

S5U1C17656T
タッチキー
参考資料

評価ボード・キット、開発ツールご使用上の注意事項

1. 本評価ボード・キット、開発ツールは、お客様での技術的評価、動作の確認および開発のみに用いられることを想定し設計されています。それらの技術評価・開発等の目的以外には使用しないで下さい。本品は、完成品に対する設計品質に適合していません。
2. 本評価ボード・キット、開発ツールは、電子エンジニア向けであり、消費者向け製品ではありません。お客様において、適切な使用と安全に配慮願います。弊社は、本品を用いることで発生する損害や火災に対し、いかなる責も負いかねます。通常の使用においても、異常がある場合は使用を中止して下さい。
3. 本評価ボード・キット、開発ツールに用いられる部品は、予告無く変更されることがあります。

本資料のご使用につきましては、次の点にご留意願います。

本資料の内容については、予告無く変更することがあります。

1. 本資料の一部、または全部を弊社に無断で転載、または、複製など他の目的に使用することは堅くお断りいたします。
2. 本資料に掲載される応用回路、プログラム、使用方法等はあくまでも参考情報であり、これらに起因する第三者の知的財産権およびその他の権利侵害あるいは損害の発生に対し、弊社はいかなる保証を行うものではありません。また、本資料によって第三者または弊社の知的財産権およびその他の権利の実施権の許諾を行うものではありません。
3. 特性値の数値の大小は、数直線上の大小関係で表しています。
4. 製品および弊社が提供する技術を輸出等するにあたっては「外国為替および外国貿易法」を遵守し、当該法令の定める手続きが必要です。大量破壊兵器の開発等およびその他の軍事用途に使用する目的をもって製品および弊社が提供する技術を費消、再販または輸出等しないでください。
5. 本資料に掲載されている製品は、生命維持装置その他、きわめて高い信頼性が要求される用途を前提としていません。よって、弊社は本（当該）製品をこれらの用途に用いた場合のいかなる責任についても負いかねます。
6. 本資料に掲載されている会社名、商品名は、各社の商標または登録商標です。

要旨

本資料は、S1C17656 を搭載した S5U1C17656T にて、入出力ポート (PPOINT) と 16 ビット PWM タイマ (T16A2) を用いて、タッチキーの容量で変わるカウンタ値 (容量相当値) をサンプリングし、UART で送信したり、タッチされたキーに応じた動作結果を LCD に表示させたりするための参考資料です。

動作環境

- S5U1C17656T (SVT17656: Software Evaluation Tool for S1C17656)
ICDmini との接続には専用ケーブル (4pin-4pin) 2 種が必要です
- ICDmini (S5U1C17001H)
パソコンとの接続には USB ケーブルが必要です
- パソコン
 - GNU17 (S5U1C17001C) 開発ツールインストール済み ※
 - ICDmini USB ドライバインストール済み
- 最新版の FLS17656 (ファイル名: fls17656.elf) /FLS17656 (ファイル名: fls17656.elf)
内蔵 FLASH へのプログラムの書き込みに必須
- S1C17656 タッチキー用プログラミングパッケージ (本パッケージ)
 - タッチキーの容量相当値の時間ごと変化の視覚化 プログラミングパッケージ (s1c17656_touch_oscillo)、VBA マクロを含む Excel ファイル (MeasTouch.xlsx)、Active-X コントロールファイル (NonComSck.ocx)
 - タッチキーをスキャンしてそのキーに応じた動作結果を LCD に表示するプログラミングパッケージ (s1c17656_touch_demo)

※ 本パッケージは、GNU17 V2.3.0 で動作確認しています。

目次

要旨	i
1. 仕様	2
2. 使用機能説明	3
3. 動作原理	4
3.1. 検出原理	4
3.2. 検出方法	4
3.3. 検出回路	6
4. 容量相当値の測定結果の一例	8
5. 基板の設計	10
6. ソフトウェア説明	11
6.1. s1c17656_touch_oscillo について	11
6.1.1. 操作手順	11
6.1.2. MeasTouch.xlsm の使い方	12
6.2. s1c17656_touch_demo について	14
6.2.1. 操作手順	14
6.2.2. サンプルプログラム操作概要	14
6.2.3. 電池寿命の見積もり	15
改訂履歴表	16

1. 仕様

S1C17656 を搭載した S5U1C17656T(SVT17656)にて、入出力ポート (PPOINT) と 16 ビット PWM タイマ (T16A) を用いてタッチキーの容量で変わるカウンタ値 (容量相当値) をサンプリングします。

サンプルプログラムのうち、s1c17656_touch_oscillo は、VBA マクロを含む Excel と連携して、一定の周期でタッチキーの容量で変わるカウンタ値 (容量相当値) をサンプリングし、UART を用いて PC に送信して Excel の表に書き込みます。Excel の表に書かれた配列の値はグラフ化されるため、時間ごと容量相当値の変化を視覚化することができます。

またサンプルプログラムのうち、s1c17656_touch_demo は、計時タイマ (CT) で一定時間ごと繰り返しタッチキーの状態をスキャンして、タッチされたキーに応じた動作結果を LCD に表示します。

2. 使用機能説明

PPORT	P0[5:2]、P1[7:4]、P2[3:0] 計 12 ポートをタッチキー用に割り当て、「L」出力／「H」出力／Hi-Z の何れかに設定します。
T16A2	タッチキーの容量相当値を得るために、カウントタイマーとして使用します。
CT	タッチキーを定期的にスキャンするためのインターバルタイマとして、計時タイマの割り込み機能を使用しています。

以下は、slc17656_touch_oscillo の場合です。

UART	PPORT の P00 を USIN0 に、P01 を USOUT0 に割り当てます。PC 上で実行する VBA マクロを含む Excel との通信に使用します。
------	---

以下は、slc17656_touch_demo の場合です。

LCD	タッチキーの状態を LCD に表示するために使用します。
SND	タッチキーのクリック音を出力するために使用します。

動作モード T16A2：タッチキー端子の容量相当値を得るために、カウントタイマーとして使用します。

システムクロック システムクロックは OSC3B（内蔵発振 4MHz）を使用します。slc17656_touch_demo は OSC1A（32.768kHz）も使用します。

割り込み 以下は、slc17656_touch_oscillo の場合。
T8 のベクタ番号とベクタアドレスは以下の通りです。
T8 ベクタ番号： 14 (0x0e)
 ベクタアドレス： 0x8038
本サンプルでは、次の割り込みを使用します。
 タイマアンダーフロー割り込み

UART のベクタ番号とベクタアドレスは以下の通りです。
UART ベクタ番号： 16 (0x10)
 ベクタアドレス： 0x8040
本サンプルでは、次の割り込みを使用します。
 受信バッファ 1 バイトフル割り込み

