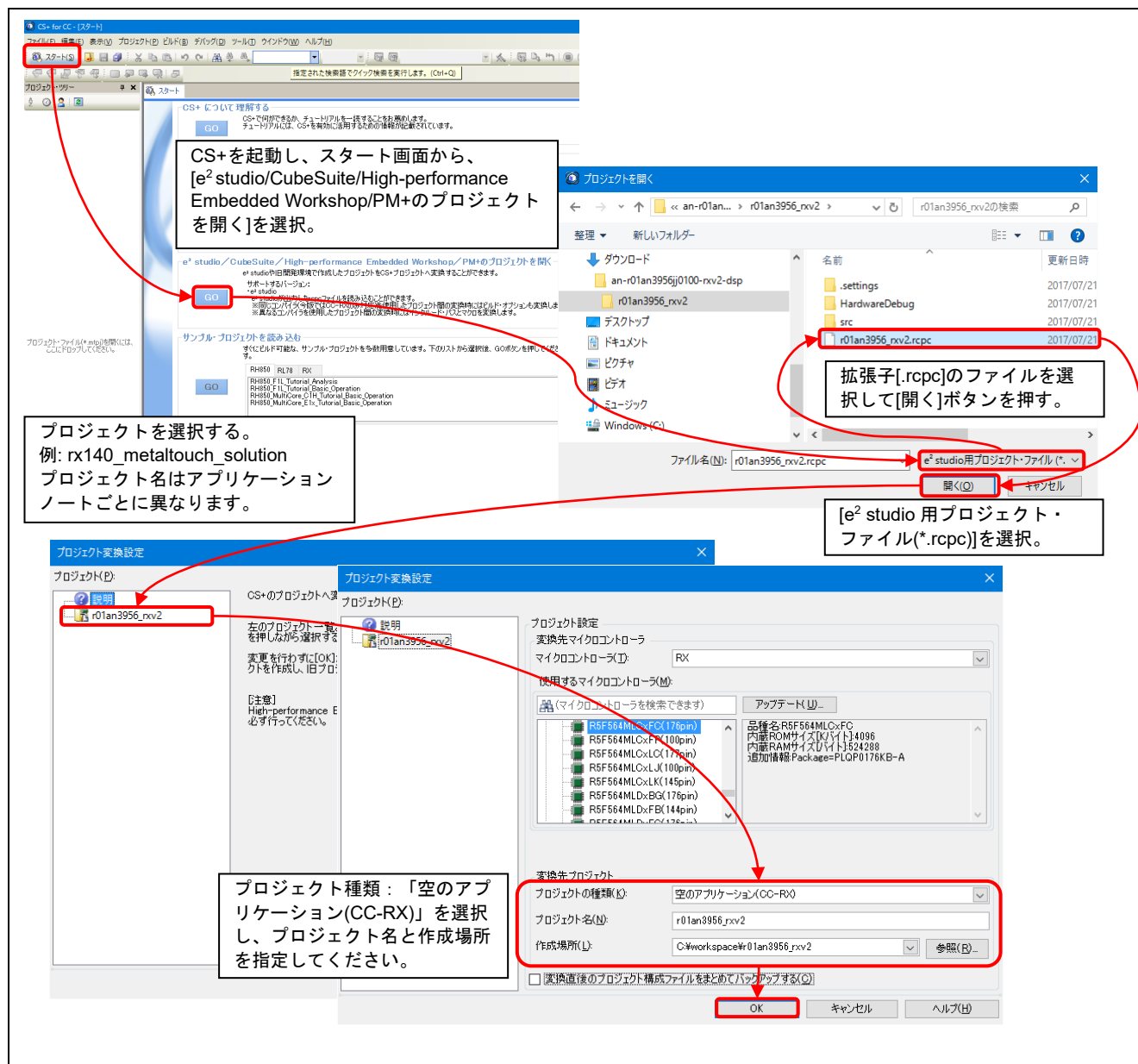


図 8-2 プロジェクトをCS+にインポートする方法

9. サンプルプログラムを使用した評価結果

9.1 使用メモリと実行サイクル数

9.1.1 ビルド条件

サンプルプログラムのビルド条件を表 9-1 に示します。

表 9-1 ビルド条件

項目	設定
Compiler	-isa=rxv2 -define=QE_TOUCH_CONFIGURATION -utf8 -nomessage -output=obj -obj_path=\${workspace_loc}/\${ProjName}/\${ConfigName}} -debug -outcode=utf8 -nologo
Linker	-noprelink -output="rx140_metaltouch_solution.abs" -form=absolute -nomessage -vect=_undefined_interrupt_source_isr -list -show=symbol -nooptimize -rom=D=R,D_1=R_1,D_2=R_2 -cpu=RAM=00000000-0000ffff, FIX=00080000-00083fff, FIX=00086000-00087fff, FIX=00088000-0008dfff, FIX=000a0000-000bffff, ROM=00100000-00101fff, FIX=007fc000-007fc4ff, FIX=007ffc00-007fffff, ROM=fffc0000-ffffffffff -nologo
