

S1C17 Family Application Note

S1C17M02/M03

アプリケーションノート

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、予告な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します。
 2. 弊社製品のご購入およびご使用にあたりましては、事前に弊社営業窓口で最新の情報をご確認いただきますとともに、弊社ホームページなどを通じて公開される最新情報に常にご注意ください。
 3. 本資料に掲載されている応用回路、プログラム、使用方法などはあくまでも参考情報です。お客様の機器・システムの設計において、応用回路、プログラム、使用方法などを使用する場合には、お客様の責任において行ってください。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ならびに損害の発生に対し、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。
 4. 弊社は常に品質、信頼性の向上に努めていますが、一般的に半導体製品は誤作動または故障する場合があります。弊社製品のご使用にあたりましては、弊社製品の誤作動や故障により生命・身体に危害を及ぼすこと又は財産が侵害されることのないように、お客様の責任において、お客様のハードウェア、ソフトウェア、システムに必要な安全設計を行うようお願いいたします。なお、設計および使用に際しては、弊社製品に関する最新の情報（本資料、仕様書、データシート、マニュアル、弊社ホームページなど）をご確認いただき、それに従ってください。また、上記資料などに掲載されている製品データ、図、表などに示す技術的な内容、プログラム、アルゴリズムその他応用回路例などの情報を使用する場合は、お客様の製品単独およびシステム全体で十分に評価を行い、お客様の責任において適用可否の判断をお願いいたします。
 5. 弊社は、正確さを期すために慎重に本資料およびプログラムを作成しておりますが、本資料およびプログラムに掲載されている情報に誤りがないことを保証するものではありません。万一、本資料およびプログラムに掲載されている情報の誤りによってお客様に損害が生じた場合においても、弊社は一切その責任を負いかねます。
 6. 弊社製品の分解、解析、リバースエンジニアリング、改造、改変、翻案、複製などは堅くお断りします。
 7. 弊社製品は、一般的な電子機器（事務機器、通信機器、計測機器、家電製品など）に使用されること（一般用途）、および本資料に個別に掲載または弊社が個別に指定する用途に使用されること（指定用途）を意図して設計、開発、製造されています。これら一般用途および指定用途以外の用途（特別な品質、信頼性が要求され、その誤動作や故障により生命・身体に危害を及ぼす恐れ、膨大な財産侵害を引き起こす恐れ、もしくは社会に深刻な影響を及ぼす恐れのある用途。以下、特定用途といいます）に使用されることを意図していません。お客様に置かれましては、弊社製品を一般用途および指定用途に使用されることを推奨いたします。もし特定用途で弊社製品のご使用およびご購入を希望される場合、弊社はお客様が弊社製品を使用されることへの商品性、適合性、安全性について、明示的・黙示的に関わらずいかなる保証を行うものではありません。お客様が特定用途での弊社製品の使用を希望される場合は、弊社営業窓口まで事前にご連絡の上、承諾を得てください。
- 【特定用途（例）】
- 宇宙機器（人工衛星・ロケットなど）/ 輸送車両並びにその制御機器（自動車・航空機・列車・船舶など）
医療機器 / 海底中継機器 / 発電所制御機器 / 防災・防犯装置 / 交通用機器 / 金融関連機器
- 上記と同等の信頼性を必要とする用途。詳細は、弊社営業窓口までお問い合わせください。
8. 本資料に掲載されている弊社製品および当該技術を国内外の法令および規制により製造・使用・販売が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。また、弊社製品および当該技術を大量破壊兵器等の開発および軍事利用の目的その他軍事用途等に使用しないでください。弊社製品または当該技術を輸出または海外に提供する場合、「外国為替及び外国為替法」、「米国輸出管理規則（EAR）」、その他輸出関連法令を遵守し、係る法令の定めるところにより必要な手続きを行ってください。
 9. お客様が本資料に掲載されている諸条件に反したことに起因して生じたいかなる損害（直接・間接を問わず）に関して、弊社は一切その責任を負いかねます。
 10. お客様が弊社製品を第三者に譲渡、貸与などをしたことにより、損害が発生した場合、弊社は一切その責任を負いかねます。
 11. 本資料についての詳細に関するお問合せ、その他お気付きの点などがありましたら、弊社営業窓口までご連絡ください。
 12. 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。

評価ボード・キット、開発ツールご使用上の注意事項

1. 弊社評価ボード・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ください。本品は、完成品に対する設計品質に適合していません。
2. 弊社評価ボード・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ください。
3. 弊社評価ボード・キット、開発ツールに用いられる部品は、予告なく変更されることがあります。

Rev. j1.2, 2023. 4

目 次

1	概要	1
1.1	システム構成	1
2	システムリソース	3
2.1	マイクロコントローラリソース	3
2.2	仕様機能説明	3
3	動作説明	4
3.1	概要	4
3.2	測定設定の切り替え	5
3.2.1	測定モードの切り替え	5
3.2.2	ピークホールド設定の切り替え	7
3.2.3	測定レンジ設定の切り替え	7
3.2.4	オートレンジ設定の切り替え	7
3.2.5	データホールド設定の切り替え	7
3.2.6	リラティブ設定の切り替え	8
3.3	通信モード	8
3.3.1	通信方式	8
3.3.2	通信データ形式	11
3.4	ローバッテリーモード	12
3.5	オートパワーオフ機能	13
3.6	オートレンジ機能	13
3.7	データホールド機能	13
3.8	リラティブ機能	14
4	ソフトウェア説明	15
4.1	機能ブロック	15
4.2	動作概念図	16
4.3	s1c17m02_m03_application_gnu17v3 について	17
4.3.1	ファイル構成(src/)	17
4.3.2	ファイル構成(src/driver/)	18
4.3.3	ファイル構成(inc/)	18
4.3.4	使用する MCU の設定	19
4.3.5	使用する周辺回路の設定	19
4.3.6	モジュール説明	20
5	ソフトウェア動作概要	23
5.1	システム初期化	24
5.2	タイマ制御インターフェース	25
5.3	測定モード	26
5.3.1	測定モード切り替え	28
5.3.2	測定レンジ切り替え	28
5.3.3	ピークホールド切り替え	29
5.3.4	オートレンジ切り替え	29
5.3.5	データホールド切り替え	29
5.3.6	リラティブ切り替え	29
5.4	通信モード	29

5.5 計測データの演算機能	31
5.5.1 A/D 変換値からの演算	32
5.5.2 ピークホールドソフトウェア処理.....	33
5.5.3 T16B_DMM PWM タイマを用いた演算	34
5.5.4 キャリブレーションによる補正	37
Appendix A S1C17M03 から S1C17M02 への変更方法	39
改訂履歴表	44

1 概要

セイコーエプソン・デジタルマルチメータ（以下、DMM）向けリファレンスシステムは、セイコーエプソン製マイクロコントローラ **S1C17M03** を用いて、DMM を設計いただくためのソリューションパッケージです。

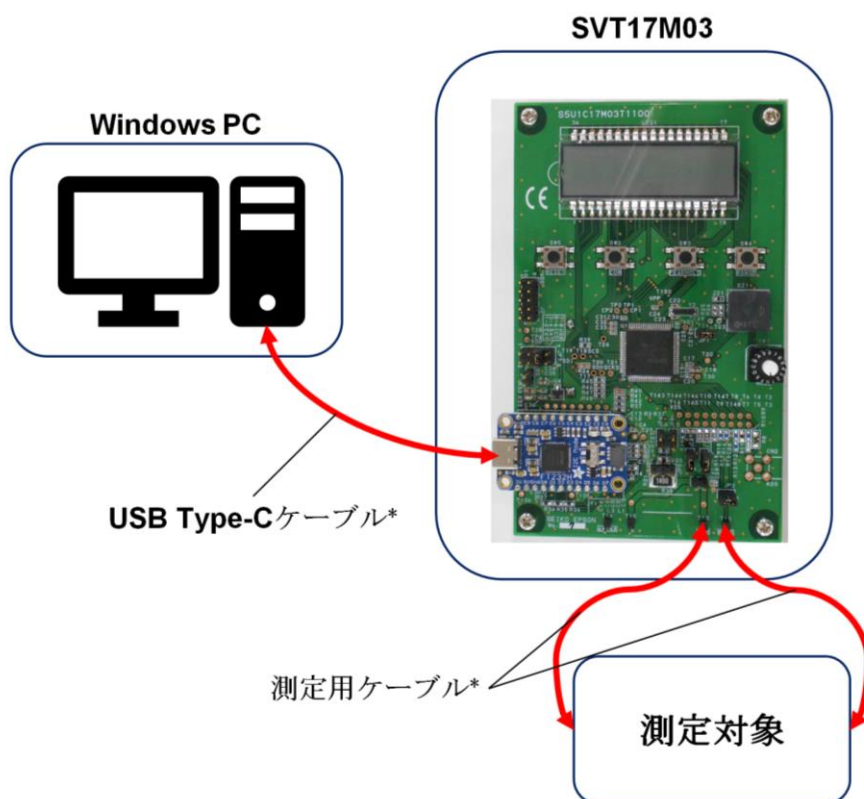
本パッケージは、**S5U1C17M03T(SVT17M03)**、リファレンスソフトウェア、アプリケーションノート（本ドキュメント）で構成され、DMM を少工数で効率よく設計いただくことを目的に提供しています。

なお、ソフトウェアは **S1C17M03** での利用に最適化されていますが、同製品シリーズの **S1C17M02** でもご利用いただくことができます。ただし、その場合はドライバ機能のみ利用が可能です。

本アプリケーションノートはソフトウェアパッケージ **S1C17M02_M03_demo_software_v2.X** に対応しています。

1.1 システム構成

本リファレンスシステムでは、**SVT17M03** に組み込まれたリファレンスソフトウェアによって、様々な回路パラメータの計測を行うことができます。また、シリアル通信によって計測データを簡単に PC に送信することができます。



*USB ケーブル、測定用ケーブルはお客様にてご用意下さい。

図 1.1-1 システム構成

表 1.1-1 提供物一覧

	構成品目	内容	内容説明
1	ハードウェア	S5U1C17M03T(SVT17M03)	評価基板
2	ソフトウェア	S1C17M02_M03_demo_software_vX.X.zip*	ソフトウェアパッケージ
3	ドキュメント	本ドキュメント	アプリケーションノート

*X：バージョン

表 1.1-2 ソフトウェアパッケージ内容物一覧

	項目	内容説明
1	s1c17m02_m03_application_verX.X* └ s1c17m02_m03_application_gnu17v3	プロジェクトフォルダ
2	DmmTools_X.XX* └ DmmEvalTool.exe └ 各種 dll ファイル	PC 通信用ソフトウェア
3	Dmm_readme_en.txt Dmm_readme_jp.txt	英文 readme ファイル 和文 readme ファイル
4	License_e.txt	ソフトウェア使用許諾契約書
5	S5U1C17M03Txxxx_jumper connections_en.xlsx* S5U1C17M03Txxxx_jumper connections_jp.xlsx*	英文ジャンパ設定表 和文ジャンパ設定表
6	DMM_Calibration_Sheet_en.xlsx DMM_Calibration_Sheet_jp.xlsx	英文キャリブレーション用ヘッ ダーファイル作成シート 和文キャリブレーション用ヘッ ダーファイル作成シート

*X：バージョン

2 システムリソース

2.1 マイクロコントローラリソース

表 2.1-1 マイクロコントローラリソース

項目	内容
CPU	S1C17M03
システムクロック	3.2MHz(OSC3 内部発振)

2.2 仕様機能説明

PPOINT P07 を SPI 通信のタイミング用信号として出力に使用します。
P40, P41, P42, P43 をロータリースイッチの入力に用います。
P44, P45, P46 をタクトスイッチの入力に用います。

DMM コントローラ DMM としての計測に使用します。

LCD4B 表示用 LCD を制御します。

SNDA ブザーによるビープ音再生に使用します。

SPIA Ch.0 UPMUX で PPOINT の P00 を #SPISSn、P01 を SDO、P03 を SPICLK に割り当てます。PC との通信に使用します。

T16 Ch.0 モード名表示用、DMM コントローラタイムアウト用に使用します。

T16 Ch.1 各種スイッチのポーリングスキャン用、LCD 表示切り替え用に使用します。

T16B_DMM Ch.0 周波数計算：周波数測定期間用に使用します。

T16B_DMM Ch.1 周波数計算：周波数測定期間中の測定波形信号のキャプチャに使用します。
T16B_DMM Ch.0 の tout 出力を CAP に割り当てます。

T16B_DMM Ch.2 周波数計算：測定波形の 1 周期中のカウントに使用します。
測定波形パルス信号を CAP に割り当てます。

SVD4 バッテリレベルの検出に用います。

UART3 Ch.0 UPMUX で PPOINT の P03 を USIN に、P02 を USOUT に割り当てます。
PC との通信に使用します。

クロック システムクロックは OSC3（内蔵発振：3.2MHz）を使用します。
LCD4B、SNDA、T16 Ch.0、T16B_DMM Ch.0、T16B_DMM Ch.2、SVD4 の駆動に OSC1 を使用します。
DMM コントローラ、T16 Ch.1 の駆動に OSC3 を使用します。

3 動作説明

3.1 概要

本システムは、SVT17M03 上の S1C17M03 に搭載された DMM コントローラや DMM 用 16 ビット PWM タイマを用いて、対応する測定レンジで計測を行います。本ソフトウェアでは S1C17M03 で測定可能なすべての測定設定を利用可能ですが、実際の測定では SVT17M03 の制約により利用できる測定モードに制限があります。詳しくは別途発行の『S5U1C17M03T マニュアル』を参照してください。