以下は、slc17656_touch_demo の場合。
CT のベクタ番号とベクタアドレスは以下の通りです。
CT ベクタ番号： 7 (0x07)
 ベクタアドレス： 0x801c
本サンプルでは、次の割り込みを使用します。
 タイマ 32Hz/2Hz/1Hz 信号割り込み

RTC のベクタ番号とベクタアドレスは以下の通りです。
RTC ベクタ番号： 8 (0x08)
 ベクタアドレス： 0x8020
本サンプルでは、次の割り込みを使用します。
 1 日/半日割り込み

3. 動作原理

3.1. 検出原理

空気やプラスチックなどの絶縁体（誘電体）で隔離された状態で複数の導体を設置すると、それら導体間には一定の容量値のコンデンサが形成されます。例えば図 3-1 のような構造でプリント基板上にセンサーパッド用の電極とグランドパターンを設けると、 $C1$ と $C2$ の2つのコンデンサが形成されるため、両コンデンサを並列接続した ($C1+C2$) の容量値がセンサーパッド用の電極とグランドパターンで観察されることとなります。

一方、人体は電気を通す良導体であり、また静電気を帯びることでわかる通り、たとえ靴など絶縁性の高いものでグランドと隔離されても人体とグランドの間には大きな容量があるため、先のセンサーパッドに近づくことで近接効果を起こすことができます。具体的には、センサーパッド上面の誘電体の表面に指を触れることでセンサーパッドに新たに容量 $C3$ が形成され、センサーパッドの容量は ($C1+C2+C3$) と増大することとなります。