セクション	SU,SI,B_1,R_1,B_2,R_2,B,R/04,PRresetPRG,C_1,C_2,C,C\$,D*,W*,L,P/0FFFC0000, EXCEPTVECT/0FFFFFFF80,RESETVECT/0FFFFFFFC

【注】 ユーザ設定以外のインクルードパス設定は省略

9.1.2 使用メモリ

サンプルプログラムのメモリ使用量を表 9-2 に示します。

表 9-2 メモリ使用量

項目			サイズ [byte]	
			設定	実使用量
ROM				18317
	Code			14546
	Data			3771
RAM			11361	10453
	Data			10081
	Stack	USER	1024	220
		INT	256	152

9.1.3 実行サイクル数と処理時間

CPU 実行サイクル数等を表 9-3 に、タッチ検出周期 20ms 中の処理時間分布を図 9-1 に示します。

表 9-3 実行サイクル数・実行時間

ICLK=48MHz

処理	最大実行サイクル数 (実行時間)		条件
	操作モード 1	操作モード 2	
SW3 押下検出処理	34cycle (0.708μs)		押下検出時
操作モード切替処理	1436cycle (29.917μs)		
タッチ検出処理	11750cycle (244.792μs)	19477cycle (405.771μs)	
ブザー発音処理	287cycle (5.979μs)		発音処理実行
押下表示処理	45689cycle (951.854μs)		
タッチ検出周期中の小計	61130cycle (1273.542μs)	68932cycle (1436.083μs)	タッチ時処理
ブザー停止処理	192cycle (4.000μs)		停止処理実行
CMIO 割り込み処理	1143 (23.813μs)		
CTSUFN 割り込み処理	88 (1.833μs)		