表 3.1-1 本ソフトウェアで利用可能な測定モードおよび測定レンジ

測定モード	測定レンジ
直流電圧	600 mV / 6 V / 60 V / 600 V / 1,000 V
交流電圧	600 mV / 6 V / 60 V / 600 V / 1,000 V
直流電流	600 μ A / 6 mA / 60 mA / 600 mA / 6 A / 10 A
交流電流	600 μ A / 6 mA / 60 mA / 600 mA / 6 A / 10 A
抵抗値(CC 方式)	600 Ω / 6 k Ω / 60 k Ω / 600 k Ω / 6 M Ω / 60 M Ω
抵抗値(CV 方式)	600 Ω / 6 k Ω / 60 k Ω
導通チェック	CV / CC
容量(CC 方式)	1 μ F / 10 μ F / 100 μ F / 1,000 μ F
容量(CV 方式)	10 nF / 100 nF
ダイオード	-
交流電圧周波数	6 V
交流電流周波数	6 mA
内部温度	-

上記の測定モードのうち、直流電圧、交流電圧、抵抗値(CC 方式)、抵抗値(CV 方式)、容量(CC 方式)、容量(CV 方式)については、線形補間によるキャリブレーション実施後の補正値が測定値として得られます。キャリブレーションの詳細については、『5.5.4 キャリブレーションによる演算』を参照してください。

測定した結果は下記の方法を通じて出力されます。

- SVT17M03 上 LCD パネルへの表示
- UART または SPI 通信による PC への送信
※内部温度測定では LCD パネルへの結果表示は行わず、測定レジスタ値のシリアル通信出力を行います。
- SVT17M03 上の導通チェックブザー再生

3.2 測定設定の切り替え

DMM としての計測動作を行う際、SVT17M03 に搭載されたタクトスイッチ(SW2～SW5)やロータリースイッチ(SW1)を操作することで、測定モード等の設定変更やシリアル通信による測定結果の送信を開始することができます。



図 3.2-1 タクトスイッチ(SW2～SW5)



図 3.2-2 ロータリースイッチ(SW1)

表 3.2-1 各スイッチを押したときの動作

スイッチ		動作
SW1		測定モードの切り替え
SW2		通信モードの開始/終了 データホールド機能の切り替え(ON→OFF) スリープ状態からの復帰
SW3		ピークホールド設定の切り替え スリープ状態からの復帰
SW4		測定レンジ設定の切り替え オートレンジ機能の切り替え(ON→OFF) リラティブ機能の切り替え(ON→OFF) スリープ状態からの復帰
SW5		ハードリセットの実行
SW3 長押し	SW2	データホールド機能の切り替え(OFF→ON)
	SW4	リラティブ機能の切り替え(OFF→ON)
SW4 長押し		オートレンジ機能の切り替え(OFF→ON)

3.2.1 測定モードの切り替え

初回起動時および、計測モードまたは通信モード中に SVT17M03 上のロータリースイッチ(SW1)を回転させることで、DMM コントローラの測定モード設定を、スイッチの番号に対応した値に設定することができます。切り替え時、SVT17M03 上の LCD に対象の測定モード名が 1s の間表示されます。また、測定レンジはデフォルトレンジが設定されます。



図 3.2.1-1 直流電圧測定に切り替え時の LCD 画面

表 3.2.1-1 SW1 の番号と対応する測定モード

SW1 番号	測定モード	デフォルトレンジ	モード名
0	直流電圧測定	6V	DCV
1	交流電圧測定	6V	ACV
2	直流電流測定	6mA	DCI
3	交流電流測定	6mA	ACI
4	抵抗値測定(CC 方式)	600Ω	OHM CC
5	抵抗値測定(CV 方式)	600Ω	OHM CV
6	導通チェック	CV	CONT
7	容量測定(CC 方式)	1uF	CAP CC
8	容量測定(CV 方式)	10nF	CAP CV
9	ダイオード VF 測定	-	DIODE
A	交流電圧周波数測定	6V	FREQ ACV
B	交流電流周波数測定	6mA	FREQ ACI
C	内部温度測定	-	TEMP

※未使用の番号が選択された場合、計測動作は行わず、LCD には"NOFUNC"と表示されます。

測定モードの表示が終わると、DMM としての計測動作を行います。計測値はボード上の LCD に下記の様に表示されます。また、導通チェックモードの場合は導通時にボード上のブザーが再生されます。

ピークホールド/データホールド/
リラティブ表示

オートレンジ表示



図 3.2.1-2 計測値表示 LCD 画面

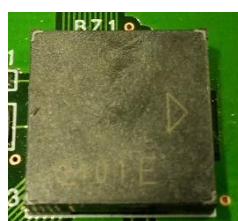


図 3.2.1-3 ブザー

3.2.2 ピークホールド設定の切り替え

計測モード中にタクトスイッチ(SW3)を押す度に、ピークホールド機能を1段階ずつ切り替えることができます。LCD画面左上の『P』の表示から、有効になっている設定を確認することができます。

*オートレンジ機能使用時はピークホールド使用不可

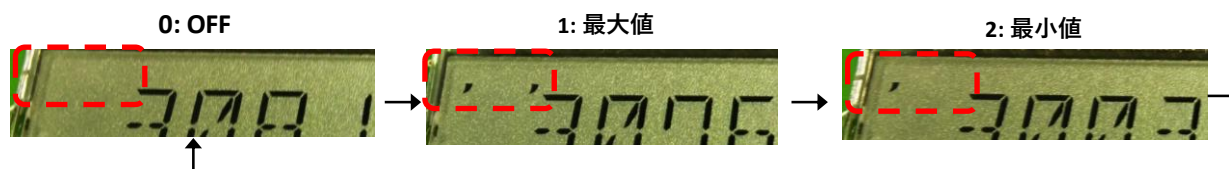


図 3.2.2-1 ピークホールド設定の切り替え時 LCD 表示

3.2.3 測定レンジ設定の切り替え

計測モード中にタクトスイッチ(SW4)を押す度に、設定可能な範囲の中で、測定レンジを1段階ずつ切り替えることができます。

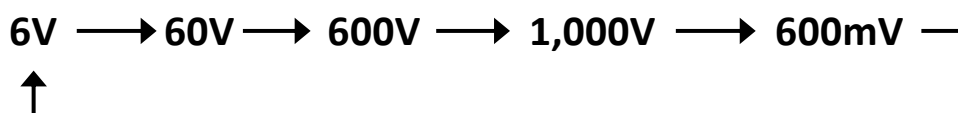


図 3.2.3-1 例) 直流電圧測定の測定レンジ切り替え

3.2.4 オートレンジ設定の切り替え

計測モード中にタクトスイッチ(SW4)を長押しすることで、オートレンジ機能がONになり、LCD画面右上に『R』が表示されます。オートレンジ機能がONの状態ではSW4を押すと、オートレンジ機能はOFFになります。



図 3.2.4-1 オートレンジ設定の切り替え時 LCD 表示

3.2.5 データホールド設定の切り替え

タクトスイッチ(SW3)を長押ししながら、タクトスイッチ(SW2)を押すことで、データホールド機能がONになり、LCD画面の左上に『H』が3つ表示されます。データホールド機能がONの状態ではSW2を押すと、データホールド機能はOFFになります。



図 3.2.5-1 データホールド設定の切り替え時 LCD 表示

3.2.6 リラティブ設定の切り替え

データホールド機能が ON の状態でタクトスイッチ(SW3)を長押ししながら、タクトスイッチ(SW4)を押すことで、リラティブ機能が ON になり、LCD 画面の左上に『』が 4 つ表示されます。リラティブ機能が ON の状態で SW4 を押すと、リラティブ機能は OFF になります。



図 3.2.6-1 リラティブ設定の切り替え時 LCD 表示

3.3 通信モード

通信モードに対応した計測モードを実行中に SVT17M03 上のタクトスイッチ(SW2)を押すことで、通信モードが開始され、シリアル通信による計測結果の送信待機状態になります。また、通信中モード中に規定数の計測データが送信完了するか、ボード上のタクトスイッチ(SW2)またはロータリースイッチ(SW1)が回転されると通信モードを終了します。通信モード中は LCD に"COM"と表示されます。



図 3.3-1 通信モード中の LCD 表示

3.3.1 通信方式

本ソフトウェアでは、通信の方式として UART または SPI による通信が可能です。通信方式の変更には『s1c17m02_m03_application_gnu17v3』のソースコードについて、下記記述を変更する必要があります。

- UART 通信を行う場合
src¥mid_communication.h の以下記述をコメントアウトする。
#define COMMUNICATION_MODE_SPI
- SPI 通信を行う場合
src¥mid_communication.h の以下記述がコメントアウトされている場合、コメントアウトを外す。
#define COMMUNICATION_MODE_SPI

3.3.1.1 UART 通信

UART 通信が有効化された状態で通信モードが開始されると、UART による計測結果の送信が開始されます。

(1) 通信条件

表 3.3.1.1-1 UART 通信条件

項目	設定値
ボーレート	230,400 bps
データ長	8 bit
ストップビット	1 bit
パリティ	無し
フロー制御	無し

(2) 動作確認手順

1. SVT17M03上のロータリースイッチやタクトスイッチによって、データ通信する測定の設定を行います
2. PCとボードをUSB Type-Cケーブルで接続します
3. PC上のターミナルソフト(Tera Termなど)を起動してSVT17M03に接続します
4. ボード上のタクトスイッチ(SW2)を押して通信モードを開始し、データを送信します
5. ターミナルソフトが受信データを表示します

3.3.1.2 SPI 通信

SPI 通信による計測データの通信には、SVT17M03 の操作と同時に、PC 通信用ソフトウェアである『DmmEvalTool』を操作する必要があります。SPI 通信が有効化された状態でスイッチ操作により通信モードが開始されると、SVT17M03 は送信待機状態となります。この状態で DmmEvalTool 上の START ボタンをクリックすることで通信が実行されます。

(1) DmmEvalTool

DmmEvalTool はソフトウェアパッケージに含まれます。パッケージ内『DmmTools_X.XX』（X：バージョン）フォルダ下の『DmmEvalTool.exe』をダブルクリックすることで起動できます。

DmmEvalTool を起動すると下記のウィンドウが PC 画面に表示されます。

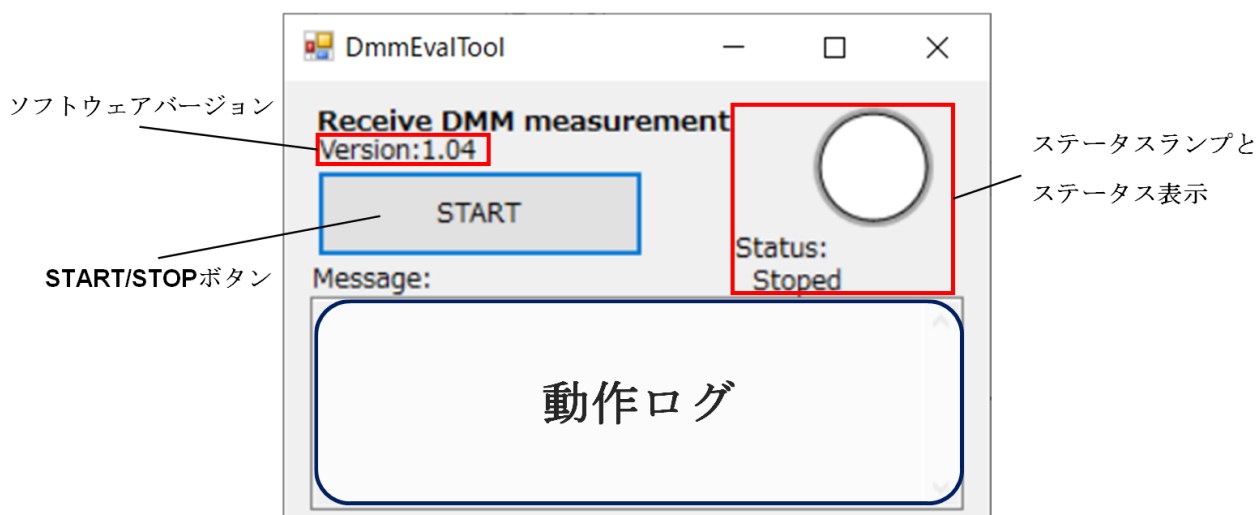


図 3.3.1.2-1 DmmEvalTool 画面

後記の方法に従って通信が完了すると、DmmEvalTool は実行ファイルと同じディレクトリ上に『Logs』フォルダを作成し、内部に『Output.csv』という名前で測定データを保存します。作成される『Output.csv』に記録されるデータフォーマットは下記の通りです。測定を複数回行うと、二回目以降は『Output.csv』の2列目以降に記録・更新されます。

表 3.3.1.2-1 測定データ記録フォーマット

行	内容
1 行目	測定数
2 行目	測定モード
3 行目	測定レンジ
4 行目	ピークホールド設定
以降	測定データ

(2) 通信条件

表 3.3.1.2-2 SPI 通信条件

項目	設定値
接続形態	スレーブ
データ順序	MSB 先頭
SPI モード	Mode3
データ長	8bit

(3) 動作確認手順

1. SVT17M03上のロータリースイッチやタクトスイッチによって、データ通信する測定の設定を行います
2. PCとボードをUSB Type-Cケーブルで接続します

3. DmmEvalToolを起動してSVT17M03に接続します
4. ボード上のタクトスイッチ(SW2)を押して通信モードを開始し、送信待機状態にします
5. DmmEvalToolのSTARTボタンをクリックし、通信を開始します
6. 通信が完了するとDmmEvalToolが受信した計測データのCSVファイルを生成します

3.3.1.3 SPI データ通信

本ソフトウェアでは DmmEvalTool を用いたデータ通信において、SPI 通信プロトコルの一部を改変し、本ソフトウェア独自の手法で通信を行います。

本ソフトウェアでは、SPI 通信における SDI 信号を使用せず、SDO 信号のみを使用する一方通行の通信を行います。そこで、データ通信の開始と終了を通信相手に伝えるため、GPIO の出力を 1 ポート利用してタイミング信号とします。これを利用し、通信開始時は LOW、通信終了時は HIGH とすることでデータ通信タイミングを通知します。GPIO の出力には P07 を使用します。