この容量値の変化を検出することで、タッチセンサを作ることができます。

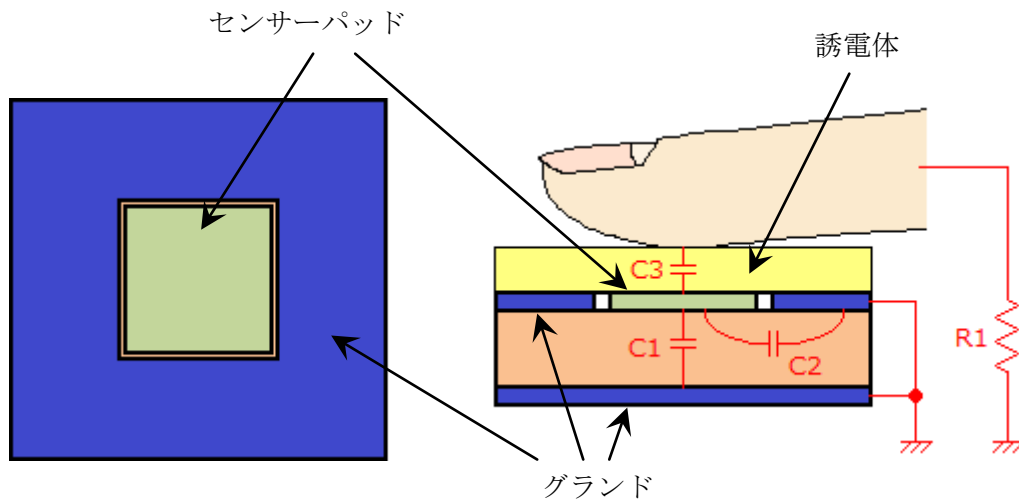


図 3-1 センサーパッドの容量（上面図・断面図）

3.2. 検出方法

容量の検出方法を、図 3-2 の容量検出回路の概念図と、図 3-3 の検出波形の時間変化を用いて説明します。

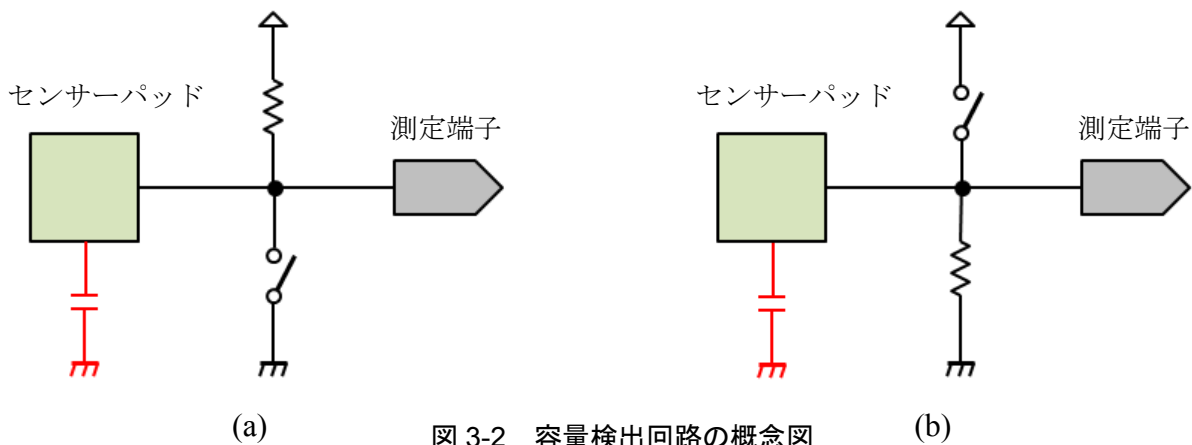


図 3-2 容量検出回路の概念図

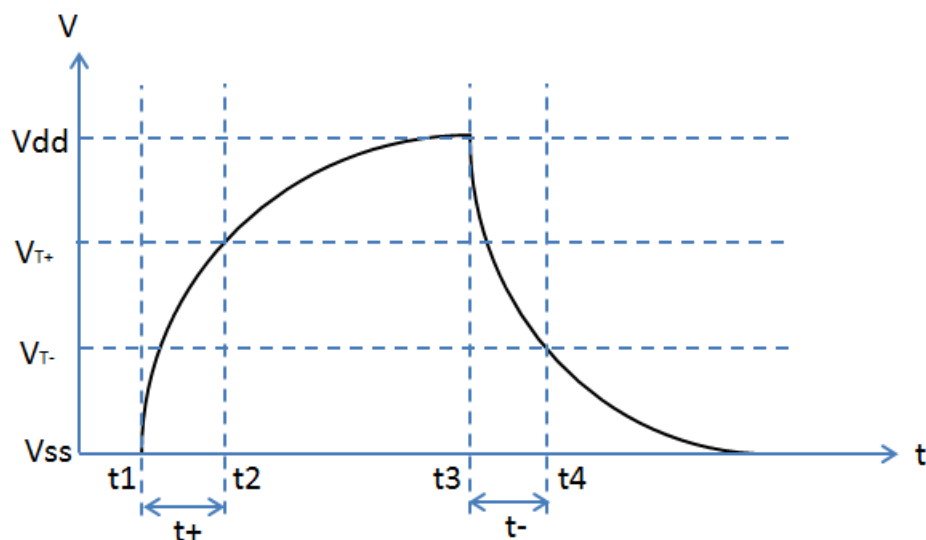


図 3-3 検出波形の時間変化

図 3-2 のセンサーパッドとグラウンドの間には容量成分が存在します。最初に図 3-2 の(a)のスイッチを一定時間オンにし、その容量成分に蓄えられた電荷をすべて放電させます。次に図 3-2 の(a)のスイッチをオフにした瞬間が図 3-3 の時刻 t_1 に相当しますが、それ以降は抵抗を通して容量成分に電荷が充電されます。従い測定端子の電位は、 $\tau=CR$ の時定数で上昇しますが、ここでは測定端子として CMOS シュミット入力回路を用います。CMOS シュミット入力回路は、 V_T 以下の電圧の場合に”L”レベル、 V_T 以上の電圧の場合に”H”レベルとして認識されます。すなわち、電圧が V_{T+} に達する時刻 t_2 までの間は、CMOS シュミット入力回路は”L”レベルを認識し、時刻 t_2 以降は”H”レベルを認識することになります。時刻 t_1 と t_2 の間の時間 t_+ を計測することでセンサーパッドの容量値を測定することができます。

逆の電位変化を起こすことでも容量値の測定は可能です。

図 3-2 の(b)のスイッチを一定時間オンにし、センサーパッドの容量成分に充電して V_{dd} まで電位を上昇させます。次に図 3-2 のスイッチをオフした瞬間が図 3-3 の時刻 t_3 に相当しますが、それ以降は抵抗を通して容量成分の電荷は放電されます。従い測定端子の電位は、 $\tau=CR$ の時定数で下降しますが、電圧が V_T に達する時刻 t_4 までの間は、CMOS シュミット入力回路は”H”レベルを認識し、時刻 t_4 以降は”L”レベルを認識することになります。時刻 t_3 と t_4 の間の時間 t_- を計測することでも、先と同様にセンサーパッドの容量値を測定することができます。

本参考資料では、 t_+ 、 t_- の時間カウントのためには T16A2 を、図 3-2 のスイッチとしては PPORT を用いています。具体的には、図 3-2 の(a)のスイッチのオン/オフは、PPORT を”L”レベル出力/Hi-Z に切り替えることで、図 3-2 の(b)のスイッチのオン/オフは、PPORT を”H”レベル出力/Hi-Z に切り替えることで代替しています。また電圧レベルの測定は、対象のセンサーパッドを PPORT の入力に接続し、レベルが変わるまでの時間を T16A2 でカウントすることで容量相当値を得ています。

次の章で示す図 3-4 の、指を触れているセンサーパッドをセンシングしているときに出力を”H”/”L”に切換えている PPORT を二重枠線で示しましたが、あわせて確認していただければ動作理解の一助になると思います。

3.3. 検出回路

S1C17656 に搭載されている PPORT、T16A2 を用いた検出回路の一例を図 3-4 に示します。

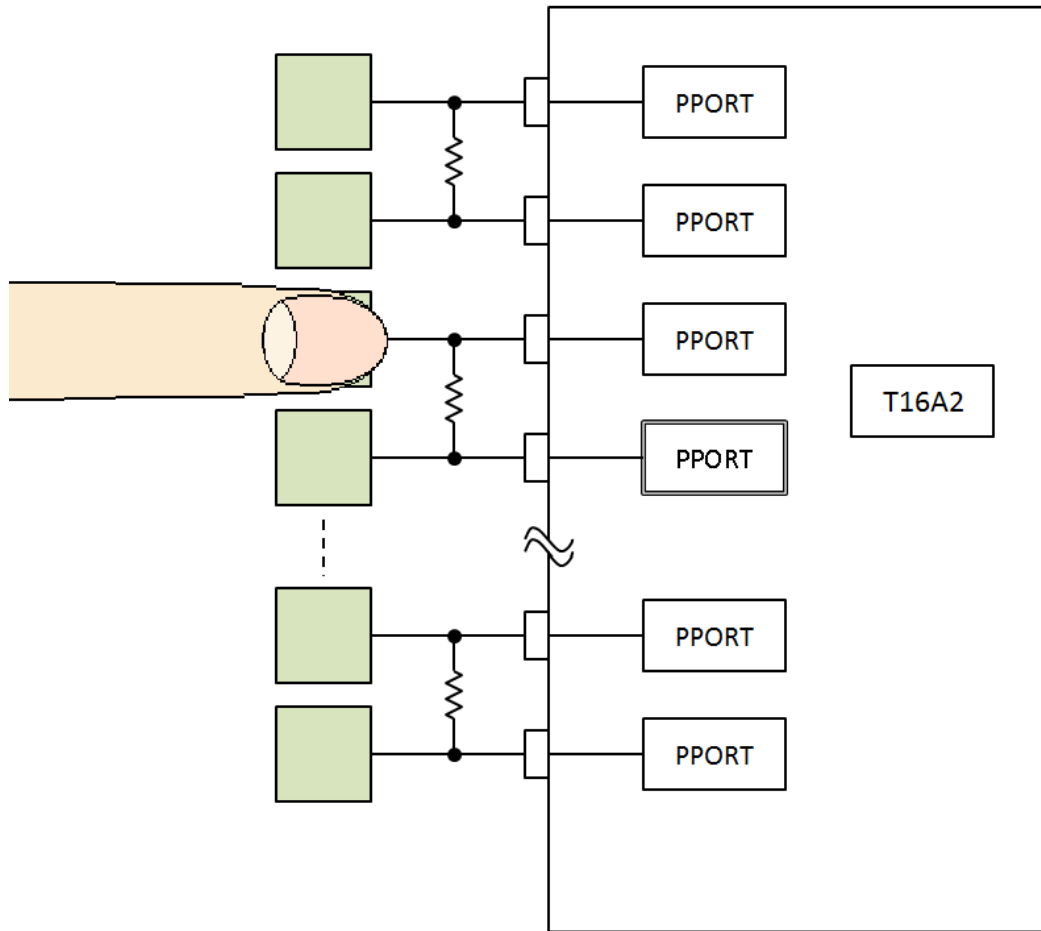


図 3-4 検出回路の一例

タッチキーは 2 つで一組になっており、間に抵抗器が設置されています。この抵抗器の値は、S1C17 シリーズの PPORT のリーク電流が $I_{LEAK} = \pm 150\text{nA}$ の範囲であること、シュミット入力スレッショルド電圧の平均値が $V_{T+} = 0.7V_{DD}$ 、 $V_{T-} = 0.3V_{DD}$ であることを考慮し、以下のように決めることができます。