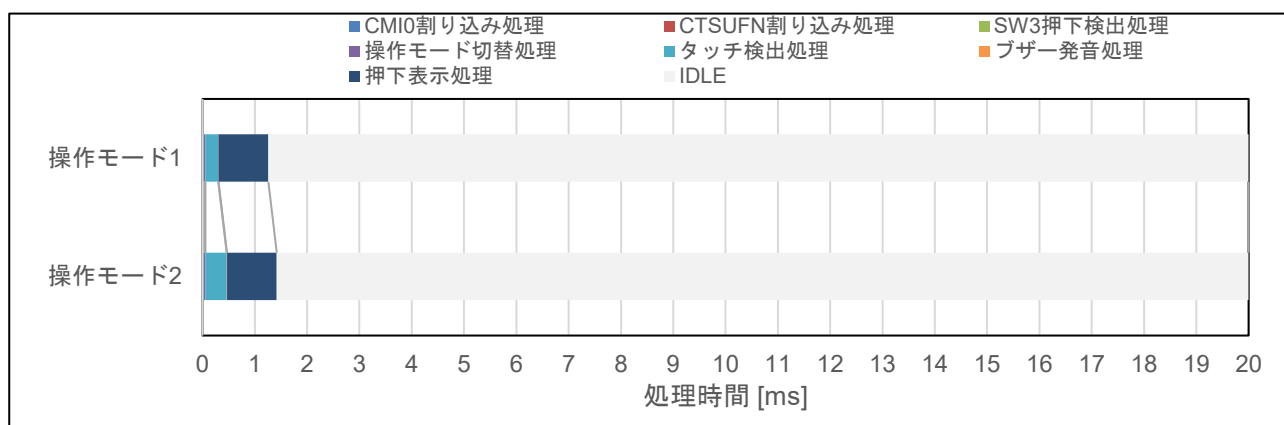


図 9-1 タッチ検出周期 20ms 中の処理時間分布

9.2 評価結果

9.2.1 条件

評価に際して、QE for Capacitive Touch でのタッチセンサ調整条件を表 9-4 に、各押下強度に対するしきい値設定を表 9-5 に示します。

表 9-4 QE for Capacitive Touch のタッチセンサ調整条件

項目	設定
config01 押下強度	500g
計測電圧	通常電圧(1.5V)
電流レンジ	40μA
センサドライブパルス周波数	F0: 0.500MHz F1: 0.430MHz, F2: 0.570MHz
計測時間	0.128ms
タッチ判定方法	VMM