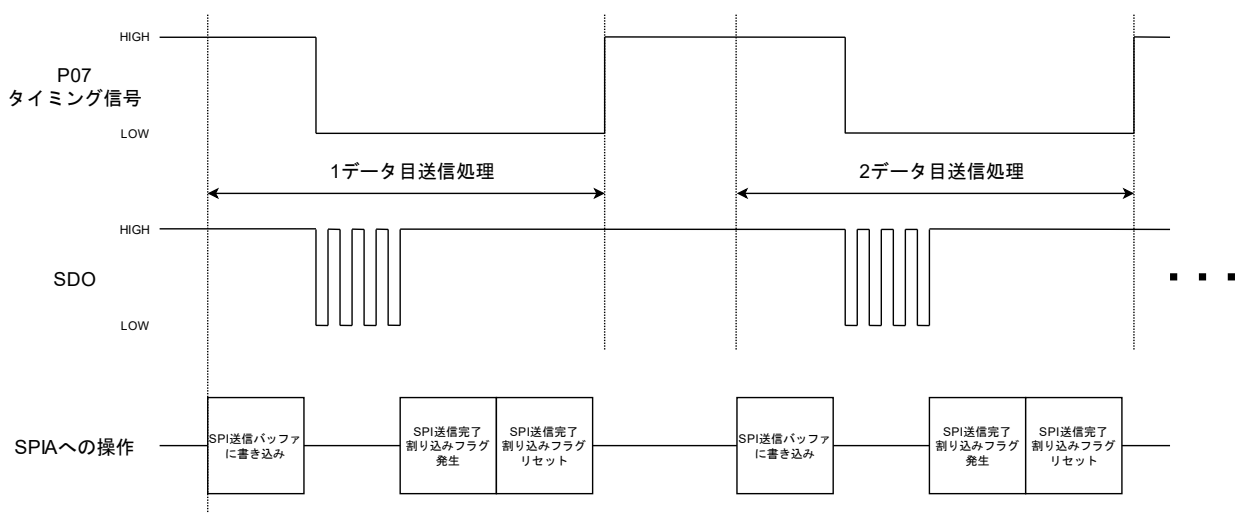


図 3.3.1.3-1 SPI によるデータ通信

3.3.2 通信データ形式

通信モードで SVT17M03 から送信されるデータは測定モードデータ、計測データ、通信終了データの3つのデータが連続した構成となっており、下記のデータフォーマットを用いて送信されます。

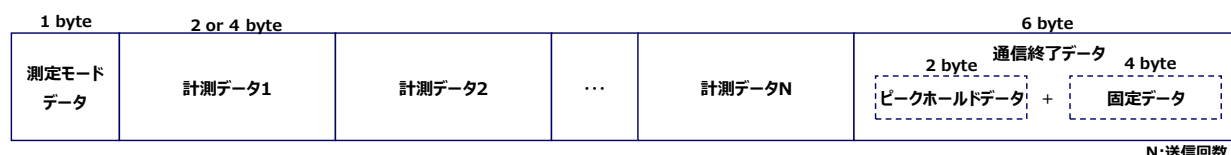


図 3.3.2-1 通信データフォーマット

3.3.2.1 測定モードデータ

通信モードが開始されると、1 バイトの測定モードデータを送信します。測定モードデータには、計測情報を識別するためのデータが下記の通り構成されており、測定モードデータを構成する測定レンジや測定モードの各情報は、DMM コントローラの各計測設定レジスタ値に対応しています。

表 3.3.2.1-1 測定モードデータフォーマット

#Bit	内容
7	0
6:4	DSADC16CTL.RANGESEL[3:0]ビット(測定レンジ)
3:0	DSADC16CTL.FUNCSEL[3:0]ビット(測定モード)

3.3.2.2 計測データ

測定モードデータの送信が完了すると、次に計測データが送信されます。計測データは計測結果レジスタ値または演算された計測値が下記の通り構成されています。なお、計測結果レジスタ値の詳細については別途発行の『S1C17M02/M03 テクニカルマニュアル』を参照ください。

計測データは上述した通り、測定モードごとに固有のデータ数が送信完了するか、タクトスイッチ(SW2)が押される、またはロータリースイッチ(SW1)が回されるまで送信されます。

表 3.3.2.2-1 計測データ構成

測定モード	データ長	データ型	送信データ数
直流電圧測定	2 byte	符号あり整数型	500
交流電圧測定	4 byte	符号なし整数型	250
直流電流測定	2 byte	符号あり整数型	500
交流電流測定	4 byte	符号なし整数型	250
抵抗値測定(CC 方式)	2 byte	符号あり整数型	500
抵抗値測定(CV 方式)	2 byte	符号あり整数型	500
導通チェック		通信未対応	
容量測定(CC 方式)	4 byte	単精度浮動小数点型	100
容量測定(CV 方式)	4 byte	単精度浮動小数点型	100
ダイオード VF 測定	2 byte	符号あり整数型	500
交流電圧周波数測定	4 byte	単精度浮動小数点型	100
交流電流周波数測定	4 byte	単精度浮動小数点型	100
内部温度測定	2 byte	符号あり整数型	500

3.3.2.3 通信終了データ

規定回数の計測データが送信されるか、タクトスイッチ(SW2)が押される、またはロータリースイッチ(SW1)が回されると、通信終了データが送信され通信モードを終了します。

通信終了データは2バイトのピークホールド機能に関するデータと4バイトの固定データの計6バイトデータで構成されており、それぞれの構成は下記の通りとなっています。

表 3.3.2.3-1 通信終了データ構成

ピークホールド機能	ピークホールドデータ	固定データ
最小値ピークホールド	FF 0D	5A 96 96 5A
最大値ピークホールド	FF 0E	5A 96 96 5A
OFF	FF 0F	5A 96 96 5A

3.4 ローバッテリーモード

本ソフトウェア動作中、電源電圧検出回路(SVD4)によって SVT17M03 に供給される電源電圧が 2.2V を下回ったことが検知されると、ローバッテリーモードが開始されます。本モードでは、LCD への表

示に必要な回路を除く他すべての周辺回路機能へクロックの供給が停止し、下図の様に LCD に“LOW BATT”と表示することでユーザーに対し、電池交換を促します。



図 3.4-1 ローバッテリーモード中の LCD 表示

3.5 オートパワーオフ機能

本ソフトウェアでは、一定時間(デフォルトは 60 s、ソフトウェアで設定可能)タクトスイッチの操作が無かった場合に、自動的に LCD が消灯し、マイコンがスリープモードへと移行します。タクトスイッチの SW2~SW4 のいずれかを押すことによって、スリープモードは解除されます。

3.6 オートレンジ機能

本ソフトウェアでは、計測値に応じて、自動的にレンジを切り替えるオートレンジ機能が実装されています。タクトスイッチ (SW4)を長押しすることで、オートレンジ機能は ON になります。また、オートレンジ機能が ON の状態で、SW4 を押すことによって、オートレンジ機能は OFF になります。各測定モードにおけるオートレンジ機能の対応状況は、以下の通りです。

表 3.6-1 オートレンジ機能対応の測定モード

測定モード	オートレンジ機能対応
直流電圧測定	△ ^{*1}
交流電圧測定	△ ^{*1}
直流電流測定	△ ^{*2}
交流電流測定	△ ^{*2}
抵抗値測定(CC 方式)	○
抵抗値測定(CV 方式)	○
導通チェック	×
容量測定(CC 方式)	○
容量測定(CV 方式)	○
ダイオード VF 測定	×
交流電圧周波数測定	×
交流電流周波数測定	×
内部温度測定	×

*1 測定レンジによって、ジャンパの設定を切り替える必要があるため、オートレンジ機能が正しく機能しない場合があります。

*2 測定レンジによって、ジャンパの設定および測定端子を切り替える必要があるため、オートレンジ機能が正しく機能しない場合があります。

3.7 データホールド機能

本ソフトウェアでは、測定中に、SW3 を長押しした状態で、SW2 を押すことで、データホールド機能が ON になります。データホールド機能が ON になると、その時点での計測値がホールドされ、かつ LCD にホールドした値が表示され続けます。

3.8 リラティブ機能

本ソフトウェアでは、データホールド機能が **ON** かつ **SW3** を長押した状態で、**SW4** を押すことで、計測値とデータホールドした値との差分値を表示するリラティブ機能が **ON** になります。リラティブ機能使用時は、オートレンジ機能および測定レンジの切り替え機能は使用できません。

4 ソフトウェア説明

4.1 機能ブロック

本ソフトウェアは、周辺回路を直接動作するドライバ層、DMM 動作を実現するアプリケーション層、及びその間のインターフェースとなるミドル層で構成されます。

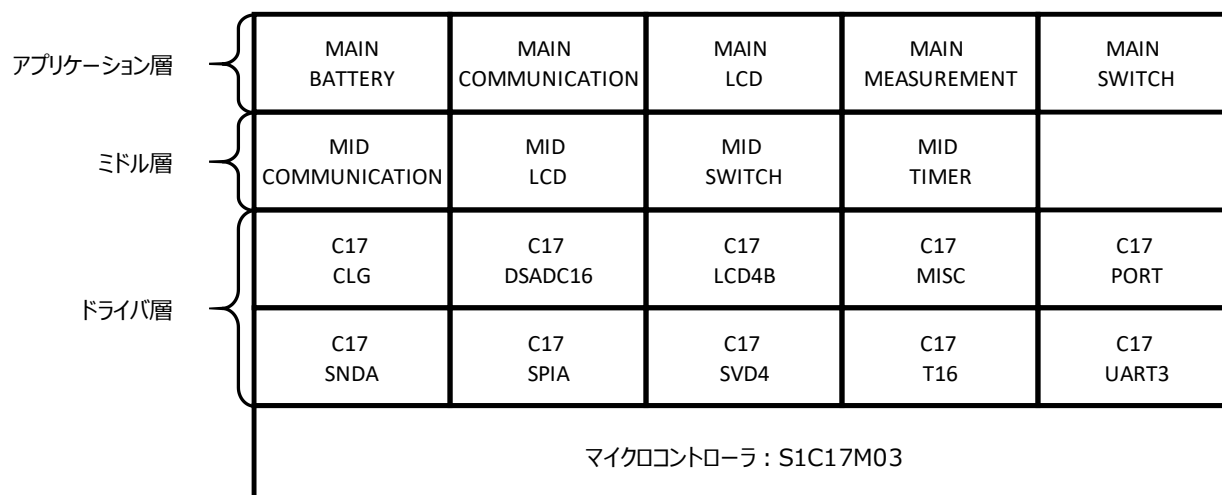


図 4.1-1 機能ブロック

表 4.1-1 機能ブロック概要

機能ブロック	概要
MAIN BATTERY	ローバッテリーモード
MAIN COMMUNICATION	通信モード
MAIN LCD	各種モード用 LCD 表示
MAIN MEASUREMENT	測定モード
MAIN SWITCH	モード切り替え
MID COMMUNICATION	通信インターフェース
MID LCD	LCD インターフェース
MID SWITCH	スイッチインターフェース
MID TIMER	タイマインターフェース
C17 CLG	CLG ドライバ
C17 DSADC16	DMM コントローラドライバ
C17 LCD4B	LCD4B ドライバ
C17 MISC	MISC ドライバ
C17 PORT	PPORT ドライバ
C17 SNDA	SNDA ドライバ
C17 SPIA	SPIA ドライバ
C17 SVD4	SVD4 ドライバ
C17 T16	T16 ドライバ
C17 UART3	UART3 ドライバ

4.2 動作概念図

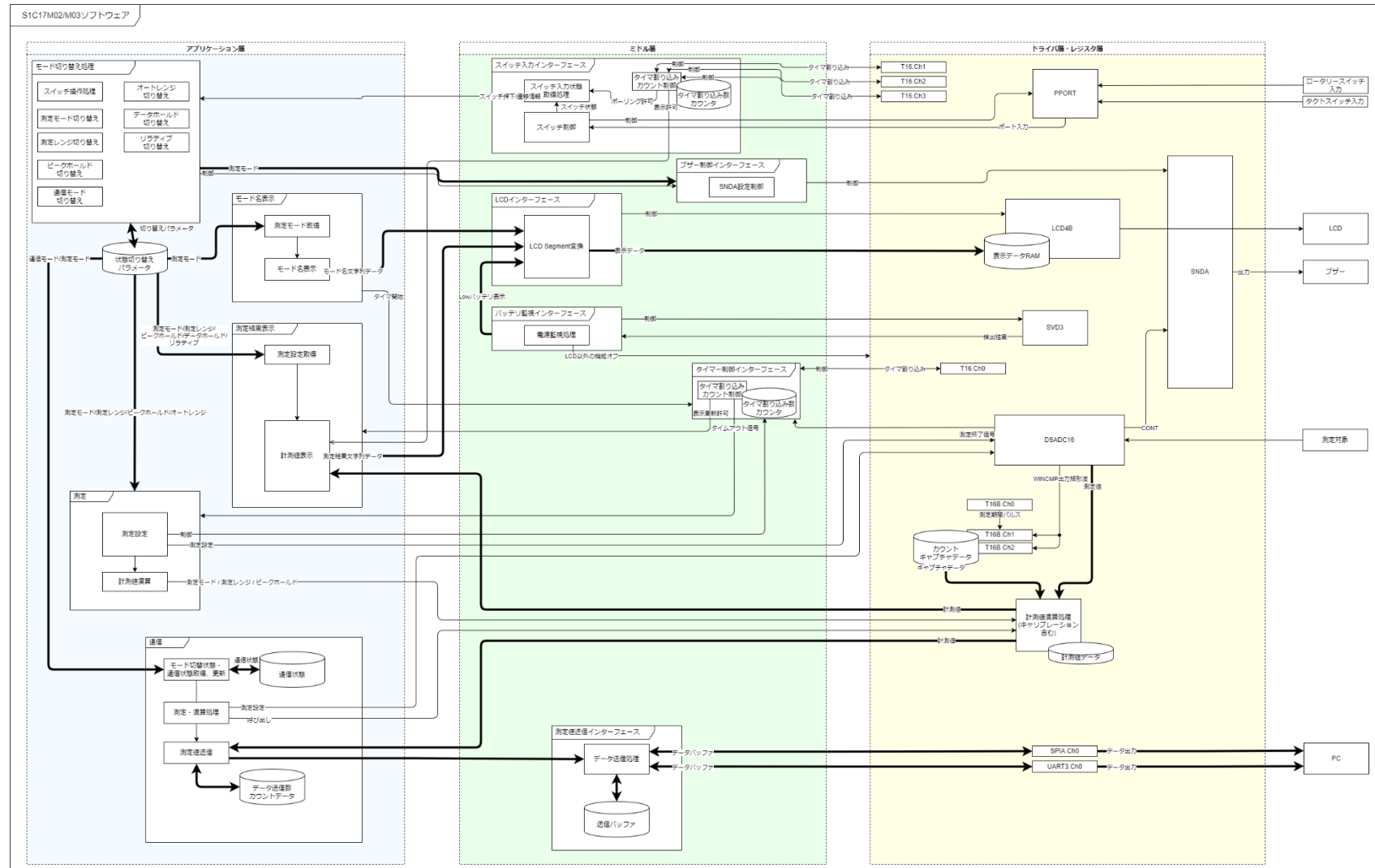


図 4.2-1 ソフトウェア動作概念図

4.3 s1c17m02_m03_application_gnu17v3 について

DMM 動作を実現するソフトウェア『s1c17m02_m03_application_gnu17v3』について説明します。

4.3.1 ファイル構成(src/)