タッチキーの形状が $12\text{mm} \times 12\text{mm}$ の正方形、プリント基板として厚さ $d = 1.2\text{mm}$ の FR-4 ガラスエポキシ基板を使用、タッチキーが単純な平行平板コンデンサであると仮定すると、面積 $S = 144\text{mm}^2$ 、比誘電率 $\epsilon_r = 4.7$ 、真空の誘電率 $\epsilon_0 = 8.855 \times 10^{-12}\text{F/m}$ を用いて容量 C を計算すると、

$$C = \frac{\epsilon_r \epsilon_0 S}{d} = \frac{4.7 \times 8.855 \times 10^{-12} \times 144 \times 10^{-6}}{1.2 \times 10^{-3}} = 5.0\text{pF}$$

図 3-3 の t_1 から t_2 の間の時間 t で、電圧が V_{SS} から V_{T+} に変化する期間に着目して計算します。

抵抗器の抵抗値 R とタッチキーの容量 C による時定数 CR の影響で、検出波形の電圧 V は t_1 からの経過時間 t に伴い以下のように変化します。

$$V = V_{DD} \times \left(1 - \exp\left(-\frac{1}{CR} \cdot t\right)\right)$$

上の式を変換して、

$$R = - \frac{t}{\ln\left(1 - \frac{V}{V_{dd}}\right)} \cdot \frac{1}{C}$$

時間 t の経過で電圧 $V = V_{T+} = 0.7V_{dd}$ に変化するので、

$$R = - \frac{t+}{\ln\left(1 - \frac{0.7 \cdot V_{dd}}{V_{dd}}\right)} \cdot \frac{1}{5.0 \times 10^{-12}}$$

$$= 1.66 \times 10^{11} \cdot t+$$

$t+$ の値として、T16A2 に入力される 4MHz のクロックの 100 カウントを用いると、

$$R = 1.66 \times 10^{11} \cdot t+$$

$$= 1.66 \times 10^{11} \cdot \frac{100}{4 \times 10^6}$$

$$= 4.15 \text{M}\Omega$$

また、C17 シリーズのリーク電流が $I_{LEAK} = \pm 150 \text{nA}$ の範囲であり、 $V_{dd} = 3.3 \text{V}$ 時の抵抗値に換算すると、

$$R = \frac{V}{I} = \frac{3.3}{150 \times 10^{-9}}$$

$$= 22 \text{M}\Omega$$

すなわち、先の $4.15 \text{M}\Omega$ はリーク電流に比べて十分小さな抵抗値で、本センシング方式が実用的なものであることがわかります。

ところで、 R の計算値は $4.15 \text{M}\Omega$ になりましたが、実際には入手しやすい E24 系列の抵抗値 $4.7 \text{M}\Omega$ 、 $5.1 \text{M}\Omega$ 、 $5.6 \text{M}\Omega$ あたりが通常使用に適した値になります。

ただし、後ほど基板設計の章でも説明しますが、実際の C の値は、平行平板コンデンサとして計算される値より小さくなるようパターン設計しますが、それでも配線の引き回しなどに起因する容量が附加されるため、実際の C の値は 5.0pF よりも大きな値になり、非タッチ状態に於いて T16A2 でカウントされる値は先の 100 カウントより大きな値が観察されるのが通常です。

なお、図 3-3 の t_3 から t_4 の間の時間 t で、電圧が V_{dd} から V_T に変化する期間に着目して計算しても、同様な結果が得られます。

4. 容量相当値の測定結果の一例

図 3-3 の検出波形で示したように、タッチキーの容量相当値は、 t_+ 、 t_- の 2 つを計測することができますが、非常に短い時間の間に測定されたこの 2 つの値を用いることで、ノイズの影響を抑制することができます。また、移動平均演算することで、更に明確に状態の区別が可能になります。その詳細については、「S1C17W22/S1C17W23 タッチキー アプリケーションノート」を参照してください。

さて、図 4-1～図 4-5 は、各々異なる確認条件で、30ms ごとにサンプリングされた 1000 点の、容量相当値 t_+ と t_- の値を足したものの平均値との差をプロットしたものです。キーにタッチした時の図 4-1、図 4-2 の場合と、キー以外の部分にタッチした時の図 4-3～図 4-5 の場合との間で明瞭な差が見られ、キー部への指のタッチが検出できていることがわかります。