表 9-5 タッチ判定しきい値割合設定

メソッド	押下強度	タッチ判定しきい値割合
config01	強	75%
config02	中	50%
config03	弱	25%

9.2.2 ボタン感度

本システムにおいて、各ボタンに 500g の分銅を置いたときの計測値とタッチ判定しきい値を図 9-2 に示します。

タッチ判定しきい値は、500g の分銅によるチューニング時の計測値から求めています。

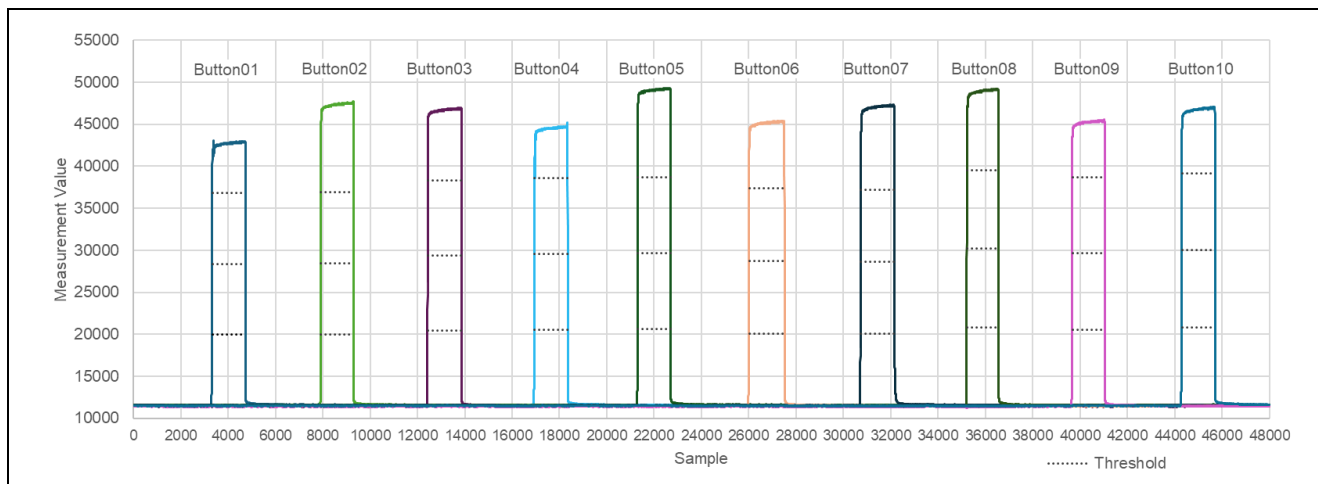


図 9-2 500g 搭載時の各ボタンの計測値とタッチ判定しきい値

また、各ボタンに 500g の分銅を置いた時に、他のボタンでの計測値変化量の最大値と押下強度弱のタッチ判定しきい値の比を図 9-3 に示します。

図 9-3 の各値は以下で定義します。

$$\text{変動比}[\%] = \frac{\text{計測値変化量}}{\text{押下強度弱のタッチ判定しきい値}} \cdot 100\%$$

他のボタンの変動比が 100% を越えると誤判定となりますが、変動比は十分に小さいので、誤判定は発生しないことがわかります。

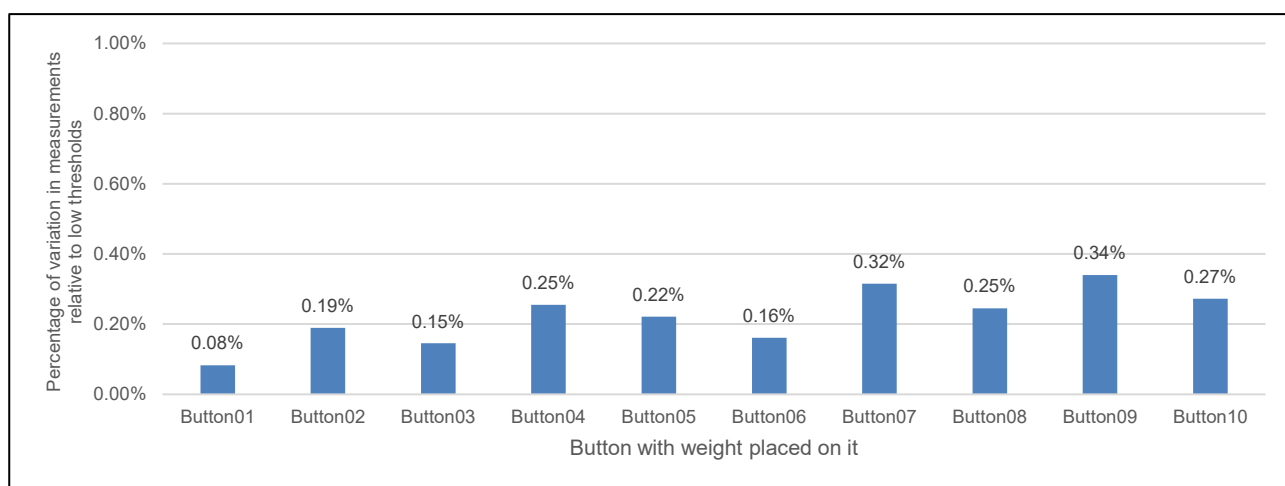


図 9-3 500g 搭載時の他のボタンへの影響

9.2.3 EMC 試験結果

本システムにおいて、EN 61000-4 に準拠した自主評価を行いました。結果を表 9-6 に示します。

表 9-6 EMC 試験結果

項目	試験レベル		性能基準
EN 61000-4-2:2009 (静電気放電)	直接放電	±8kV	A
	間接放電	±8kV	A
EN 61000-4-3:2020 (放射電磁界)	10 V/m		A
EN 61000-4-4:2012 (電氣的ファスト・トランジェントバースト)	±4kV (5kHz)		A
	±4kV (100kHz)		A
EN 61000-4-6:2014 (放射電磁界によって誘導された伝導妨害)	10V		A

性能基準 A は試験中および試験後も意図したとおりのタッチ判定ができるものとします。

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	Apr.10.25	－	初版

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 静電気対策

CMOS 製品の取り扱いの際は静電気防止を心がけてください。CMOS 製品は強い静電気によってゲート絶縁破壊を生じることがあります。運搬や保存の際には、当社が出荷梱包に使用している導電性のトレーやマガジンケース、導電性の緩衝材、金属ケースなどを利用し、組み立て工程にはアースを施してください。プラスチック板上に放置したり、端子を触ったりしないでください。また、CMOS 製品を実装したボードについても同様の扱いをしてください。

