

EV-200-USB-3 (RL78/I1B 評価キット)	2016/08/01	SJAA-150010-01	1/54
アプリケーションノート 基本編			

EV-200-USB-3 (RL78/I1B 評価キット) アプリケーションノート 基本編 第 1 版

※ 本アプリケーションノートに記載の全ての情報は発行時点のものであり、予告なしに仕様を変更することがあります。

目次

1.	はじめに.....	3
2.	安全にお使いいただくために.....	4
3.	ユーザプログラムの組み込み.....	8
3.1.	サンプルプログラムの無効化.....	10
3.2.	LCD 切替スイッチの使用方法.....	11
3.3.	LCD の使用方法.....	13
3.4.	LED の使用方法.....	15
3.5.	EEPROM の使用方法.....	17
3.6.	WDT 処理.....	20
3.7.	電力測定結果の参照.....	21
4.	拡張シリアル I/F.....	26
4.1.	I/F 仕様.....	26
4.2.	ブロック図.....	27
4.3.	電気的特性.....	28
5.	PMOD モジュール I/F.....	30
5.1.	I/F 仕様.....	30
5.2.	電気的特性.....	33
5.3.	モジュールの取り付け.....	35
6.	電池の交換.....	37
6.1.	概要.....	37
6.2.	電池の交換方法.....	38
7.	電力測定処理の速度改善.....	39
7.1.	外部クロックを使用する.....	39
7.2.	クロック補正処理を無効にする.....	41
7.3.	固定ゲインモードを使用する.....	42
8.	サンプリング値の変換係数.....	43
8.1.	$\Delta\Sigma$ ADC の特性.....	43
8.2.	電圧用係数.....	44
8.3.	電流 (CT) 用係数.....	45
8.4.	電流 (シャント) 用係数.....	46
8.5.	誤差補正.....	47
9.	定格電源の変更.....	48
9.1.	規格について.....	48
9.2.	ケーブルの交換と設定変更.....	49
9.3.	ソフトウェアの変更箇所.....	50
10.	付録.....	51
10.1.	各電力の計算式.....	51
11.	改版履歴.....	54

1. はじめに

本書では、EV-200-USB-3 (RL78/I1B 評価キット) でお客様の作成したプログラムを動作させるための基本的な方法について説明します。

以下、EV-200-USB-3 (RL78/I1B 評価キット) は「本製品」と記述します。

ルネサスエレクトロニクス株式会社 (<http://japan.renesas.com/>) 製マイコンの RL78/I1B に関しましては、マイコンのマニュアルをご参照下さい。

E1：ルネサスエレクトロニクス株式会社製 オンチップデバッグエミュレータです。尚、購入する際は本製品の購入元にお問い合わせ下さい。

CS+：ルネサスエレクトロニクス株式会社製 統合開発環境です。

また、関連文書として以下に示した資料を使用します。

関連文書	発行者
EV-200-USB-3 (RL78/I1B 評価キット) ユーザーズ・マニュアル ハードウェア編	(株)内藤電誠町田製作所
EV-200-USB-3 (RL78/I1B 評価キット) ユーザーズ・マニュアル ソフトウェア編	(株)内藤電誠町田製作所
EV-200-USB-3 (RL78/I1B 評価キット) ユーザーズ・マニュアル 拡張シリアル I/F コマンド仕様編	(株)内藤電誠町田製作所
EV-200-USB-3 (RL78/I1B 評価キット) ユーザーズ・マニュアル 電力測定用アプリケーション編	(株)内藤電誠町田製作所

2. 安全にお使いいただくために

本製品は、安全に十分配慮して設計されています。しかし、誤った使い方をすると、火災や感電などにより人身事故になることがあります危険です。事故を防ぐために次のことを必ずお守り下さい。

表記の意味

本製品を安全にお使いいただくための項目を次のように記載しています。

記載内容を守っていただけない場合、どの程度影響があるかを表しています。

 警告	人が死亡または重症を負うことが想定される内容を示します。
 感電	人が感電してしまうことが想定される内容を示します。
 注意	人が傷害を負うことが想定される内容、および、物的損害の発生が予想される内容を示します。