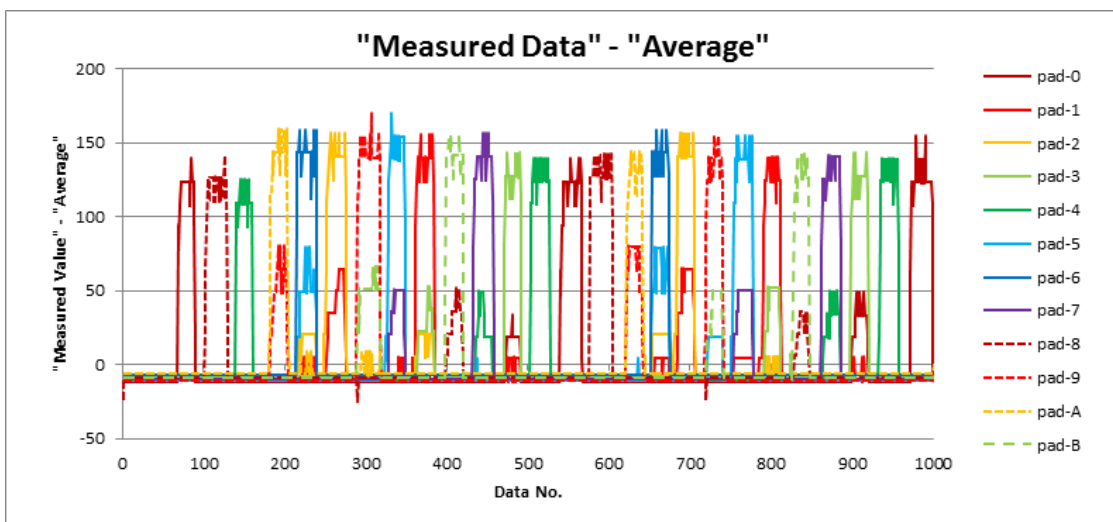


図 4-1 じっくり指をタッチした時

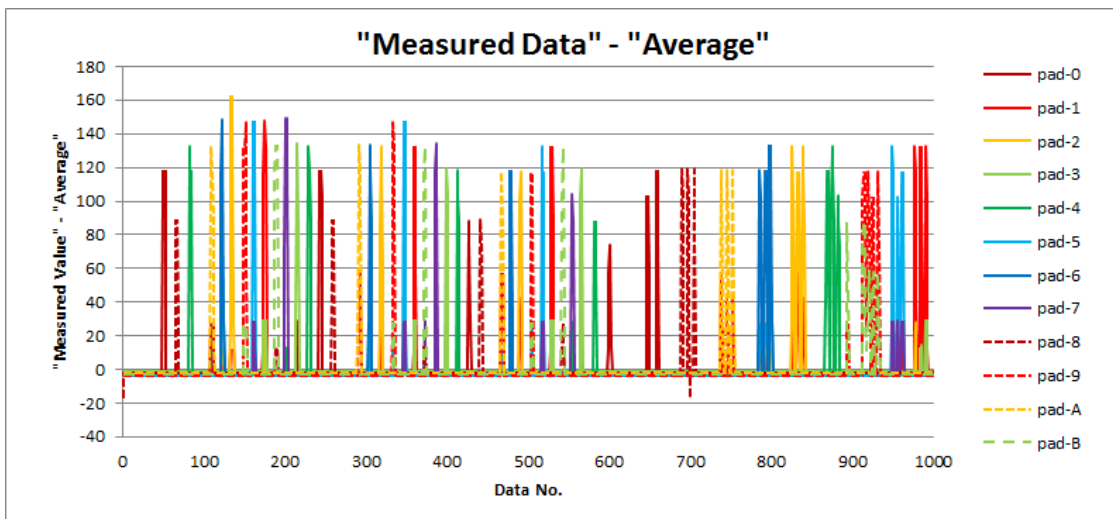


図 4-2 軽く指をタッチした時

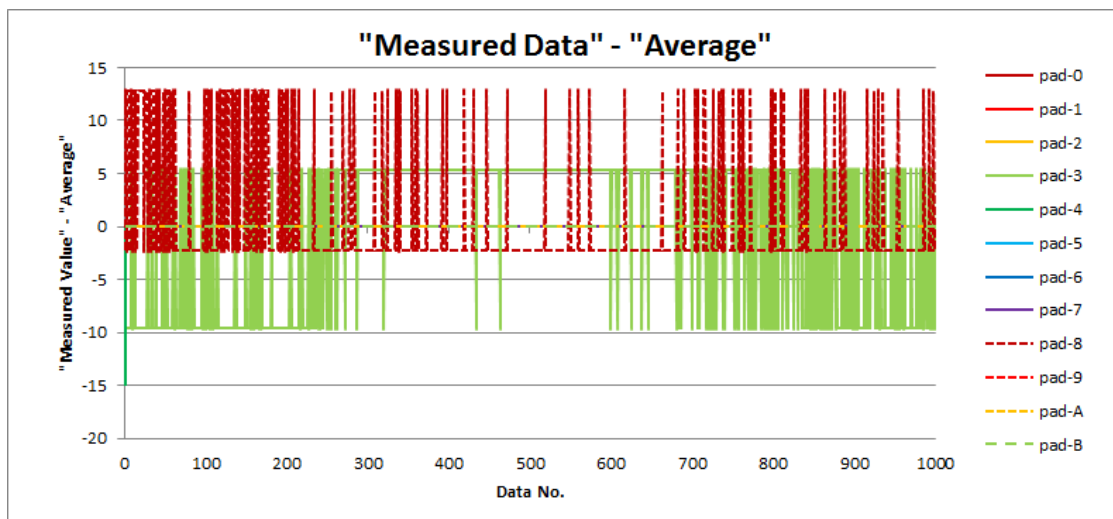


図 4-3 基板側面を手で持った時

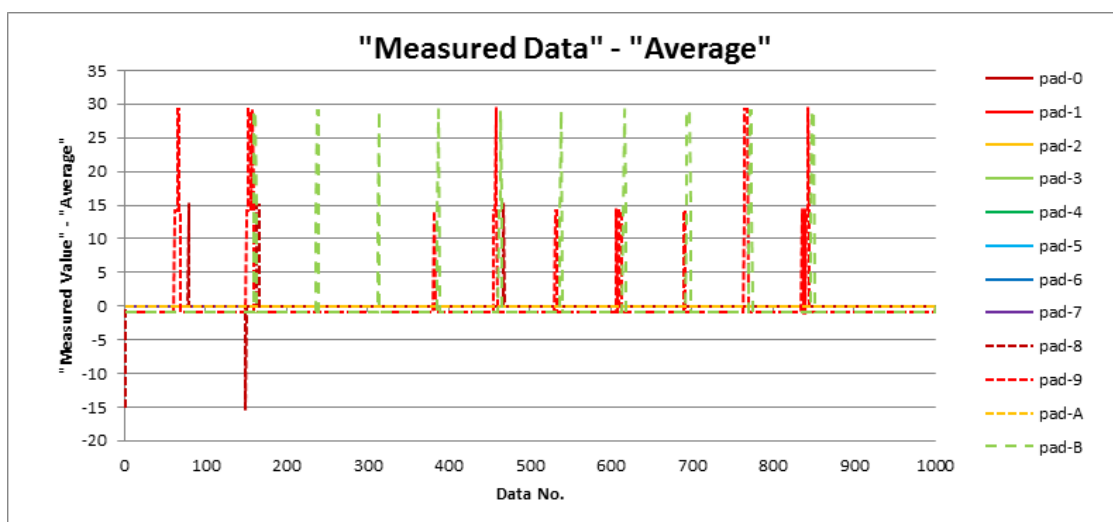


図 4-4 基板表側の左右の GND に交互に指を接触した時

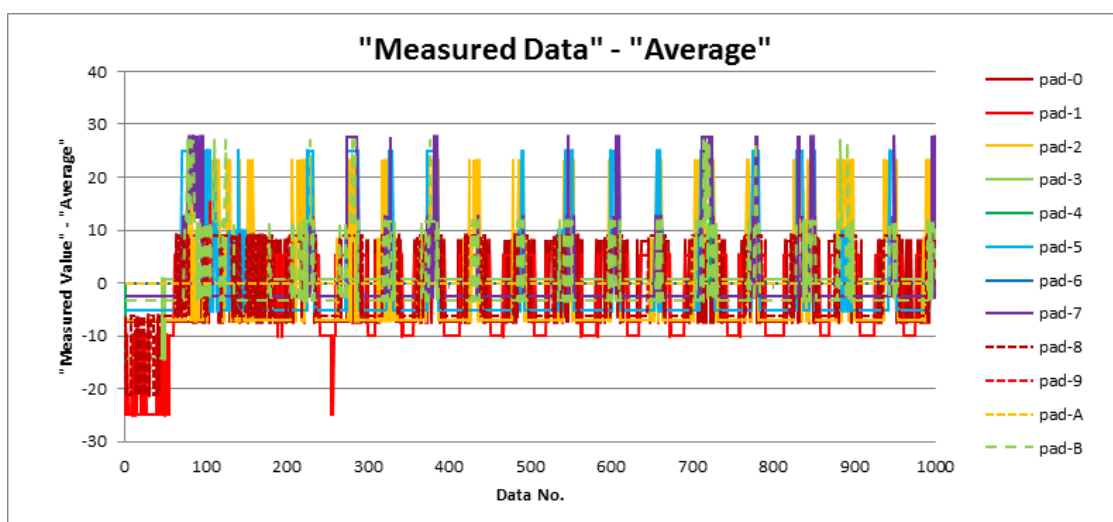


図 4-5 基板裏面に接触した指を揺動した時

5. 基板の設計

図 5-1 に S5U1C17656T(SVT17656)の基板パターンを示します。

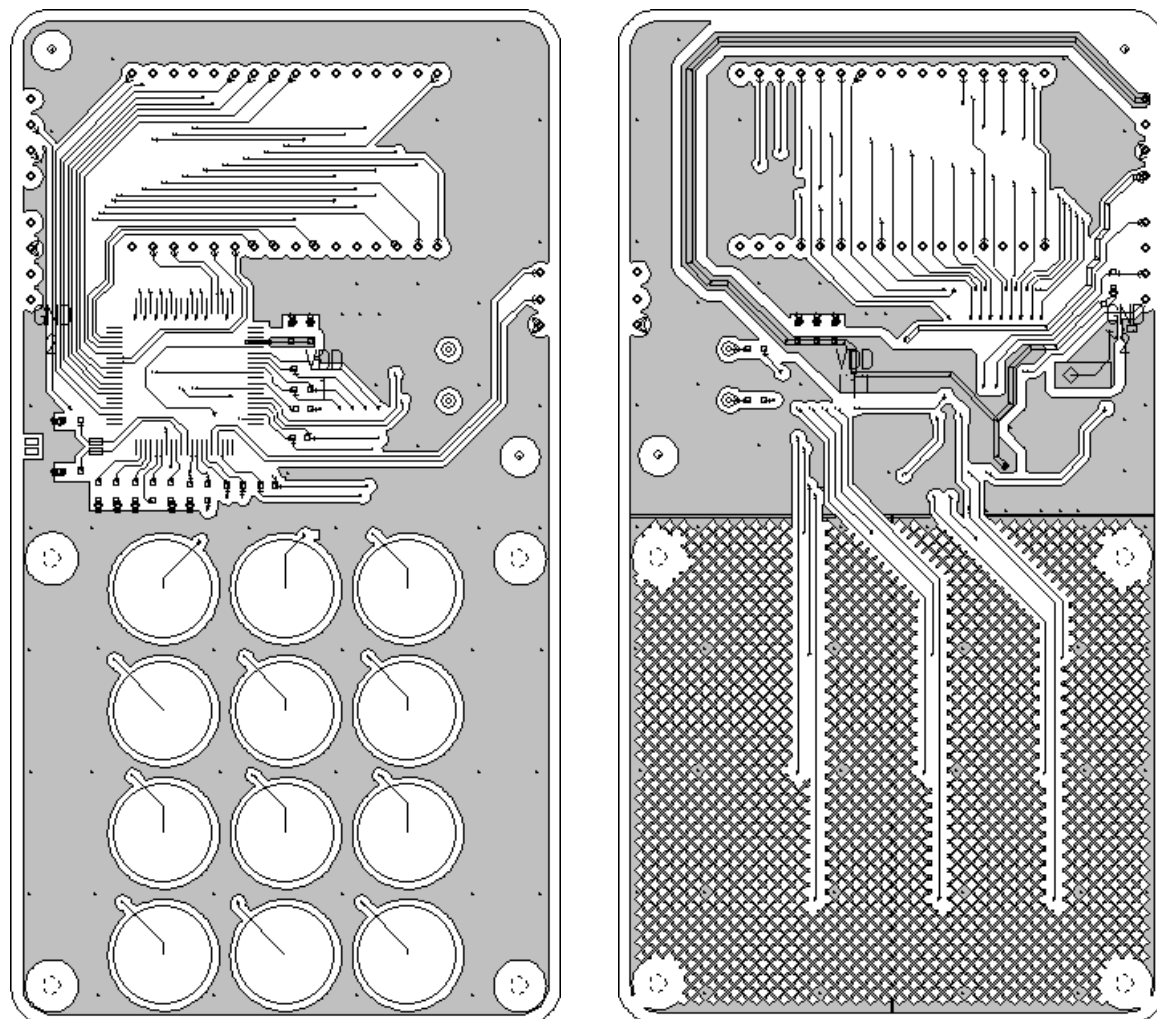


図 5-1 S5U1C17656T の基板パターン（表・裏）

S5U1C17656T(SVT17656)の基板パターンは、「S1C17W22/S1C17W23 タッチキー アプリケーションノート」に記載したノウハウに沿ってレイアウトしたものです。詳細については、その資料を参照してください。

6. ソフトウェア説明

6.1. s1c17656_touch_oscillo について

VBA マクロを含む Excel と連携して、一定の周期でタッチキーの容量で変わるカウンタ値（容量相当値）をサンプリングし、UART を用いて PC に送信して Excel の表に書き込むソフトウェア、s1c17656_touch_oscillo について説明します。

6.1.1. 操作手順

プロジェクトのインポート