ファイル名の先頭のプレフィックスは、アプリケーション層(main_)、ミドル層(mid_)、ドライバ層(c17_)を表します。

表 4.3.1-1 アプリケーション層ファイル一覧

ファイル名	機能
main_battery.c/.h	ローバッテリーモードプログラムファイル
main_communication.c/.h	通信モードプログラムファイル
main_lcd.c/.h	各種モード用 LCD 表示プログラムファイル
main_measurement.c/.h	測定モードプログラムファイル
main_switch.c/.h	モード切り替えプログラムファイル
main.c	メインルーチンプログラムファイル

表 4.3.1-2 ミドル層ファイル一覧

ファイル名	機能
lcd_font.h	LCD 表示用フォント定義ファイル
mid_communication.c/.h	通信インターフェースプログラムファイル
mid_lcd.c/.h	LCD インターフェースプログラムファイル
mid_switch.c/.h	スイッチインターフェースプログラムファイル
mid_timer.c/.h	タイマインターフェースプログラムファイル

4.3.2 ファイル構成(src/driver/)

driver フォルダには S1C17M02/M03 周辺回路ドライバファイルが格納されています。

表 4.3.2-1 ドライバ層ファイル一覧

ファイル名	機能
boot.c	デバイス起動時初期化プログラムファイル
c17_clg.c/.h	CLG ドライバプログラムファイル
c17_dsadc16.c/.h	DMM コントローラドライバプログラムファイル
c17_eepromc.c/.h	EEPROMC ドライバプログラムファイル
c17_i2c.c/.h	I2C ドライバプログラムファイル
c17_init_config.h	C17 ドライバコンフィグレーションファイル
c17_lcd4b.c/.h	LCD4B ドライバプログラムファイル
c17_misc.c/.h	MISC ドライバプログラムファイル
c17_port.c/.h	PPORT ドライバプログラムファイル
c17_pwg.c/.h	PWG ドライバプログラムファイル
c17_sndc.c/.h	SNDA ドライバプログラムファイル
c17_spia_asm.s	SPI データ送受信時アセンブリファイル
c17_spia.c/.h	SPI ドライバプログラムファイル
c17_svd4.c/.h	SVD4 ドライバプログラムファイル
c17_t16.c/.h	T16 ドライバプログラムファイル
c17_uart3.c/.h	UART3 ドライバプログラムファイル
c17_wdt2.c/.h	WDT2 ドライバプログラムファイル
crt0.h	C ランタイムファイル
libfloat.c/.h	数値・文字列変換プログラムファイル
strnum.c/.h	文字列作成・連結プログラムファイル

4.3.3 ファイル構成(inc/)

inc フォルダには S1C17M02/M03 周辺回路のレジスタ定義ファイルが格納されています。

表 4.3.3-1 inc フォルダ内ファイル一覧

ファイル名	機能
reg/	S1C17M02/M03 周辺回路レジスタ定義ファイル格納フォルダ
c17_mcu_select.h	MCU 選択用ヘッダーファイル
c17m02_reg.h	S1C17M02 周辺回路レジスタ定義ファイル
c17m03_reg.h	S1C17M03 周辺回路レジスタ定義ファイル

4.3.4 使用する MCU の設定

本ソフトウェアでは『inc/c17_mcu_select.h』内の下記マクロ記述について、どちらか片方を有効にすることで動作させる MCU を指定することができます。

```
#define C17_MCUSEL_M02      S1C17M02 で動作させる場合に有効にしてください
#define C17_MCUSEL_M03      S1C17M03 で動作させる場合に有効にしてください
```

S1C17M02 に無い周辺機能を S1C17M03 で使用しているため、使用する MCU を S1C17M02 に指定した場合、一部の機能が使用不可となります。各 MCU で使用可能なソフトウェア機能一覧及び Flash リソースは次の通りです。

表 4.3.4-1 使用可能なソフトウェア機能

機能ブロック	S1C17M02	S1C17M03
アプリケーション層	△*	○
ミドル層	×	○
ドライバ層	○	○

*基本的に使用できませんが、main_measurement.c は一部編集を行うことで利用できます。

表 4.3.4-2 Flash リソース

機能ブロック	S1C17M02 [byte]	S1C17M03 [byte]
アプリケーション層	-	9,714
ミドル層	-	2,669
ドライバ層	12,532	16,936
標準ライブラリ	-	12,890

4.3.5 使用する周辺回路の設定

本ソフトウェアでは『src/c17_init_config.h』内のマクロ記述により、使用する周辺回路の設定を行うことができます。C17_PE_XX で定められるマクロを有効化することで、マクロ名に対応する周辺回路の関数が利用可能になります。

表 4.3.5-1 周辺回路関数の有効化マクロ

マクロ名	対応する機能
C17_PE_DSADC16	DMM コントローラドライバ
C17_PE_EEPROMC	EEPROMC ドライバ
C17_PE_I2C_MASTER	I2C ドライバのマスターモード機能
C17_PE_I2C_SLAVE	I2C ドライバのスレーブモード機能
C17_PE_LCD4B	LCD4B ドライバ
C17_PE_PORT	PPOINT ドライバ
C17_PE_PWG	PWG ドライバ
C17_PE_SNDA	SNDA ドライバ
C17_PE_SPIA_SLAVE	SPIA ドライバのスレーブモード機能
C17_PE_SPIA_MASTER	SPIA ドライバのマスターモード機能
C17_PE_SVD4	SVD4 ドライバ
C17_PE_T16	T16 ドライバ
C17_PE_UART3	UART3 ドライバ
C17_PE_WDT2	WDT2 ドライバ

4.3.6 モジュール説明

ファイル中のモジュールについて、DMM を構成する機能を中心に、関数名とその機能を説明します。

4.3.6.1 main_battery.c

関数名	機能
initBattery()	バッテリー監視に関係する機能を初期化します。
SvdIntHandler()	電源電圧検出回路の割込みハンドラです。 LCD 表示に必要な周辺回路以外を停止させ、LCD にローバッテリー表示します。
stopPeripheralExceptDispOnLcd()	LCD 表示に必要な周辺回路以外の周辺回路を停止します。
dispLowBattOnLcd()	LCD にローバッテリー表示します。

4.3.6.2 main_communication.c

関数・グローバル変数名	機能
runComMode()	通信モードの機能を実行するメインのルーチンです。
updateComState()	通信の状態を管理します。
setComStartDataToBuf()	通信の開始データを送信データバッファに格納します。
setMeasDataToBuf()	計測データを送信データバッファに格納します。
setComEndDataToBuf()	通信の終了データを送信データバッファに格納します。
sentCount	合計データ送信数

4.3.6.3 main_lcd.c

関数名	機能
dispModeNameOnLcd()	モード名および測定モード名を LCD に表示します。
dispResultOnLcd()	測定結果を LCD に表示します。

4.3.6.4 main_measurement.c

関数・グローバル変数名	機能
runMeasMode()	測定モードの機能を実行するメインのルーチンです。
stopMeasurement()	測定を停止します。
startMeasurement()	測定を開始します。
result	演算後の計測値変数
switchRange()	オートレンジでのレンジ切り替えを行います。

4.3.6.5 main_switch.c

関数名	機能
switchMeasMode()	測定モードの遷移を行います。
switchMeasRange()	測定レンジの遷移を行います。
switchPeakHold()	ピークホールドの遷移を行います。
switchAutoRange()	オートレンジの遷移を行います。
switchDataHold()	データホールドの遷移を行います。
switchRelativeValue()	リラティブ機能の遷移を行います。

4.3.6.6 main.c

関数名	機能
main()	DMM 機能を実現するメインのルーチンです。
initPeripheral()	使用するすべての周辺回路およびインターフェースの初期化を行います。
setClg()	クロックの設定を行います。
checkLcdDisp()	LCD の全セグメントを点灯させます。
runAutoPowerOff()	一定時間スイッチ操作が無い場合に、MCU をスリープさせます。
runAutoRange()	オートレンジによってレンジが切り替わった際に、LCD 表示を更新します。
dispDataHold()	データホールドが ON 時の LCD 表示を更新します。
dispRelativeValue()	リラティブ機能が ON 時の LCD 表示を更新します。

4.3.6.7 mid_communication.c

関数・グローバル変数名	機能
initCommunication()	通信インターフェースを初期化します
startCommunication()	通信インターフェースの動作を開始します。
stopCommunication()	通信インターフェースの動作を停止します。
initBuf()	送信データバッファを初期化します。
sendTxDataToBuf()	送信データバッファにデータを格納します
sendTxBuf()	送信データバッファの内容を送信します。
c17sendSpiaData()	SPI 通信によりデータを送信します。
comMode	現在の通信モード状態変数

4.3.6.8 mid_lcd.c

関数名	機能
putStrOnLcd()	任意の文字列を LCD に表示します。
putChrOnLcd()	任意の文字を LCD に表示します。
putQuoteOnLcd()	任意の位置の『』LCD 表示についてオン/オフを制御します。
getLcdVramAddr()	任意の LCD セグメントの VRAM アドレスを取得します。

4.3.6.9 mid_switch.c

関数・グローバル変数名	機能
initSwitch()	スイッチインターフェースを初期化します。
updateSwitch()	スイッチの状態遷移を監視します。
getTactileSwitchInput()	すべてのタクトスイッチの入力を取得します。
getRotarySwitchInput()	ロータリースイッチの入力を取得します。
transition	スイッチ状態遷移構造体変数
checkLongPress	スイッチが長押しされたかを判定します。

4.3.6.10 mid_timer.c

関数・グローバル変数名	機能
initTimer()	タイマインターフェースを初期化します。
startTimer()	指定のタイマについてカウントを開始します。
stopTimer()	指定のタイマについてカウントを停止・リセットします。
T16Ch0IntHandler()	T16Ch0 の割込みハンドラです。時間の生成を行います。
T16Ch1IntHandler()	T16Ch1 の割込みハンドラです。時間の生成を行います。
T16Ch2IntHandler()	T16Ch2 の割込みハンドラです。時間の生成を行います。
T16Ch3IntHandler()	T16Ch3 の割込みハンドラです。時間の生成を行います。
modeNameDispFlg	モード名表示用タイマフラグ変数
timeOutFlg	タイムアウトタイマフラグ変数
swPollingFlg	スイッチポーリング用タイマフラグ変数
lcdDispRefreshFlg	LCD 表示更新用タイマフラグ変数
powerOffTimeOutFlg	オートパワーオフ用タイマフラグ変数
longPressFlg	長押し判定用タイマフラグ変数
rangeUpFlg	レンジアップ用タイマフラグ変数

4.3.6.11 c17_dsadc16.c

関数・グローバル変数名	機能
c17calcMeasValue()	計測値の計算を行います。
c17calcCapasitance()	容量値の計算を行います。
c17intT16bCh1()	T16B_DMM Ch1 の割込みハンドラです。周波数の計算に必要な値のカウントおよび周波数の計算を行います。
c17intT16bCh2()	T16B_DMM Ch2 の割込みハンドラです。周波数の計算に必要な値のカウントを行います。
c17dsadc16AdcEndFlg	A/D 変換終了フラグ変数
c17dsadc16OverWriteFlg	オーバーライトフラグ変数
t16bCh1IntCapFlg	T16B_DMM Ch1 キャプチャフラグ変数
frequency	演算周波数値変数
meas	測定設定構造体変数
acPeakMax	ピークホールド最大値変数
acPeakMin	ピークホールド最小値変数
t16bint_occurred_Flg	T16B 割り込み発生フラグ変数
noMeasurementFlg	未計測フラグ変数

5 ソフトウェア動作概要

本ソフトウェアには、DMM 機能を実現するために SVT17M03 を制御する機能が含まれています。ソフトウェアに含まれる動作については『図 4.2-1 ソフトウェア動作概念図』を参照してください。

主なソフトウェア機能

- DMM モード動作の制御機能
- 計測データの演算機能
- 計測値のシリアル送信機能

5.1 システム初期化

ここでは、MCU を動作させるためのクロックの設定を含め、DMM 動作に必要な周辺回路機能およびインターフェースをまとめて初期化する処理をしています。本ソフトウェアにおける各種周辺回路の設定値は下記の通りで、ドライバの `c17_init_config.h` から任意の値に変更することができます。ここでは実際に本ソフトウェアで使用する周辺回路のみ記載しています。

表 5.1-1 各種周辺回路設定値一覧(1)