傷害や事故の発生を防止するための禁止事項は次のマークで表しています。

 禁止	してはいけないことの内容を示します。
---	--------------------

傷害や事故の発生を防止するための指示事項は次のマークで表しています。

 厳守	必ず行っていただきたい事項の内容を示します。
---	------------------------

 警告事項 **感電注意**

人が死亡または重症を負うことが想定される内容を示します。
また人が感電してしまうことが想定される内容を示します。

 禁止

- 分解・改造はしないで下さい。
感電・漏電・故障・火傷・火災・発熱の原因となります。
- 近くで、喫煙や飲食をしないで下さい。
感電・漏電・故障・火傷・火災・発熱の原因となります。
- 落としたり、強い衝撃を与えたりしないで下さい。
感電・漏電・故障・火傷・火災・発熱の原因となります。
- 次のような場所では使用、保管しないで下さい。
 - ・ 屋外など直射日光が当たる場所
 - ・ 湯気、塵、油煙などの多い場所
 - ・ 静電気や電磁気的なノイズが発生しやすい場所
 - ・ 振動するような場所
 - ・ 不安定な場所
 - ・ 腐食性ガスが発生するような場所
 - ・ 水がかかる場所
 - ・ 結露した状態感電・漏電・故障・火傷・火災・発熱の原因となります。
- 薬品の近くで使用や保管はしないで下さい。
溶けたり、変形したり、故障の原因となります。
- 過電圧での使用、保証温湿度範囲外での使用や保管はしないで下さい。
感電・漏電・故障・火傷・火災・発熱の原因となります。
- 火中へ投下したり、熱いものに近づけたり、加熱しないで下さい。
故障・火傷・火災・発熱の原因となります。
- 濡れた手で、本製品及び測定対象となる電気機器のプラグを抜き差ししないで下さい。
感電・漏電・故障の原因となります。
- 電源が切れていても、濡れた手で触らないで下さい。
感電・漏電・故障の原因となります。
- 通電中にケースの蓋を開けないで下さい。必ず電源コードのプラグを抜いてからケースの蓋を開けて下さい。
感電・故障の原因となります。
- 端子をショートさせないで下さい。
感電・故障の原因となります。
- 電源コードを束ねた状態では使用しないで下さい。
感電・故障・火傷・火災・発熱の原因となります。

	2016/08/01	SJAA-150010-01	6/54
EV-200-USB-3 (RL78/I1B 評価キット)	アプリケーションノート 基本編		

	- 電源コードの被覆が損傷していた場合は、使用を中止して下さい。 感電・漏電・故障・火傷・火災・発熱の原因となります。 - 本製品及び測定対象となる電気機器のプラグが変形しているものは使用しないで下さい。 感電・漏電・故障・火傷・火災・発熱の原因となります。 - Pmod モジュール I/F は、一次側電源と絶縁されておりません。電源が入っている場合は、感電にご注意下さい。 感電・故障の原因となります。
 厳守	- ご使用の前には、必ず入力電源電圧および電源コードの外観などに異常がないかご確認下さい。確認の際には、必ず電源コードのプラグを抜いて下さい。 感電・漏電・故障・火傷・火災・発熱の原因となります。 - 万一、煙や異臭、異常な音、異常な発熱などが発生した時は、すぐにコンセントから本製品の電源コードのプラグを抜いて下さい。 感電・故障・火傷・火災・発熱の原因となります。 - 電源を切った後も部品が発熱している恐れがあります。ケースの蓋を開ける場合は、十分に注意して下さい。 火傷の原因となります。 - 消費電流が 15A を越えないように接続して下さい。 感電・故障・火傷・火災・発熱の原因となります。