2. 電源投入時の処置

電源投入時は、製品の状態は不定です。電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. 電源オフ時における入力信号

当該製品の電源がオフ状態のときに、入力信号や入出力プルアップ電源を入れないでください。入力信号や入出力プルアップ電源からの電流注入により、誤動作を引き起こしたり、異常電流が流れ内部素子を劣化させたりする場合があります。資料中に「電源オフ時における入力信号」についての記載のある製品は、その内容を守ってください。

4. 未使用端子の処理

未使用端子は、「未使用端子の処理」に従って処理してください。CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。

5. クロックについて

リセット時は、クロックが安定した後、リセットを解除してください。プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

6. 入力端子の印加波形

入力ノイズや反射波による波形歪みは誤動作の原因になりますので注意してください。CMOS 製品の入力がノイズなどに起因して、 $V_{IL}(\text{Max.})$ から $V_{IH}(\text{Min.})$ までの領域にとどまるような場合は、誤動作を引き起こす恐れがあります。入力レベルが固定の場合はもちろん、 $V_{IL}(\text{Max.})$ から $V_{IH}(\text{Min.})$ までの領域を通過する遷移期間中にチャタリングノイズなどが入らないように使用してください。

7. リザーブアドレス（予約領域）のアクセス禁止

リザーブアドレス（予約領域）のアクセスを禁止します。アドレス領域には、将来の拡張機能用に割り付けられている リザーブアドレス（予約領域）があります。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

8. 製品間の相違について

型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。同じグループのマイコンでも型名が違うと、フラッシュメモリ、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が違う製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

ご注意書き

1. 本資料に記載された回路、ソフトウェアおよびこれらに関連する情報は、半導体製品の動作例、応用例を説明するものです。回路、ソフトウェアおよびこれらに関連する情報を使用する場合、お客様の責任において、お客様の機器・システムを設計ください。これらの使用に起因して生じた損害（お客様または第三者いずれに生じた損害も含みます。以下同じです。）に関し、当社は、一切その責任を負いません。
2. 当社製品または本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用回路例等の情報の使用に起因して発生した第三者の特許権、著作権その他の知的財産権に対する侵害またはこれらに関する紛争について、当社は、何らの保証を行うものではなく、また責任を負うものではありません。
3. 当社は、本資料に基づき当社または第三者の特許権、著作権その他の知的財産権を何ら許諾するものではありません。
4. 当社製品を組み込んだ製品の輸出入、製造、販売、利用、配布その他の行為を行うにあたり、第三者保有の技術の利用に関するライセンスが必要となる場合、当該ライセンス取得の判断および取得はお客様の責任において行ってください。
5. 当社製品を、全部または一部を問わず、改造、改変、複製、リバースエンジニアリング、その他、不適切に使用しないでください。かかる改造、改変、複製、リバースエンジニアリング等により生じた損害に関し、当社は、一切その責任を負いません。
6. 当社は、当社製品の品質水準を「標準水準」および「高品質水準」に分類しており、各品質水準は、以下に示す用途に製品が使用されることを意図しております。

標準水準： コンピュータ、OA 機器、通信機器、計測機器、AV 機器、家電、工作機械、パーソナル機器、産業用ロボット等

高品質水準： 輸送機器（自動車、電車、船舶等）、交通制御（信号）、大規模通信機器、金融端末基幹システム、各種安全制御装置等

当社製品は、データシート等により高信頼性、Harsh environment 向け製品と定義しているものを除き、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物的損害を発生させるおそれのある機器・システム（宇宙機器と、海底中継器、原子力制御システム、航空機制御システム、プラント基幹システム、軍事機器等）に使用されることを意図しておらず、これらの用途に使用することは想定していません。たとえ、当社が想定していない用途に当社製品を使用したことにより損害が生じても、当社は一切その責任を負いません。