周辺回路	設定項目	設定値
CLG	システムクロック	OSC3
	周波数	3.2 MHz
DSADC16	割込みレベル	3
	クロックソース	OSC3
	分周比	1:4
LCD4B	割込みレベル	0
	クロックソース	OSC1
	分周比	1:1
P07	割込みレベル	0
P07	Input / Output	Output
	チャタリング防止フィルタ	無効
	プルアップ / ダウン抵抗	無効
P40, P41, P42, P43, P44, P45, P46	Input / Output	Input
	チャタリング防止フィルタ	無効
	プルアップ / ダウン抵抗	プルアップ有効
SNDA	割込みレベル	0
	クロックソース	OSC1
	分周比	1:1
SPI	割込みレベル	0
	マスター / スレーブ	スレーブ
	入力端子プルアップ/ダウン	無効
	クロック位相/極性	mode 3
	データ順序	MSB 先頭
	データ長	8 bit
SVD4	割込みレベル	3
T16 Ch.0	クロックソース	OSC1
	分周比	1:256
	モード	リピートモード
	リロード値	127
T16 Ch.1	クロックソース	OSC3
	分周比	1:128
	モード	リピートモード
	リロード値	1249
T16 Ch.2	クロックソース	OSC1
	分周比	1:256
	モード	リピートモード
	リロード値	127

表 5.1-2 各種周辺回路設定値一覧(2)

周辺回路	設定項目	設定値
T16 Ch.3	クロックソース	OSC3
	分周比	1:128
	モード	リピート
	リロード値	1249
T16B_DMM Ch.0	割込みレベル	0
	クロックソース	OSC1
	分周比	1:1
	モード	リピートモード
T16B_DMM Ch.1	割込みレベル	6
	クロックソース	EXCL0
	分周比	1:1
	モード	リピートモード
T16B_DMM Ch.2	割込みレベル	0
	クロックソース	OSC1
	分周比	1:1
	モード	リピートモード
UART	割込みレベル	0
	クロックソース	OSC3
	分周比	1:1
	パリティ	None
	データ長	8 bit
	ボーレート	230400
	フロー制御	None

5.2 タイマ制御インターフェース

本ソフトウェアでは、随所でタイマを使用して時間を生成するため、T16 Ch.0、T16 Ch.1、T16 Ch.2、T16 Ch.3を用いて、『モード名表示タイマ』、『タイムアウトタイマ』、『スイッチポーリングスキャンタイマ』、『LCD 表示切り替えタイマ』、『オートパワーオフ切り替えタイマ』、『長押し判定タイマ』の6つのタイマ動作を実現します。

表 5.2-1 タイマ動作一覧

名称	使用する周辺回路機能	周期
モード名表示タイマ	T16 Ch.0	1s
タイムアウトタイマ	T16 Ch.0	3s
スイッチポーリングスキャンタイマ	T16 Ch.1	50 ms
LCD 表示切り替えタイマ	T16 Ch.1	500 ms
オートパワーオフ切り替えタイマ	T16 Ch.2	60 s *ソフトウェアで任意の時間に設定可能
長押し判定タイマ	T16 Ch.3	1 s

5.3 測定モード

ここでは、DMM コントローラを動作させ、測定処理を行います。本モードにおける動作は、DMM コントローラに設定中の測定モードにより異なり、種別は下表に従います。本モードの動作フローは下記の通りです。

表 5.3-1 測定モード動作分類

測定モード	動作分類
直流電圧測定	(1)
交流電圧測定	(1)
直流電流測定	(1)
交流電流測定	(1)
抵抗値測定(CC 方式)	(1)
抵抗値測定(CV 方式)	(1)
導通チェック	(1)
容量測定(CC 方式)	(2)
容量測定(CV 方式)	(2)
ダイオード VF 測定	(1)
交流電圧周波数測定	(2)
交流電流周波数測定	(2)
内部温度測定	(1)

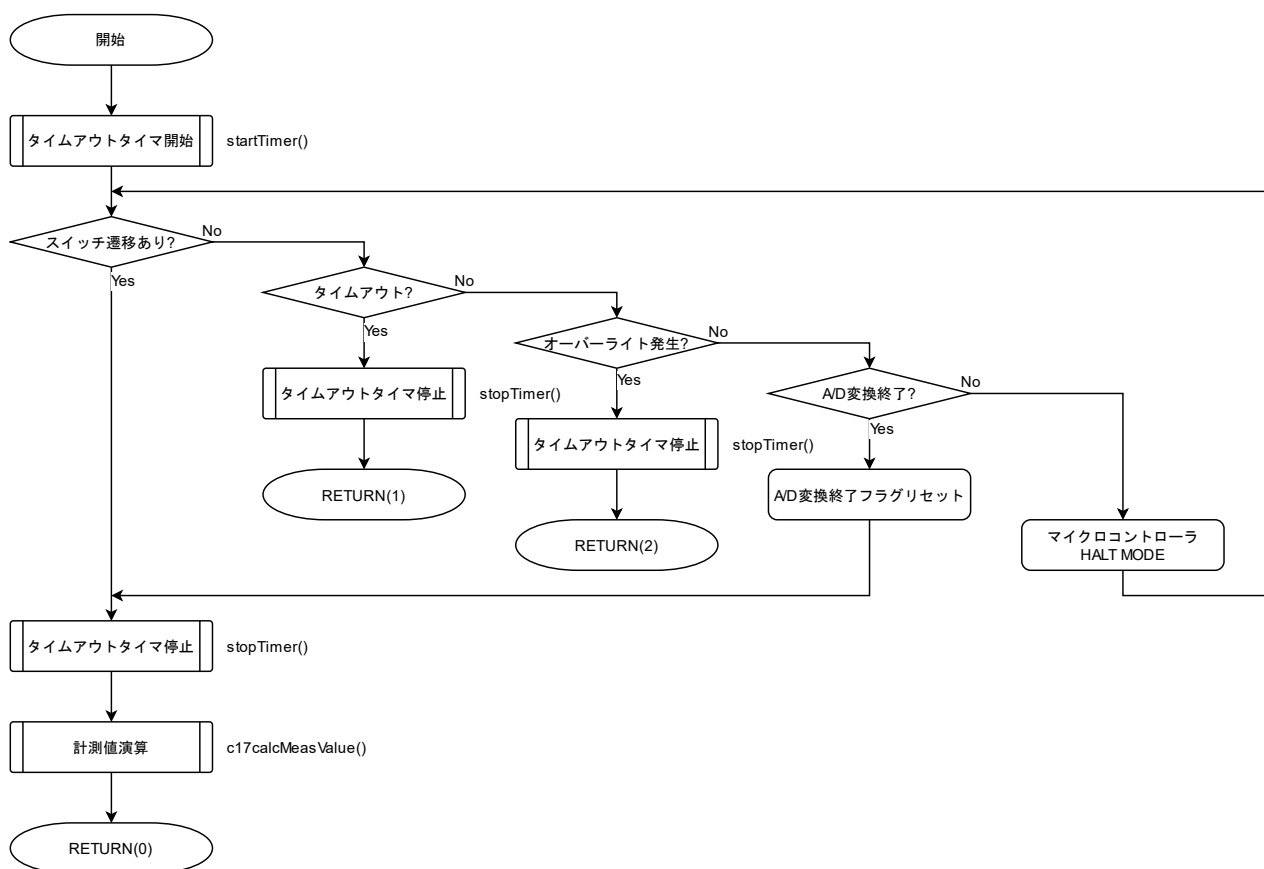


図 5.3-1 測定モード動作(1)

動作が(1)に分類される測定モード動作では、主に DMM コントローラの A/D 変換完了を待機する処理を行います。通常はマイクロコントローラの HALT モードで待機し、A/D 変換完了の割り込みが入ると動作を復帰します。また、3s のタイムアウト期間を設けており、A/D 変換動作が完了しない場合やオーバーライトエラーが発生した場合はその時点で動作を終了します。

また、スイッチ遷移による設定変更を検知すると A/D 変換待ち動作を中断します。ただし、測定モードによって通信への対応や測定レンジ切り替え・ピークホールド切り替えへの対応が異なるため、それぞれの測定モード毎に、監視対象とするスイッチが異なります。

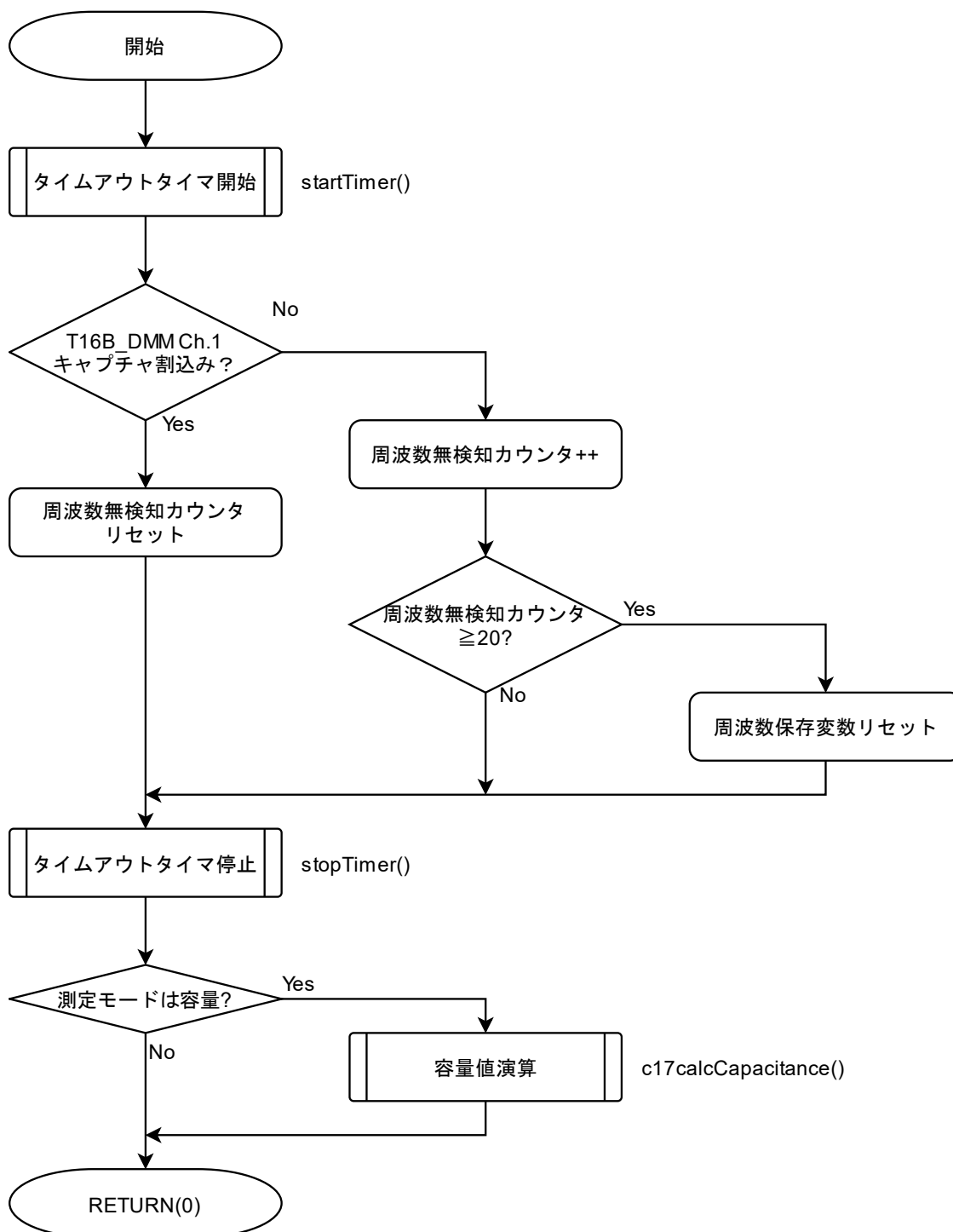


図 5.3-2 測定モード動作(2)

動作が(2)に分類される測定モード動作では、主に周波数の測定処理に関する動作を行います。周波数自体の演算は測定モードではなく、T16B_DMM Ch.1 割込みハンドラ上で行われますので、ここでは演算した周波数のリセット動作のみ行います。T16B_DMM Ch.1 キャプチャ割込みは、測定信号の入力に連動します。従って、割込みが発生しない場合というのは、測定対象とする入力信号が無い場合となります。この入力信号が無い状態が 20 回連続すると、これまでに保存している演算周波数をリセットして 0 にします。最後に、測定モードが容量であった場合に容量値を演算します。

5.3.1 測定モード切り替え

ここでは、初回起動時および測定モードまたは通信モード中にロータリースイッチの遷移を検知すると、測定中の場合は測定を中断し、ロータリースイッチの値に対応した測定モードに DMM コントローラを設定します。

5.3.2 測定レンジ切り替え

測定モード中にタクトスイッチ(SW4)が押されると、実行中の測定モードが測定レンジの切り替えに対応する場合に、測定レンジの切り替え処理を行います。測定モードが容量(CC 方式)の場合、1000uF レンジのみ T16B_DMM Ch.0 コンパレータカウンタ値が異なるため、ここで設定の変更を行います。本ソフトウェアにおける測定レンジ切り替え処理フローは以下の通りです。