(1) IDE を起動して、「s1c17656_touch_oscillo」プロジェクトをインポートしてください。

※ インポートの方法は、S5U1C17001C Manual “3.ソフトウェア開発手順”を参照してください。

ビルド

(1) IDE を使用して、「s1c17656_touch_oscillo」プロジェクトをビルドしてください。

Excel VBA を使用するための準備

- (1) デスクトップに MeasTouch フォルダをコピーしてください。このフォルダには、Excel ファイル「MeasTouch.xlsm」、Active-X コントロールファイル「NonComSck.ocx」が入っています。同梱されていない、マイクロソフトが提供する Active-X コントロールファイル「MSCOMM32.OCX」は、種々のサイトからダウンロードすることが可能ですが、それを入手し、先の MeasTouch フォルダ内にコピーしてください。
- (2) [すべてのプログラム]→[アクセサリ]→[コマンド プロンプト]上で右ボタンを押し、[管理者として実行...]を選択すると、「管理者: コマンド プロンプト」が起動します。
- (3) 「cd C:¥Desktop¥MeasTouch」、「regsvr32.exe MSCOMM32.OCX」、「regsvr32.exe NonComSck.ocx」を実行して Active-X コントロールファイルを有効にします。
- (4) 上記操作でうまくいかない場合、Active-X コントロールファイルを「C:¥Windows¥System32」にコピーし、「cd C:¥Windows¥System32」、「regsvr32.exe MSCOMM32.OCX」、「regsvr32.exe NonComSck.ocx」を実行してください。