注意事項

人が傷害を負うことが想定される内容、および、物的損害の発生が予想される内容を示します。

 禁止	• 何らかの異常に気が付いた場合は直ちに使用を中止して下さい。 故障の原因となります。 • 本製品を踏んだり、本製品の上に物を置いたりしないで下さい。 故障の原因となります。 • 本製品（プリント基板）へのねじれ・たわみ・衝撃等のストレスは故障の原因になります。また取扱いの際には、静電気対策を行った上で、基板端を持ち直接部品に触れないよう注意して下さい。 怪我・故障の原因となります。 • 本製品及び電気機器のプラグを必要以上の力を加えて抜き差ししないで下さい。 故障の原因となります。 • 本製品を有機溶剤で拭いたり、可燃性ガスを含んだスプレーを吹き付けたりしないで下さい。 故障の原因となります。 • 本製品が洗剤や殺虫剤などの液体に浸った場合は、使用を中止して下さい。 絶縁不良、金属の腐食等が発生し大変危険です。
 厳守	• 本製品は評価キットです。 評価目的やデモンストレーション目的等で利用されることを想定しています。 • 本製品の汚れは乾いた布で軽く拭き取って下さい。 故障の原因となります。

3. ユーザプログラムの組み込み

本ソフトウェアはユーザプログラムを実装するための I/F 関数を備えています。
図 3-1 のユーザプログラム (サンプルプログラム) に該当する部分がユーザプログラム I/F 関数になります。