7. あらゆる半導体製品は、外部攻撃からの安全性を 100%保証されているわけではありません。当社ハードウェア／ソフトウェア製品にはセキュリティ対策が組み込まれているものもありますが、これによって、当社は、セキュリティ脆弱性または侵害（当社製品または当社製品が使用されているシステムに対する不正アクセス・不正使用を含みますが、これに限りません。）から生じる責任を負うものではありません。当社は、当社製品または当社製品が使用されたあらゆるシステムが、不正な改変、攻撃、ウイルス、干渉、ハッキング、データの破壊または窃盗その他の不正な侵入行為（「脆弱性問題」といいます。）によって影響を受けないことを保証しません。当社は、脆弱性問題に起因したまたはこれに関連して生じた損害について、一切責任を負いません。また、法令において認められる限りにおいて、本資料および当社ハードウェア／ソフトウェア製品について、商品性および特定目的との合致に関する保証ならびに第三者の権利を侵害しないことの保証を含め、明示または黙示のいかなる保証も行いません。
8. 当社製品をご使用の際は、最新の製品情報（データシート、ユーザーズマニュアル、アプリケーションノート、信頼性ハンドブックに記載の「半導体デバイスの使用上の一般的な注意事項」等）をご確認の上、当社が指定する最大定格、動作電源電圧範囲、放熱特性、実装条件その他指定条件の範囲内でご使用ください。指定条件の範囲を超えて当社製品をご使用された場合の故障、誤動作の不具合および事故につきましては、当社は、一切その責任を負いません。
9. 当社は、当社製品の品質および信頼性の向上に努めていますが、半導体製品はある確率で故障が発生したり、使用条件によっては誤動作したりする場合があります。また、当社製品は、データシート等において高信頼性、Harsh environment 向け製品と定義しているものを除き、耐放射線設計を行っておりません。仮に当社製品の故障または誤動作が生じた場合であっても、人身事故、火災事故その他社会的損害等を生じさせないよう、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証を行ってください。特に、マイコンソフトウェアは、単独での検証は困難なため、お客様の機器・システムとしての安全検証をお客様の責任で行ってください。
10. 当社製品の環境適合性等の詳細につきましては、製品個別に必ず当社営業窓口までお問合せください。ご使用に際しては、特定の物質の含有・使用を規制する RoHS 指令等、適用される環境関連法令を十分調査のうえ、かかる法令に適合するようご使用ください。かかる法令を遵守しないことにより生じた損害に関して、当社は、一切その責任を負いません。
11. 当社製品および技術を国内外の法令および規則により製造・使用・販売を禁止されている機器・システムに使用することはできません。当社製品および技術を輸出、販売または移転等する場合は、「外国為替及び外国貿易法」その他日本国および適用される外国の輸出管理関連法規を遵守し、それらの定めるところに従い必要な手続きを行ってください。
12. お客様が当社製品を第三者に転売等される場合には、事前に当該第三者に対して、本ご注意書き記載の諸条件を通知する責任を負うものとしします。
13. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。
14. 本資料に記載されている内容または当社製品についてご不明な点がございましたら、当社の営業担当者までお問合せください。

注 1. 本資料において使用されている「当社」とは、ルネサス エレクトロニクス株式会社およびルネサス エレクトロニクス株式会社が直接的、間接的に支配する会社をいいます。

注 2. 本資料において使用されている「当社製品」とは、注 1 において定義された当社の開発、製造製品をいいます。

(Rev.5.0-1 2020.10)

本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレシア）

www.renesas.com

お問合せ窓口

弊社の製品や技術、ドキュメントの最新情報、最寄の営業お問合せ窓口に関する情報などは、弊社ウェブサイトをご覧ください。

www.renesas.com/contact/

商標について

ルネサスおよびルネサスロゴはルネサス エレクトロニクス株式会社の商標です。すべての商標および登録商標は、それぞれの所有者に帰属します。