接続、電源投入

- (1) S5U1C17656T(SVT17656)、ICDmini、USB ケーブル、PC を接続してください。
- (2) S5U1C17656T(SVT17656)の UART Ch.0 と、Excel VBA を実行する PC のシリアル端子（USB 接続のシリアル通信 I/F を使うのが一般的です。市販品の例を Excel VBA ファイルのシート「Note」に示しました。）を接続してください。PC に搭載された CPU の処理能力が十分なら、この PC は先の PC と兼用することもできます。
- (3) S5U1C17656T(SVT17656)、ICDmini をリセットしてください。

Excel の VBA の実行

(1) Excel ファイル「MeasTouch.xlsm」をダブルクリックして実行してください。

実行

(1) 必要に応じて、コマンドファイル（s1c17656_touch_oscillo_gnu17IDE.cmd）内の fls17656.elf へ

のパスを変更してください。デフォルトでは、
C:/EPSON/GNU17/mcu_model/17656/fls/fls17656.elf となっています。
※ コマンドファイルの修正方法は、S5U1C17001C Manual を参照してください。

- (2) IDE を使用して、「s1c17656_touch_oscillo」プロジェクトを実行してください。
- (3) 「s1c17656_touch_oscillo」の実行後、Excel VBA からのコマンドで、タッチキーをスキャンします。

6.1.2. MeasTouch.xlsm の使い方

MeasTouch.xlsm の外観を図 6-1 に示します。

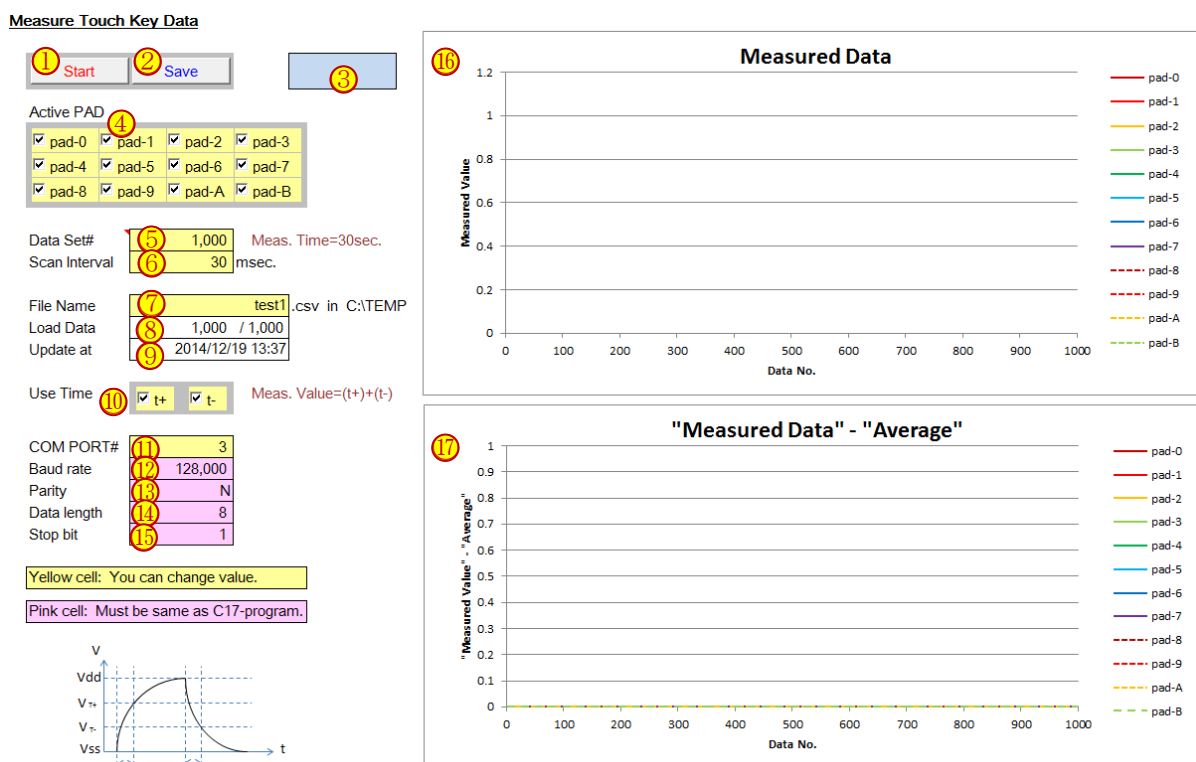


図 6-1 MeasTouch.xlsm の外観

- | | |
|--|---|
| <ol style="list-style-type: none"> ① 測定開始ボタン ② データセーブボタン ③ 状態インジケータ ④ 測定対象タッチキー選択 ⑤ データセット数 ⑥ スキャンインターバル ⑦ セーブファイル名 ⑧ データ取り込み数 ⑨ データ採取日時 ⑩ t+、t- 使用選択 ⑪ COM ポート番号 | <p>タッチキーの容量相当値の取得を開始します。</p> <p>計測データを、CSV 形式でセーブします。</p> <p>測定中、セーブ中にその旨 表示します。</p> <p>測定対象のタッチキーを選択します。</p> <p>取り込むデータセット数を指定します。</p> <p>データの取り込み間隔を指定します。</p> <p>計測データを保管するファイルの名称を指定します。</p> <p>計測したデータセットの数を逐次更新して表示します。</p> <p>データ採取の最終日時を表示します。</p> <p>t+とt-のどちら(または両方)を容量相当値変換するか指定します。</p> <p>シリアル通信のCOM ポートの番号を指定します。</p> |
|--|---|

-
- | | |
|-------------------|--------------------------------|
| ⑫ ボーレート | シリアル通信のボーレートを指定します。 |
| ⑬ パリティ | シリアル通信のパリティ使用/不使用、偶数/奇数を指定します。 |
| ⑭ データ長 | シリアル通信のデータ長を指定します。 |
| ⑮ ストップビット数 | シリアル通信のストップビットの数を指定します。 |
| ⑯ 測定データ グラフ | 縦軸「測定データ」のグラフをプロットします。 |
| ⑰ 測定データ - 平均値 グラフ | 縦軸「測定データ - 平均値」のグラフをプロットします。 |

注1) 本サンプルプログラムで用いている UART によるシリアル通信は 2 線式通信ですが、S5U1C17656T(SVT17656)はデータを一方的に送る仕様で作られているため、PC 側の受信バッファが追従できなくなると通信データが欠落します。その検出はデータの上位、下位の値が範囲外になることで検知する方法で行っていますが、通信が追従できない旨のエラーが表示された場合、④、⑤、⑥の設定値を変更することで回避するようにしてください。