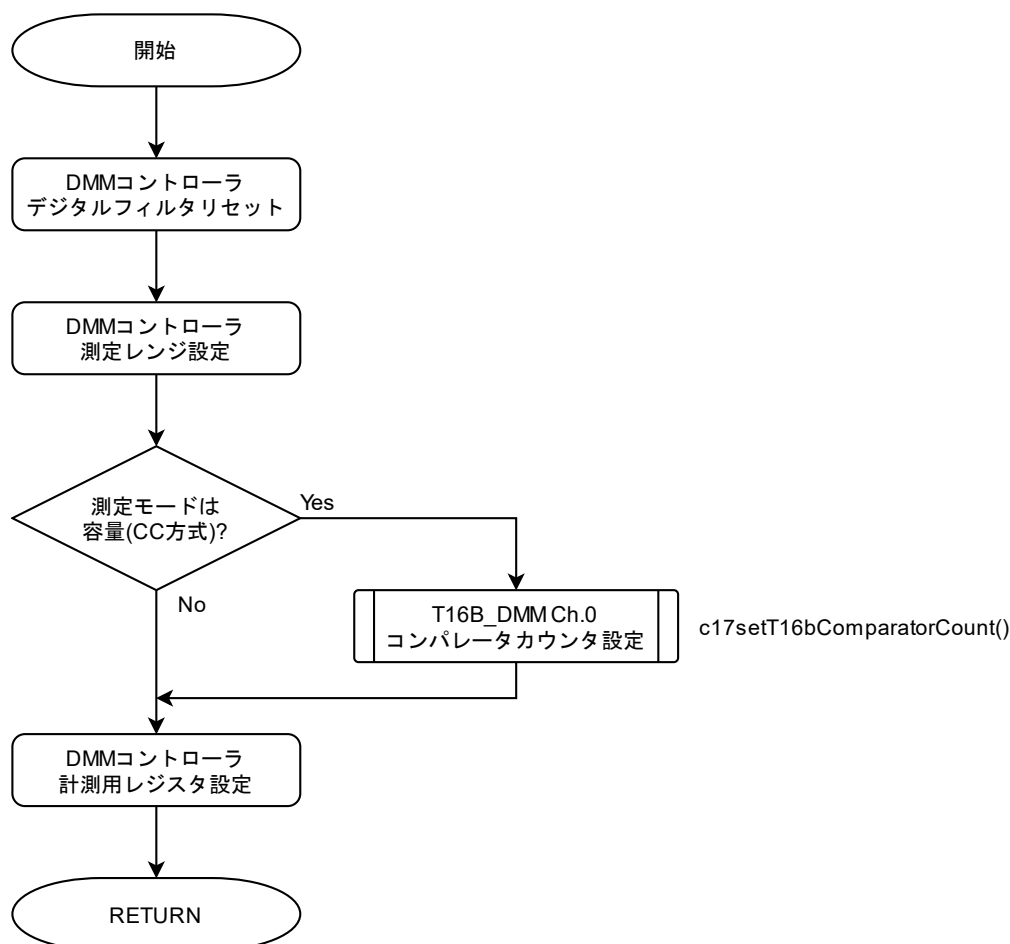


図 5.3.2-1 測定レンジ切り替え処理

5.3.3 ピークホールド切り替え

測定モード中にタクトスイッチ(SW3)が押されると、実行中の測定モードがピークホールド機能に対応している場合に、ピークホールドの切り替え処理を行います。ここでは、DMM コントローラのピークホールド設定に加えて、後述するピークホールドのソフトウェア処理での実現時に必要な変数の初期化も行います。

5.3.4 オートレンジ切り替え

測定モード中にタクトスイッチ(SW4)が長押しされると、実行中の測定モードがオートレンジ機能に対応している場合に、オートレンジの切り替え処理(OFF→ON)を行います。また、オートレンジ機能が ON の際に、タクトスイッチ(SW4)が押されると、実行中の測定モードがオートレンジ機能に対応している場合に、オートレンジの切り替え処理(ON→OFF)を行います。

5.3.5 データホールド切り替え

測定モード中にタクトスイッチ(SW3)を長押ししながら、タクトスイッチ(SW2)を押すと、データホールドの切り替え処理(OFF→ON)を行います。また、データホールド機能が ON の際に、タクトスイッチ(SW2)が押されると、データホールドの切り替え処理(ON→OFF)を行います。

5.3.6 リラティブ切り替え

データホールド機能が ON の際に、タクトスイッチ(SW3)を長押ししながら、タクトスイッチ(SW4)を押すと、リラティブの切り替え処理(OFF→ON)を行います。また、リラティブ機能が ON の際に、タクトスイッチ(SW4)が押されると、リラティブの切り替え処理(ON→OFF)を行います。

5.4 通信モード

測定モード中にタクトスイッチ(SW2)が押されると、実行中の測定モードが通信によるデータ送信に対応している場合に、通信モードの処理に入ります。通信モードに入ると、次の処理フローに従って処理を行います。フロー中の通信状態の更新処理では、次の状態遷移図に従って通信状態を更新します。同時に、ここで通信モード中にタクトスイッチ(SW2)を押すことによる通信終了の処理も行います。

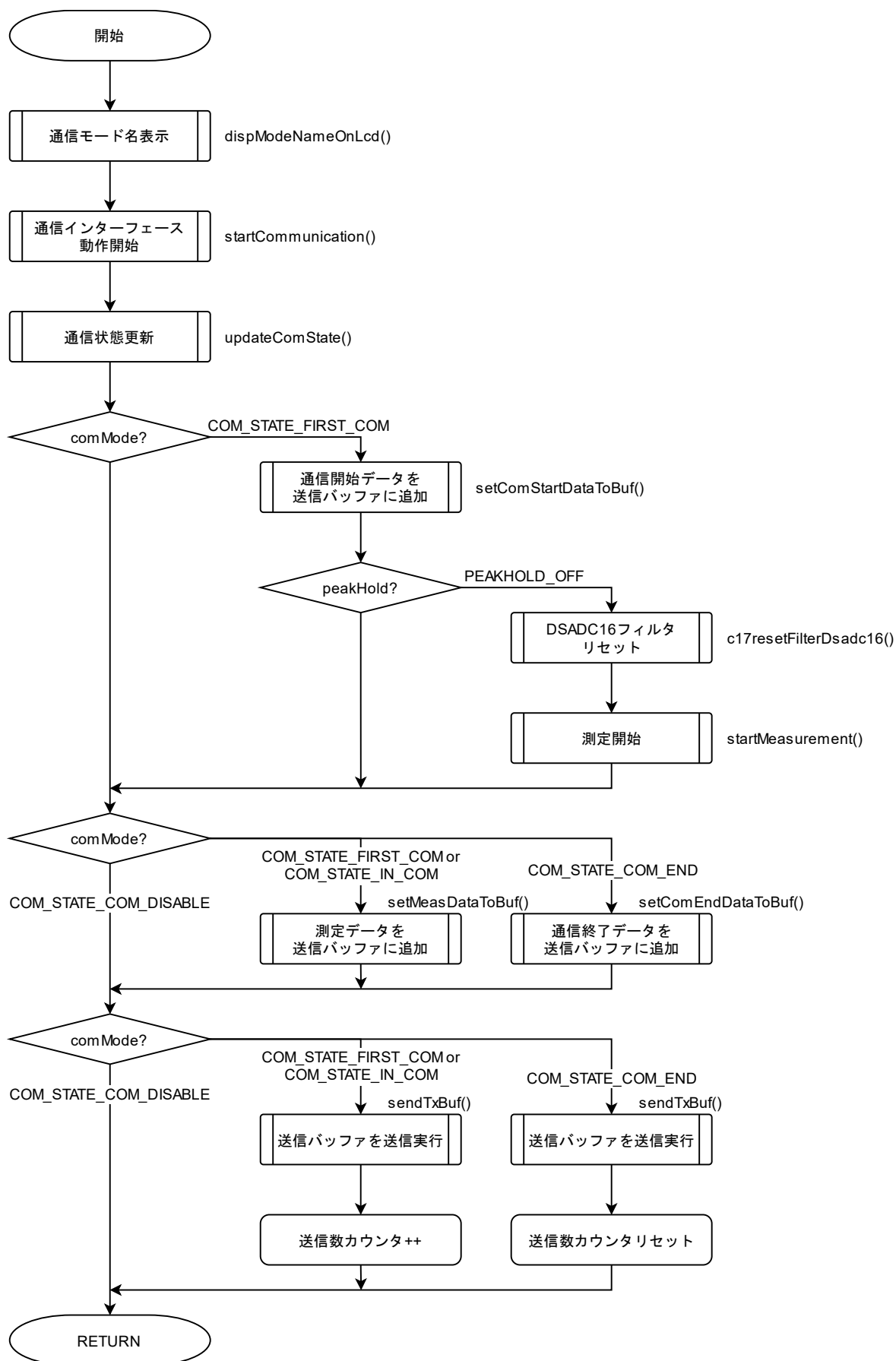


図 5.4-1 通信モード動作

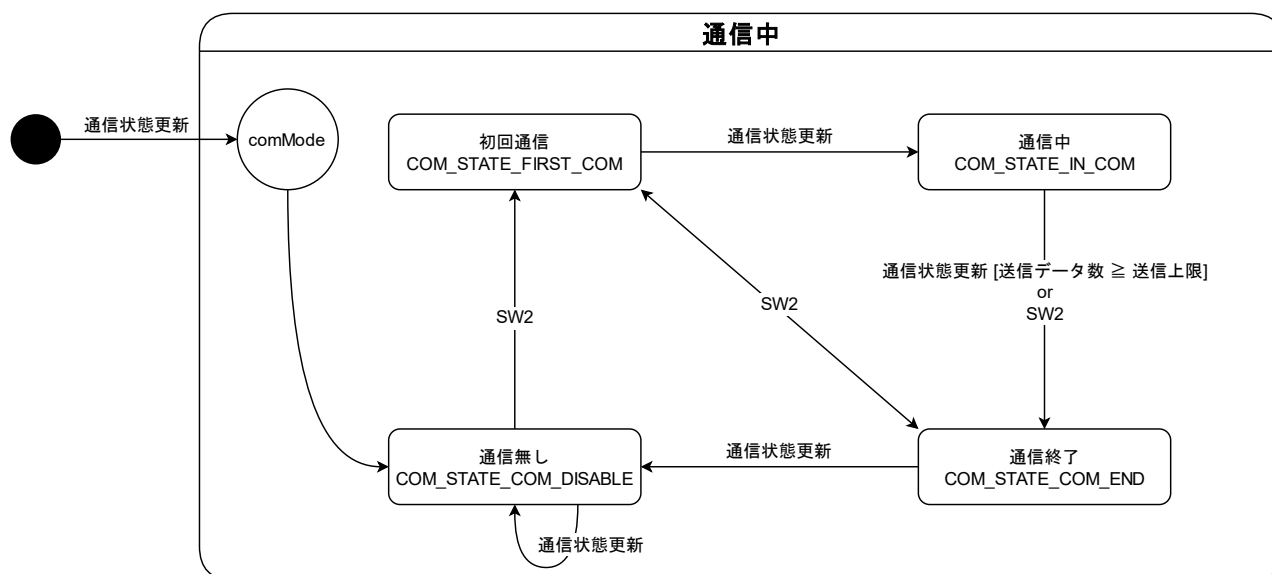


図 5.4-2 通信モード切り替えの遷移

5.5 計測データの演算機能

SVT17M03 に搭載されている S1C17M03 では、DMM コントローラや DMM 用 16 ビット PWM タイマを適切に設定することで、様々な回路パラメータの計測をすることができます。本ソフトウェアでは、回路から出力される A/D 変換値やタイマのカウント値を基に、電圧[V]や周波数[Hz]、容量[F]といった測定単位への変換演算を行います。

DMM コントローラにより計測可能な測定モードでは、下記 3 つの方法で測定単位の変換処理を行います。なお、本ソフトウェアでは導通チェックモード時に MCU のブザー出力制御と DMM コントローラの連携機能を使用するため、特別に演算は行っていません。また、内部温度測定においても本ソフトウェアでは測定温度の導出は行わない仕様のため、演算は行っていません。以下にそれぞれの変換方法について説明を行います。ただし、『(3)周波数を用いた演算』については本ソフトウェアとして特別な処理は行っていないため、説明を省略します。

表 5.5-1 測定値演算の方法分類

測定モード	変換方法
直流電圧測定	(1) A/D 変換値からの演算 / (4) キャリブレーションによる補正
交流電圧測定	(1) A/D 変換値からの演算 / (4) キャリブレーションによる補正
直流電流測定	(1) A/D 変換値からの演算 / (4) キャリブレーションによる補正
交流電流測定	(1) A/D 変換値からの演算 / (4) キャリブレーションによる補正
抵抗値測定(CC 方式)	(1) A/D 変換値からの演算 / (4) キャリブレーションによる補正
抵抗値測定(CV 方式)	(1) A/D 変換値からの演算 / (4) キャリブレーションによる補正
導通チェック	なし
容量測定(CC 方式)	(3)周波数を用いた演算 / (4) キャリブレーションによる補正
容量測定(CV 方式)	(3)周波数を用いた演算 / (4) キャリブレーションによる補正
ダイオード VF 測定	(1) A/D 変換値からの演算
交流電圧周波数測定	(2)T16B_DMM PWM タイマを用いた演算
交流電流周波数測定	(2)T16B_DMM PWM タイマを用いた演算
内部温度測定	なし

5.5.1 A/D 変換値からの演算

本方式では、DMM コントローラにより取得した A/D 変換値へそれぞれの測定モードや測定レンジに固有の数値を掛け合わせることで測定単位への変換が可能です。(キャリブレーション機能未使用時) 演算は次ページに示す変換係数 K を用いて次の式で表されます。

$$\text{測定値[任意の単位]} = A/D \text{ 変換値} \times K \quad (\text{式 5.5.1.1})$$

表 5.5.1-1 測定値の演算係数

測定モード	測定レンジ	変換係数 K
直流電圧測定	600mV	9.1552734e-5
	6V	9.1635214e-4
	60V	9.1561799e-3
	600V	9.1644287e-2
	1000V	8.4518667e-2
交流電圧測定	600mV	9.1552734e-5
	6V	9.1635214e-4
	60V	9.1561799e-3
	600V	9.1644287e-2
	1000V	8.4518667e-2
直流電流測定	600uA	8.3665198e-8
	6mA	9.0637298e-7
	60mA	8.3673482e-6
	600mA	9.0646272e-5
	6A	8.4510216e-4
	10A	9.1552734e-3
交流電流測定	600uA	8.3665198e-8
	6mA	9.0637298e-7
	60mA	8.3673482e-6
	600mA	9.0646272e-5
	6A	8.4510216e-4
	10A	9.1552734e-3
抵抗値測定(CC 方式)	600Ω	5.493164e-2
	6kΩ	4.577637e-1
	60kΩ	4.577637
	600kΩ	4.5771835e1
	6MΩ	4.5735164e2
	60MΩ	4.5776367e3
抵抗値測定(CV 方式)	600Ω	5.493164e-2
	6kΩ	4.577637e-1
	60kΩ	4.577637
ダイオード VF 測定	-	2.2888184e-4