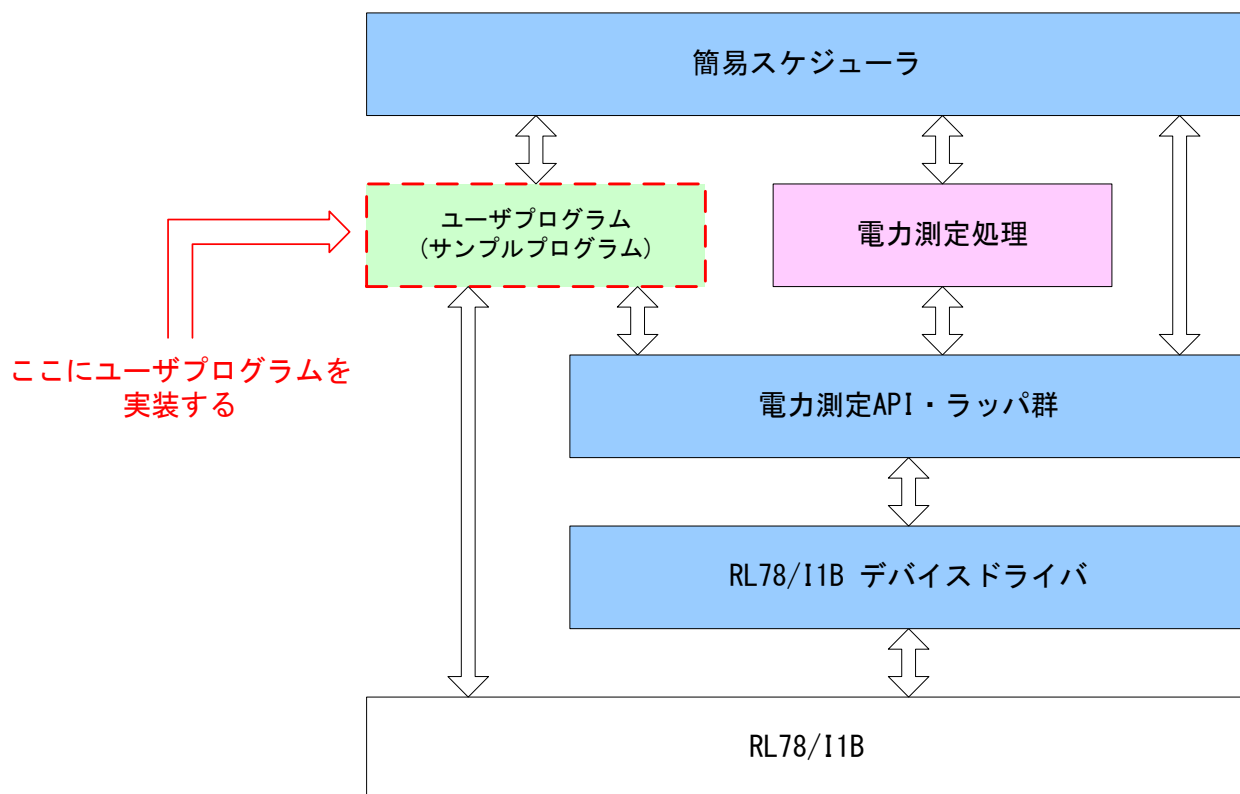


図 3-1 ソフトウェア構成図

ユーザプログラム I/F は簡易スケジューラによって順番に呼び出されます。
本ソフトウェアは、図 3-1 で「サンプルプログラム」と記載されているように、デフォルト状態ではスイッチ/LCD 操作とコンソールプログラムの二つのサンプルプログラムが組み込まれています。
お客様が独自のプログラムをユーザプログラム I/F に実装するには、これらサンプルプログラムを無効にする必要があります。