注2) シリアル通信に関する設定、⑪、⑫、⑬、⑭、⑮は、通常は変更する必要のない情報です。もしもこれらの情報を変更した場合、s1c17656_touch_oscillo の当該箇所の変更が必要になりますのでご注意ください。

注3) グラフ⑯、⑰の描画は全データ取得後に行われます。すなわち、データ取得時には逐次更新されず、以前のデータがプロットされている状態をキープします。

注4) UART による通信の基準クロックは、S5U1C17656T(SVT17656)の OSC3B(4MHz)をもとに作成しています。その OSC3B は、CR 発振を用いているので、使用環境により周波数がばらつきます。そのため、シリアル通信のタイミングがずれて正常な通信が行えない場合がありますので、その場合には設定パラメータを調整するなどして問題を回避してください。

6.2. s1c17656_touch_demo について

タッチキーのデモ用に、タッチキーの状態、容量相当値などを液晶表示するソフトウェア、s1c17656_touch_demo について説明します。

6.2.1. 操作手順

プロジェクトのインポート

(1) IDE を起動して、「s1c17656_touch_demo」プロジェクトをインポートしてください。

※ インポートの方法は、S5U1C17001C Manual “3.ソフトウェア開発手順”を参照してください。

ビルド

(1) IDE を使用して、「s1c17656_touch_demo」プロジェクトをビルドしてください。

接続、電源投入

(1) S5U1C17656T(SVT17656)、ICDmini、USB ケーブル、PC を接続してください。このとき、先の s1c17656_oscillo と同様な環境で操作を行うのなら、UART によるシリアル通信用の配線は取り除いてください。

(2) S5U1C17656T(SVT17656)、ICDmini をリセットしてください。

実行

(1) 必要に応じて、コマンドファイル (s1c17656_touch_demo_gnu17IDE.cmd) 内の fls17656.elf へのパスを変更してください。デフォルトでは、C:/EPSON/GNU17/mcu_model/17656/fls/fls17656.elf となっています。

※ コマンドファイルの修正方法は、S5U1C17001C Manual を参照してください。

(2) IDE を使用して、「s1c17656_touch_demo」プロジェクトを実行してください。

(3) 「s1c17656_touch_demo」の実行後、タッチキー動作のデモンストレーションを開始します。

6.2.2. サンプルプログラム操作概要

1. タッチボタン

⌘(Enter)、←(Back Space)、0~9 の 12 個のキー で構成されています。

2. LCD 表示

6 桁の 7 セグメント数字とドット、3 つの電池残マーク、ロック状態を示す L 表示、1 分間有効なトークンを似せた数値の寿命を 10 秒単位で示す 6 個のバー、2 つのコロン で構成されています。

3. 動作モードの切り替え

電源オン直後に、「INIT #」形式でカウントダウンしている間は、タッチキーのバックグラウンドレベルの情報を収集しています。基板周辺に手や体の一部を近づけないようにしてください。

キー	動作
N	LCDにL表示がされているロック状態の時、2秒以上 N にタッチし、KEY=表示がされたら、30秒以内に1、2、3、4、5、 N とタッチするとロック解除され、LCDのL表示が消えます。キーワードが違ったり、30秒以内に正しいキーワードの入力がなかったりした場合、再びロックされます。なお、何もキータッチがない状態で2分間経過した場合も、ロックモードに移ります。
←	数値入力時は、直前の入力値を取り消すバックスペースキーとして機能します。一番左側にカーソルがあるときにタッチすると、現在のコマンドをキャンセルします。
0	各キーのコマンドを説明するヘルプ動作を開始します。
1	トークンに似せた数値を表示します。(注1)
2	RTCの現在時刻を表示します。(注1)
3	日付を約4秒間表示したあと、2のRTCの現在時刻を表示するモードに移行します。
4	チャレンジレスポンスを模倣した動作です。任意の数値を入力して N をタッチすると、60秒間有効なトークンを表示した後に1のトークンモードに移行します。30秒以内に操作を完了しないときも、そのトークンモードに移行します。
5	故意にハングアップさせます。そのためWDTが動作してリセットされ、初期化から再開します。その際、RTCの日時と日付の情報は保持されますが、起動時にOSC1が停止するため、約1.5秒時刻が遅れますのでご注意ください。
6	キーにタッチした時のクリック音をオン/オフします。
7	日付、時刻を設定します。2015/01/09 12:34:56の場合、150109123456と入力します。60秒間過ぎるまでに入力を完了しないと、直前の動作モードに戻ります。ただし、まだ日時が入力がされていない状態で120秒経過した場合は、SLEEP表示後、CPUはHALTします。その場合はリセットするか電池を抜き差しするかし、初期化動作から行ってください。
8	乱数の種SEEDの値を変える際に使用します。数値入力後に N をタッチしてください。30秒間過ぎるまでに入力を完了しないと、直前の動作モードに戻ります。
9	手動でロック状態に移行させる場合、このキーをタッチしてください。

注 1: 直前の動作モードとして有効なのはこれら 2 つのモードだけです。すなわちロック状態ではどちらかの動作モードの表示が続きます。

6.2.3. 電池寿命の見積もり

サンプルプログラム `slc17656_touch_demo` を S5U1C17656T(SVT17656)上で実行した時の消費電流の実測値から、ボタン電池で駆動した場合の電池寿命を見積もりました。その際の計算条件は以下の通りです。

- ボタン電池： CR2032 (公称容量=220mAh) 1個
- タッチキー読み取り時間： 1日あたり 30分間

非ロック状態でのタッチキー読み取り時、1秒間に32回、全12個のキーに対してスキャンしています。またロック状態の時は、1秒間に2回、1個のキーのみタッチセンシングしています。各々の場合の平均消費電流を測定したところ、以下のような結果が得られました。

非ロック時： 32.9uA
 ロック時： 11.2uA

従い、電池の使用可能時間を計算すると、以下のようになります。

$$220 / ((32.9 \times (0.5 / 24) + 11.2 \times (23.5 / 24)) / 1000) / 24 / 365 = 2.15 \text{ 年}$$

セイコーエプソン株式会社

マイクロデバイス事業部 デバイス営業部

東京 〒191-8501 東京都日野市日野 421-8
TEL (042) 587-5313 (直通) FAX (042) 587-5116

大阪 〒541-0059 大阪市中央区博労町 3-5-1 エプソン大阪ビル 15F
TEL (06) 6120-6000 (代表) FAX (06) 6120-6100

ドキュメントコード:413001000
2015年4月 作成