5.5.2 ピークホールドソフトウェア処理

本方式では、一部のレジスタ設定において DMM コントローラのピークホールド機能を代替するため、ソフトウェアによる処理を行っています。ピークホールド機能をソフトウェア処理する場合のレジスタ設定の条件と処理のフローは以下の通りです。処理後、計測値の演算にはフロー中で保存した『ひとつ前の測定結果レジスタ値』を用いることでピークホールドの機能を実現します。

表 5.5.2-1 ピークホールド機能をソフトウェア処理する条件

DSADC16CTL.FUNCSEL	DSADC16CONF.TRUERMS_ON
1 (交流電圧測定)	1
3 (交流電流測定)	1

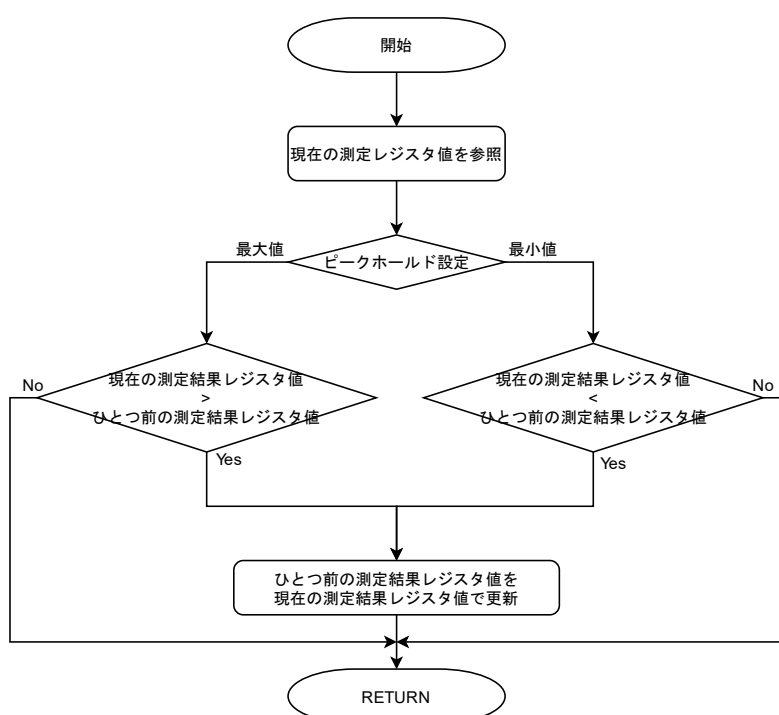


図 5.5.2-1 ピークホールド機能のソフトウェア処理

5.5.3 T16B_DMM PWM タイマを用いた演算

周波数は、DMM コントローラの WINCMP から出力される矩形波(周波数測定対象の入力パルス)を、DMM 用 16 ビット PWM タイマ 3 チャンネルを利用して測定し、周波数に相当する値に変換します。周波数測定に用いる PWM タイマの役割や設定の詳細については、『S1C17M02/M03 テクニカルマニュアル』を参照してください。

テクニカルマニュアル記載の通りに T16B_DMM 設定を行うと、下記に示す構成で周波数を測定することができます。本ソフトウェアでは、周波数の演算を T16B_DMM Ch.1 の CMPCAP0 割込みハンドラ内で処理することで、即応性の高い周波数演算を可能にしています。

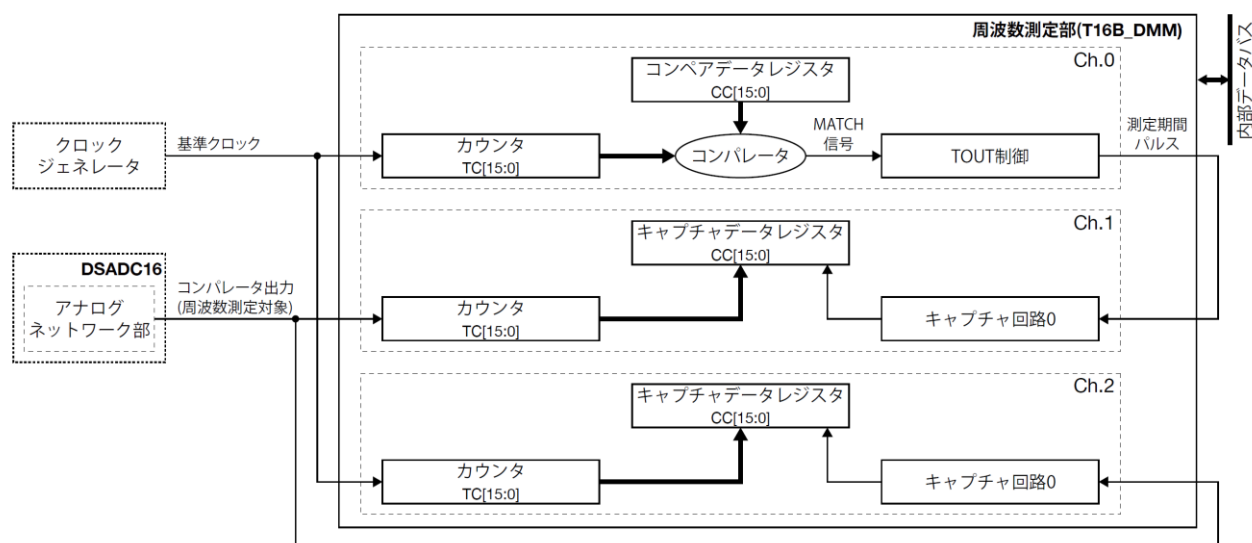


図 5.5.3-1 周波数測定回路ブロック図(テクニカルマニュアルより抜粋)

5.5.3.1 周波数計算例

周波数計算の演算については、テクニカルマニュアル記載の方法で行います。ここでは 5Hz の入力信号を測定する場合を例に挙げて説明を行います。

5Hz の矩形波信号が入力されたとき、周波数測定回路ブロックでは下記の通りにタイマの各チャンネルでカウントがキャプチャされ、次の手順で周波数を計算します。ここで、各カウントはオーバーフローするとカウント値が 0 になるため、本ソフトウェアでは Ch.1、Ch.2 において CNTMAX 割込みを利用し、オーバーフローした回数をカウントしておくことで、計測期間中における累積のカウントキャプチャ値を計算しています。

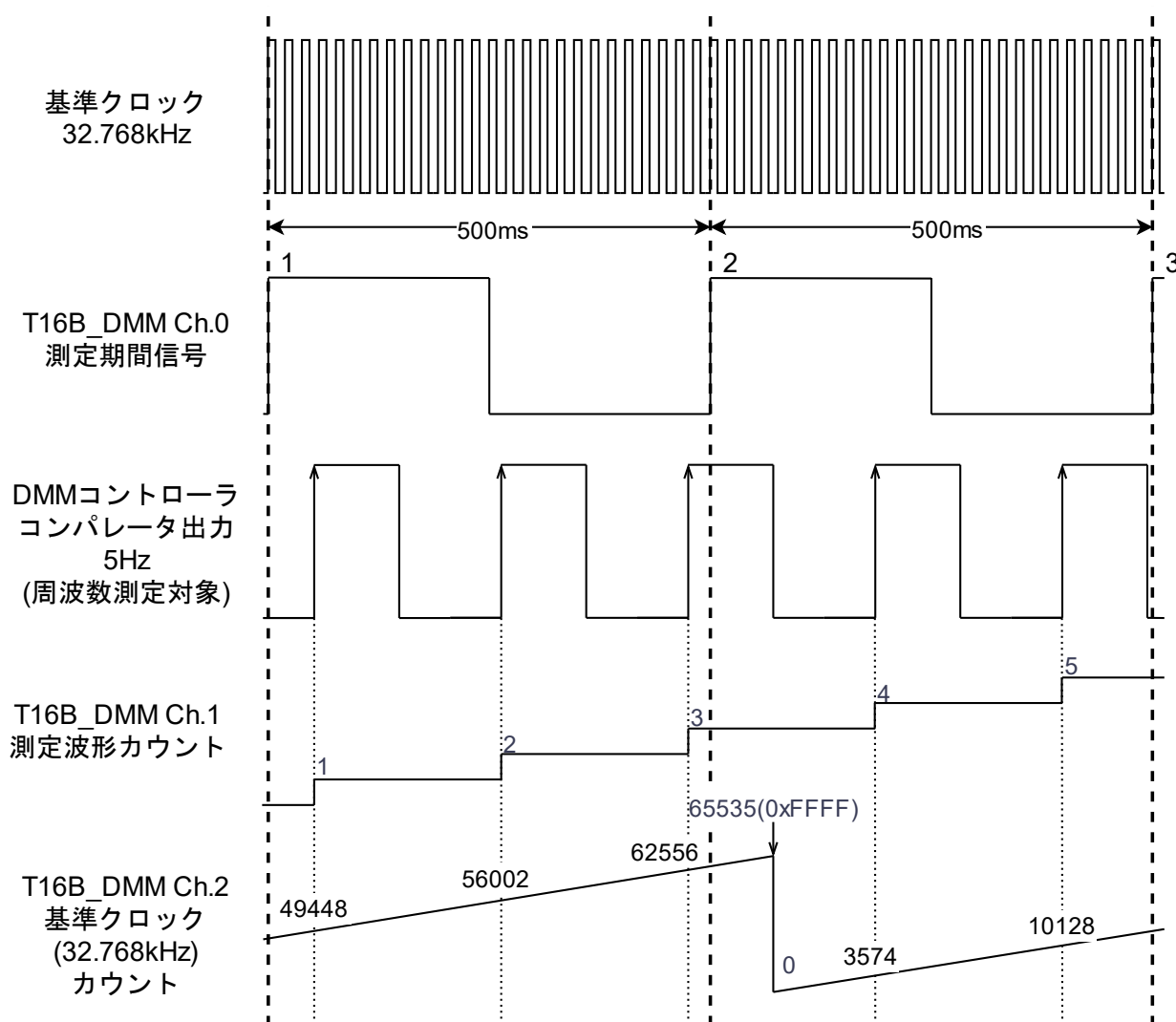


図 5.5.3.1-1 T16B_DMM 各 Ch キャプチャ値

- 周波数計算式

T16B_DMM Ch.1 1 回目キャプチャ値： n_{IN1}

T16B_DMM Ch.1 2 回目キャプチャ値： n_{IN2}

T16B_DMM Ch.2 1 回目キャプチャ値： n_{E1}

T16B_DMM Ch.2 2 回目キャプチャ値： n_{E2}

f_R ：基準クロック周波数(32,768Hz)

n_R ：計測期間($2 \times$ T16B0CCR0 レジスタ設定値)

$n_{IN1} - n_{IN2} \geq 4000$ のとき

$$F[\text{Hz}] = \frac{f_R \times (n_{IN2} - n_{IN1})}{n_R} \quad (\text{テクニカルマニュアル式 14.1})$$

$n_{IN1} - n_{IN2} < 4000$ のとき

$$F[\text{Hz}] = \frac{f_R \times (n_{IN2} - n_{IN1})}{n_{E2} - n_{E1}} \quad (\text{テクニカルマニュアル式 14.2})$$

1. T16B_DMM Ch.0の測定期間信号の2回目立ち上がり時、つまりT16B_DMM Ch.1 CMPCAP0割込み発生時、T16B_DMM Ch.1測定波形カウントキャプチャ値($n_{IN1} = 3$)とT16B_DMM Ch.2の基準クロックカウントキャプチャ値($n_{E1} = 62556$)を保存します。
2. 次のT16B_DMM Ch.0の測定期間信号の立ち上がり(3回目)時、同様にT16B_DMM Ch.1測定波形カウントキャプチャ値($n_{IN1} = 5$)とT16B_DMM Ch.2の基準クロックカウントキャプチャ値 n_{E2} を保存します。ここで、T16B_DMM Ch.2のキャプチャ値はカウントオーバーフローを1回経ているため、 $n_{E2} = 10128 + 65536 \times 1 = 75664$ となります。
3. n_{IN} の差分を計算します。
 $n_{IN2} - n_{IN1} = 5 - 3 = 2$
4. $n_{IN2} - n_{IN1} = 2 < 4000$ であるため、テクニカルマニュアル式14.2に従い周波数を計算します。

$$F[\text{Hz}] = \frac{f_R \times (n_{IN2} - n_{IN1})}{n_{E2} - n_{E1}} = \frac{32768 \times (5 - 3)}{75664 - 62556} \approx 4.999695$$

5.5.4 キャリブレーションによる補正

本ソフトウェアでは、線形補間によるキャリブレーションを行うことによって、得られた計測値の補正を行っています。また、本ソフトウェアでは、キャリブレーション機能は常に ON となっており、LCD に表示される計測値は全て補正後の値となっています。

以下に、キャリブレーションの対象となる測定モードおよび測定レンジを示します。

表 5.5.4-1 キャリブレーションの対象測定モード・測定レンジ

対象測定モード	対象測定レンジ
直流電圧	全レンジ
交流電圧	全レンジ
直流電流	全レンジ
交流電流	全レンジ
抵抗値(CC 方式)	全レンジ
抵抗値(CV 方式)	全レンジ
容量値(CC 方式)	全レンジ
容量値(CV 方式)	全レンジ

本ソフトウェアでは、キャリブレーションを行うために、事前に各測定モードおよびレンジ毎にキャリブレーション用の基準点を 2 点(基準点_High および基準点_Low)測定する必要があります。以下に、キャリブレーションを行うための事前準備手順を示します。

用意するもの：

- SVT17M03
- PC
- 校正器
- DMM_Calibration_Sheet

キャリブレーション用の基準値測定

1. PC、SVT17M03、および校正器を接続します。(下図参照)



図 5.5.4-1 キャリブレーション用基準値測定時の接続図

2. 校正器から基準となる任意の値を出力させ、その値をSVT17M03より計測します。各測定モードおよび測定レンジにつき、2点(基準点_Highおよび基準点_Low)の計測を行います。
3. PCにてDmmEvalTool.exeを実行し、SVT17M03より計測した基準点のA/D変換値を取得します。(容量測定の場合は、A/D変換値の代わりに容量値を取得します)
4. 校正器からの入力値および得られたA/D変換値をDMM_Calibration_Sheet内の"table"シートに入力します。(容量値測定の場合は、校正器からの入力値およびDmmEvalToolより得られた容量値を入力します)