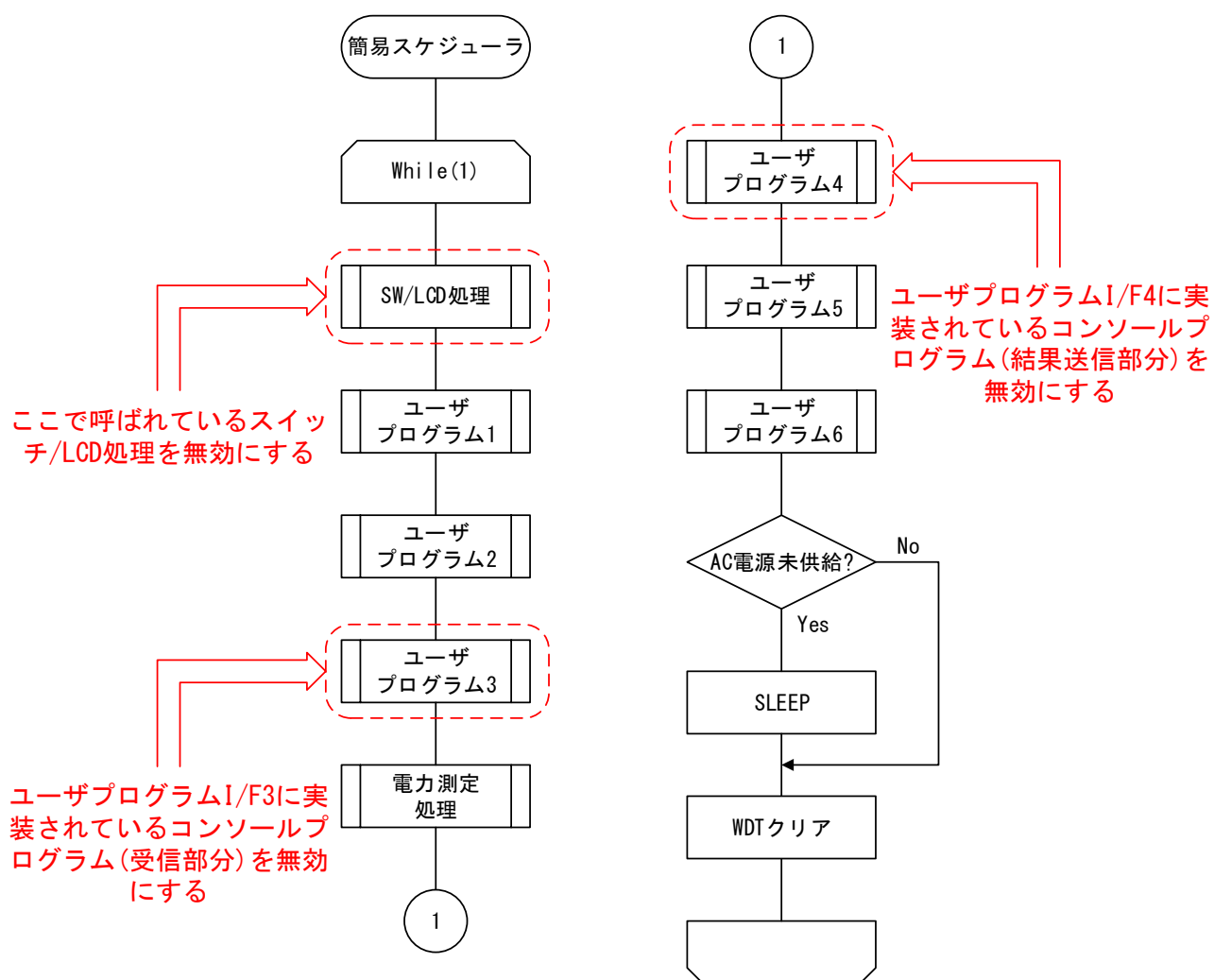


図 3-2 簡易スケジューラ フローチャート

以下、二つのサンプルプログラムを無効化する方法と、本ソフトウェアが提供する機能をユーザプログラムで使用方法を解説します。

また、本ソフトウェアが提供する電力測定 API (グローバル変数、マクロ) や各種関数はヘッダファイル `include¥em_common.h` にて宣言されています。

この `include¥em_common.h` はユーザプログラム I/F 関数が実装されているソースファイルである `user_if¥user_if.c` の冒頭でインクルードしているため、ヘッダファイルのインクルードや関数のプロトタイプ宣言をお客様側で意識する必要はありません。

3.1. サンプルプログラムの無効化

以下に「スイッチ/LCD 操作」と「コンソールプログラム」の二つのサンプルプログラムを無効化する方法を示します。

1) スイッチ/LCD 操作

スイッチ/LCD 操作は `em_lcd_sw` という関数に実装されています。
この関数は簡易スケジューラ (`em_simple_scheduler`) で呼ばれていますので、これを無効化します。表 3-1 に示す修正を行って下さい。

表 3-1 スイッチ/LCD 操作 修正箇所

修正ファイル	修正箇所	備考
<code>em¥em_simple_scheduler.c</code>	関数 <code>em_simple_scheduler</code> 内、72 行目を『 <code>#if 0</code> 』から『 <code>#if 1</code> 』に修正	二箇所とも修正して下さい
<code>include¥em_config.h</code>	57 行目、『 <code>#define EM_ENABLE_SW_LCD (1)</code> 』をコメントアウト	

2) コンソールプログラム

コンソールプログラムはユーザプログラム I/F3 とユーザプログラム I/F4 に実装されています。
ユーザプログラム I/F3 とユーザプログラム I/F4 の関数内をすべてコメントアウトするか削除することで無効化します。また、`r_cg_sau_user.c` で拡張シリアル I/F の制御を行っている処理に関しても、コメントアウトまたは削除して下さい。

表 3-2 コンソールプログラム 修正箇所

修正ファイル	修正箇所	備考
<code>user_if¥user_if.c</code>	関数 <code>em_user_program3</code> 内、476 行目～652 行目を削除またはコメントアウト	全修正箇所を修正して下さい
	関数 <code>em_user_program4</code> 内、665 行目～693 行目を削除またはコメントアウト	
<code>cg_src¥r_cg_sau_user.c</code>	72 行目、73 行目のグローバル変数の宣言を削除またはコメントアウト	
	関数 <code>r_uart2_callback_receiveend</code> 内、345 行目、346 行目を削除またはコメントアウト	
	関数 <code>r_uart2_callback_sendend</code> 内、370 行目を削除またはコメントアウト	

3.2. LCD 切替スイッチの使用方法

本製品に搭載されている LCD 切替スイッチは RL78/I1B のタイマ・アレイ・ユニット チャンネル 1 に接続されています。

本ソフトウェアでは、タイマ・アレイ・ユニット チャンネル 1 を外部イベント・カウンタ (両エッジ検出、規定カウント数 1) として使用し LCD 切替スイッチの押下を検出しています。