1	DC voltage				
2		Calibrator		SVT17M03	
3		Reference Input	Reference Value[V]	Reference ADC Value	Example Value
4					6640
5					-6704
6					6614
7					-6678
8					6630
9					-6696
10					6561
11					-6635
12					10323
13					-10563
14					

図 5.5.4-2 DMM_Calibration_Sheet 内の"table"シート

5. リファレンスソフトウェアの c17dsadc.h の該当箇所(159行目～447行目)の define 文を DMM_Calibration_Sheet の"header"シートにて生成された define 文に置き換えます。

1	/// Reference values for calibration		
2	///DCV		
3	#define CALIB_MEAS_REF_DCV_600MV_HIGH	0	///< Reference Value of DCV_+600mV
4	#define CALIB_MEAS_REF_DCV_600MV_LOW	0	///< Reference Value of DCV_-600mV
5	#define CALIB_MEAS_REF_DCV_6V_HIGH	0	///< Reference Value of DCV_+6V
6	#define CALIB_MEAS_REF_DCV_6V_LOW	0	///< Reference Value of DCV_-6V
7	#define CALIB_MEAS_REF_DCV_60V_HIGH	0	///< Reference Value of DCV_+60V
8	#define CALIB_MEAS_REF_DCV_60V_LOW	0	///< Reference Value of DCV_-60V
9	#define CALIB_MEAS_REF_DCV_600V_HIGH	0	///< Reference Value of ACV_+600V
10	#define CALIB_MEAS_REF_DCV_600V_LOW	0	///< Reference Value of DCV_-600V
11	#define CALIB_MEAS_REF_DCV_1000V_HIGH	0	///< Reference Value of ACV_+1000V
12	#define CALIB_MEAS_REF_DCV_1000V_LOW	0	///< Reference Value of DCV_-1000V
13			
14	///ACV		
15	#define CALIB_MEAS_REF_ACV_600MV_HIGH	0.424264069	///< Reference Value of ACV_600mV
16	#define CALIB_MEAS_REF_ACV_600MV_LOW	0	///< Reference Value of ACV_00hm(@600mV)
17	#define CALIB_MEAS_REF_ACV_6V_HIGH	4.242640687	///< Reference Value of ACV_6V
18	#define CALIB_MEAS_REF_ACV_6V_LOW	0	///< Reference Value of ACV_00hm(@6V)
19	#define CALIB_MEAS_REF_ACV_60V_HIGH	42.42640687	///< Reference Value of ACV_60V
20	#define CALIB_MEAS_REF_ACV_60V_LOW	0	///< Reference Value of ACV_00hm(@60V)

図 5.5.4-3 DMM_Calibration_Sheet 内の"header"シート

6. GNU17を用いて再ビルドを実行後、SVT17M03にソフトウェアを書き込みます。

線形補間によるキャリブレーションを行う際は、以下の式を用います。

(例) DCV 6V レンジ

基準点_High の(計測値, A/D 変換値) : (6.0V, 6640)

基準点_Low の(計測値, A/D 変換値) : (-6.0V, -6704)

$$\begin{aligned}
 & \text{補正電圧値} = \text{基準点_Low での計測値} + (\text{基準点_High での計測値} - \text{基準点_Low での計測値}) \\
 & \quad * (\text{実測A/D変換値} - \text{基準点_Low でのA/D変換値}) / (\text{基準点_High でのA/D変換値} \\
 & \quad - \text{基準点_Low でのA/D変換値}) \\
 & = -6.0 + \{6640 - (-6704)\} * \{\text{実測A/D変換値} - (-6704)\} / \{6 - (-6)\}
 \end{aligned}$$

Appendix A S1C17M03 から S1C17M02 への変更方法

本ソフトウェアパッケージの中におけるアプリケーション・ミドルソフトウェアは、S1C17M03 が搭載された評価ボードである SVT17M03 を動作させることを目的としたソフトウェアです。そのため、S1C17M02 で動作させることを目的とした場合、該当のソフトウェアはそのまま利用することはできません。そこで、本章ではお客様の環境にて、本ソフトウェアパッケージのうち計測に関わる部分を流用する場合の注意点を示します。

流用する対象のソフトウェアとして以下のプログラムファイルを想定しています。必要に応じて編集を行ってください。

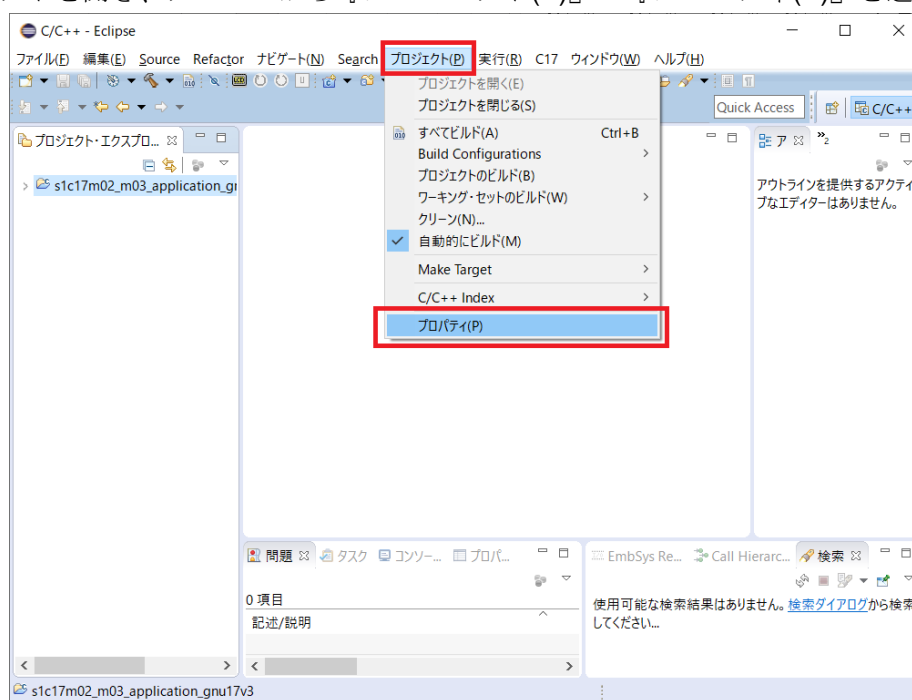
表 0-1 編集が必要なプログラムファイル一覧

プログラムファイル名	概要
inc/c17_mcu_select.h	MCU 選択用ヘッダーファイル
main_measurement.c	測定モードプログラムファイル

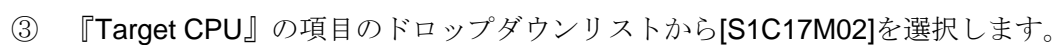
開発環境(GNU17v3 IDE)の設定

開発環境(GNU17v3 IDE)で S1C17M02 を開発するにあたり、プロジェクトの設定が必要です。以下の手順で設定を行ってください。

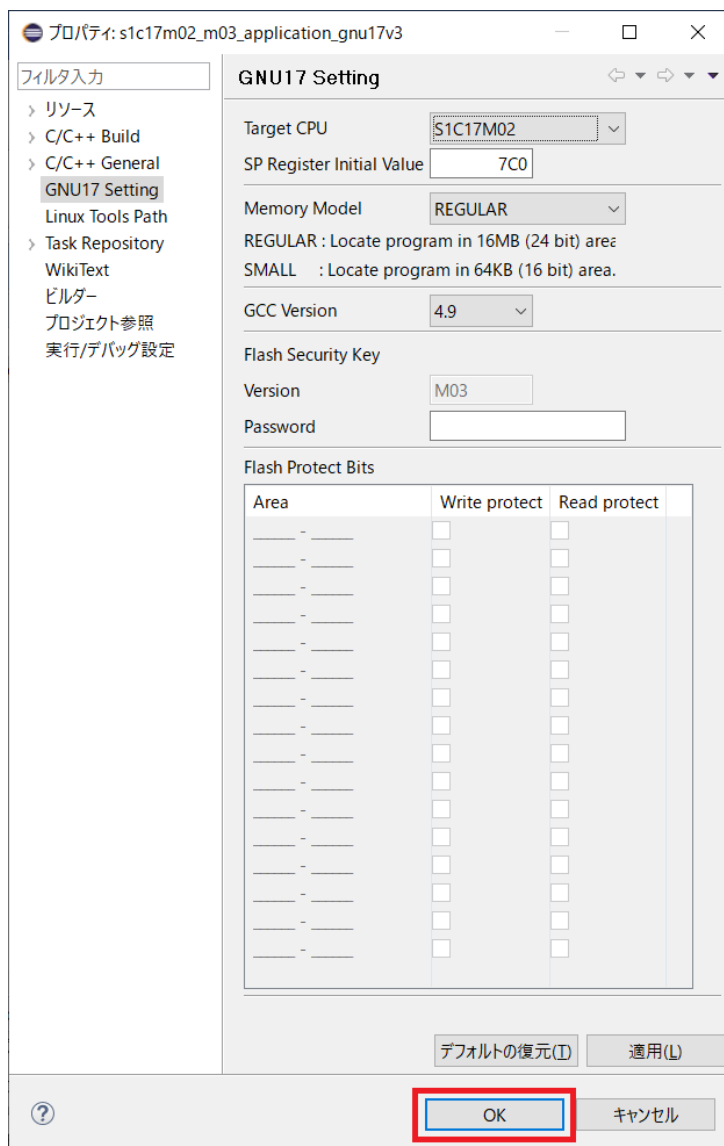
- ① プロジェクトを開き、メニューから『プロジェクト(P)』→『プロパティ(P)』を選択します。



② 『GNU17 Setting』を選択します。



- ④ 『OK』 ボタンをクリックします。



c17_mcu_select.h

本ソフトウェアパッケージは前述の通り、SVT17M03 の動作に特化しています。従って初期設定では S1C17M03 上で動作するように設定されています。S1C17M02 で使用したい場合は『4.3.4 使用する MCU の設定』で述べたように、inc/c17_mcu_select.h 内のマクロ定義 C17_MCUSEL_M03 を無効化し、C17_MCUSEL_M02 を有効化してください。

main_measurement.c

本ファイルは、各種ドライバプログラムを用いて DMM 動作を実現するためのアプリケーションプログラムファイルです。本プログラム内の各種関数を利用することで、簡単に DMM 動作を実現することができます。

本プログラムでは利用するプログラムについて仕様上の都合から、お客様の環境へ流用する際に考慮すべき箇所があります。以下に該当の箇所を示します。

inc/c17_mcu_select.h で定める C17_MCUSEL_M02 を有効化すると『表 A-1 指定が必要なプログラムファイル一覧』のうち、main_measurement.c 内の関数がはじめてから利用できなくなっています。これは、S1C17M02 と S1C17M03 のハードウェア仕様の差によるエラーを回避するために施した措置です。そのため、当該ファイル内の関数を使用するためにいくつかの箇所を編集する必要があります。

関数 : runMeasMode()

本関数は、DSADC16 や T16B_DMM による計測を開始した後、計測終了から測定値を演算するまでを行う関数です（具体的な動作については図 5.3-1 測定モード動作(1)を参照ください）。本関数の動作の中で、以下の図で示す部分にハードウェア構成に依存する箇所があります。

```
switch(m meas.mode)
{
    case MEASMODE_DCV:
    case MEASMODE_ACV:
    case MEASMODE_DCI:
    case MEASMODE_ACI:
    case MEASMODE_OHMC:
    case MEASMODE_OHMCV:
    case MEASMODE_DIODE:
    case MEASMODE_TEMP:
        // Wait until the AD conversion is complete.
        while(1) {
            // Polling monitoring of switch inputs
            if(swPollingFlg == 1) {
                updateSwitchState();
                swPollingFlg = 0;
            }
            // Break if there is a switch input transition
            if( (transition.swMeasModeFlg == 1) || (transition.swComModeFlg == 1) || (transition.swMeasRangeFlg == 1) ) {
                break;
            }

            if(timeOutFlg == 1) { // If timeout, exit this function
                stopTimer(TIMER_TIMEOUT);
                return EXIT_TIMEOUT;
            }

            if(c17dsadcl6OverWriteFlg == 1) { // Exit this function if Over write error has occurred.
                stopTimer(TIMER_TIMEOUT);
                return EXIT_OVER_WRITE;
            }

            if(c17dsadcl6AdcEndFlg == 1) { // Break if the AD conversion is complete
                c17dsadcl6AdcEndFlg = 0;
                break;
            }
            asm("halt");
        }
        stopTimer(TIMER_TIMEOUT);

        result.f = c17calcMeasValue(m meas.mode, m meas.range, m meas.peakHold);
        break;
}
```

図 0-1 runMeasMode 関数におけるハードウェア依存部分

①部分

図 A-1 中の①で示す部分の処理は、SVT17M03 に実装されているタクトスイッチやロータリースwitch に結びつく GPIO の入力状態をポーリングによって更新している箇所になります。本ソフトウェアパッケージでは、入出力ポート(PPORT)のうち P40～P46 を使用していますので、お客様の環境によって編集（または削除）する必要があります。詳細は mid_switch.c 内の関数 getTactileSwitchInput および関数 getRotarySwitchInput を確認してください。

②部分

図 A-1 中の②で示す部分の処理は、タクトスイッチやロータリースwitchに入力操作が行われた場合にループ処理を抜ける箇所になります。入力操作有無の識別には `mid_switch.c` で定義される `transition` 構造体変数を用いていますので、お客様の環境によって編集（または削除）する必要があります。詳細は同プログラム内の関数 `updateSwitchState` を確認してください。

改訂履歴表

[illegible]

セイコーエプソン株式会社

営業本部 MD営業部

東京 〒191-8501 東京都日野市日野 421-8

大阪 〒530-6122 大阪市北区中之島 3-3-23 中之島ダイビル 22F

ドキュメントコード : 414282201

2022 年 3 月 作成

2025 年 7 月 改訂