LCD 切替スイッチの押下は『短押し』と『長押し』の 2 パターンで検出できます。

タイマ・アレイ・ユニット チャンネル 2 をインターバルタイマとして使用し、押下時間を計測しています。表 3-3 に LCD 切替スイッチの押下時間の詳細を示します。

表 3-3 LCD 切替スイッチ 押下時間判定

押下時間	判定
0.1 秒未満	無効
0.1 秒以上、2.0 秒未満	短押し
2.0 秒以上	長押し

『長押し』の判定は LCD 切替スイッチを押し続けて 2.0 秒間経過した場合に行われます。これらの押下判定は電力操作 API の『LCD 切替スイッチ押下フラグ』を参照することで、ユーザプログラムで使用できます。

[変数] unsigned char em_g_sw

概要	LCD 切替スイッチ押下フラグ
説明	LCD 切替スイッチの押下状態が設定されます。LCD 切替スイッチは短押し (2 秒以内) と長押し (2 秒以上) に対応しています。また、電力測定処理の実行中は LCD 切替スイッチの入力は受け付けていません。
数値	0x00 : 押下なし 0x01 : 短押し 0x02 : 長押し
関連マクロ	EM_IS_SW_NOT_PUSH EM_IS_SW_SHORT_PUSH EM_IS_SW_LONG_PUSH EM_SW_CLEAR_PUSH
備考	なし

[マクロ名] EM_IS_SW_NOT_PUSH

概要	LCD 切替スイッチ押下フラグ確認 (押下なし)
説明	LCD 切替スイッチの押下がないことを確認します。
引数	なし
リターン値	0 : 押下あり 0 以外 : 押下なし
備考	em_g_sw の値を確認

[マクロ名] EM_IS_SW_SHORT_PUSH

概要	LCD 切替スイッチ押下フラグ確認(短押し)
説明	LCD 切替スイッチが短押しされたことを確認します。
引数	なし
リターン値	0 : 短押しされていない 0 以外 : 短押しされた
備考	em_g_sw の値を確認

[マクロ名] EM_IS_SW_LONG_PUSH

概要	LCD 切替スイッチ押下フラグ確認(長押し)
説明	LCD 切替スイッチが長押しされたことを確認します。
引数	なし
リターン値	0 : 長押しされていない 0 以外 : 長押しされた
備考	em_g_sw の値を確認

[マクロ名] EM_SW_CLEAR_PUSH

概要	LCD 切替スイッチ押下フラグクリア
説明	LCD 切替スイッチ押下フラグをクリア(押下なし状態に変更)します。
引数	なし
リターン値	なし
備考	em_g_sw の値を変更

以下にマクロを使用したサンプルコードを示します。

```

/* スイッチ処理 */
if (!EM_IS_SW_NOT_PUSH) {
    /* 押下あり */
    if (EM_IS_SW_SHORT_PUSH) {
        /* 短押し時の処理 */
    } else if (EM_IS_SW_LONG_PUSH) {
        /* 長押し時の処理 */
    }
    EM_SW_CLEAR_PUSH; /* スイッチ押下フラグクリア */
}

```

3.3. LCD の使用方法

本製品に搭載されている LCD は RL78/I1B の LCD コントローラ/ドライバ機能に接続されています。本ソフトウェアでは、LCD コントローラ/ドライバ機能を使用して本製品の LCD に文字を表示させるための関数を提供しています。ユーザプログラムで LCD に文字を表示したい場合はこれらの関数を利用して下さい。

[関数名] em_putc_lcd

概要	LCD 表示 (1 桁)
説明	引数 column で指定された桁に引数 c で与えられた文字を表示します。
引数	int column: 表示する桁番号 (左端が 7、右端が 0) char c: 表示する文字 (詳細は備考参照)
リターン値	なし
備考	表示可能な文字は以下の通り

アスキー コード	バイナリ データ	表示文字
'0' ~ '9'	0x30~0x39	0 ~ 9
'A' ~ 'F' または 'a' ~ 'f'	0x41~0x46 または 0x61~0x66	A, b, c, d, E, F
'H' または 'h'	0x48 または 0x68	H
'P' または 'p'	0x50 または 0x70	P
'U' または 'u'	0x55 または 0x75	U
'_'	0x2d	-
' '	0x20	(空白)
上記以外	—	設定禁止

[関数名] em_putd_lcd

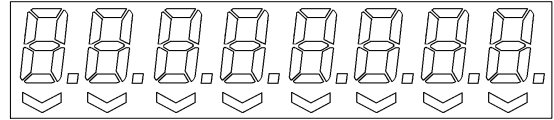
概要	LCD ドット表示
説明	引数 column で指定された桁のドットを点灯/消灯します。
引数	int column: 表示する桁番号 (左端が 7、右端が 0) char on: 1: 点灯、0: 消灯、それ以外: 設定禁止
リターン値	なし
備考	なし

[関数名] em_puts_lcd

概要	LCD 表示 (全桁一括)
説明	引数で与えられた 8 文字を LCD に表示します。
引数	char *s: 表示する文字列 (8 文字) (詳細は備考参照)
リターン値	なし
備考	表示可能な文字は、関数 em_putc_lcd と同じ

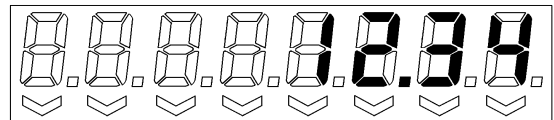
以下に LCD を表示させる関数のサンプルコードを示します。

```
/* 初期状態 */
```



```
/* 12.34 と表示 */
```

```
em_putc_lcd( 3, '1' );  
em_putc_lcd( 2, '2' );  
em_putc_lcd( 1, '3' );  
em_putc_lcd( 0, '4' );  
em_putd_lcd( 2, 1 );
```



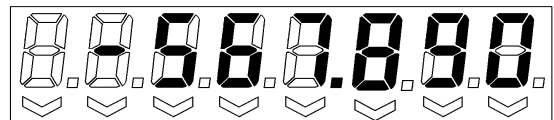
```
/* -567.890 と表示 */
```

```
char str[8];  
str[0] = ' ';  
str[1] = '-';  
str[2] = '5';  
str[3] = '6';  
str[4] = '7';  
str[5] = '8';  
str[6] = '9';  
str[7] = '0';
```

```
/* 2桁目のドットを消す */
```

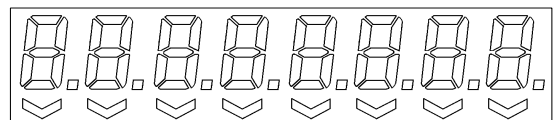
```
em_putd_lcd( 2, 0 );
```

```
em_puts_lcd( str );  
em_putd_lcd( 3, 1 );
```



```
/* 全桁、全ドットを消す */
```

```
int i;  
for( i = 0; 8 > i; ++i ) {  
    em_putc_lcd( i, ' ' );  
    em_putd_lcd( i, 0 );  
}
```



3.4. LED の使用方法

本製品に搭載されている LED は RL78/I1B のポート P30 と P31 に接続されています。本ソフトウェアでは、ポートを制御して本製品の LED を点灯/消灯/点滅させる関数を提供しています。また、点滅の制御にはタイマ・アレイ・ユニット チャネル 1 を使用しています。

[関数名] em_led_on

概要	LED 点灯
説明	引数で指定した LED を点灯させます。
引数	unsigned char n : 0 : LED 赤、1 : LED 黄、それ以外 : 設定禁止
リターン値	なし
備考	なし

[関数名] em_led_off

概要	LED 消灯
説明	引数で指定した LED を消灯させます。
引数	unsigned char n : 0 : LED 赤、1 : LED 黄、それ以外 : 設定禁止
リターン値	なし
備考	なし

[関数名] em_led_blink

概要	LED 点滅
説明	引数で指定した LED を点滅させます。
引数	unsigned char n : 0 : LED 赤、1 : LED 黄、それ以外 : 設定禁止
リターン値	なし
備考	なし

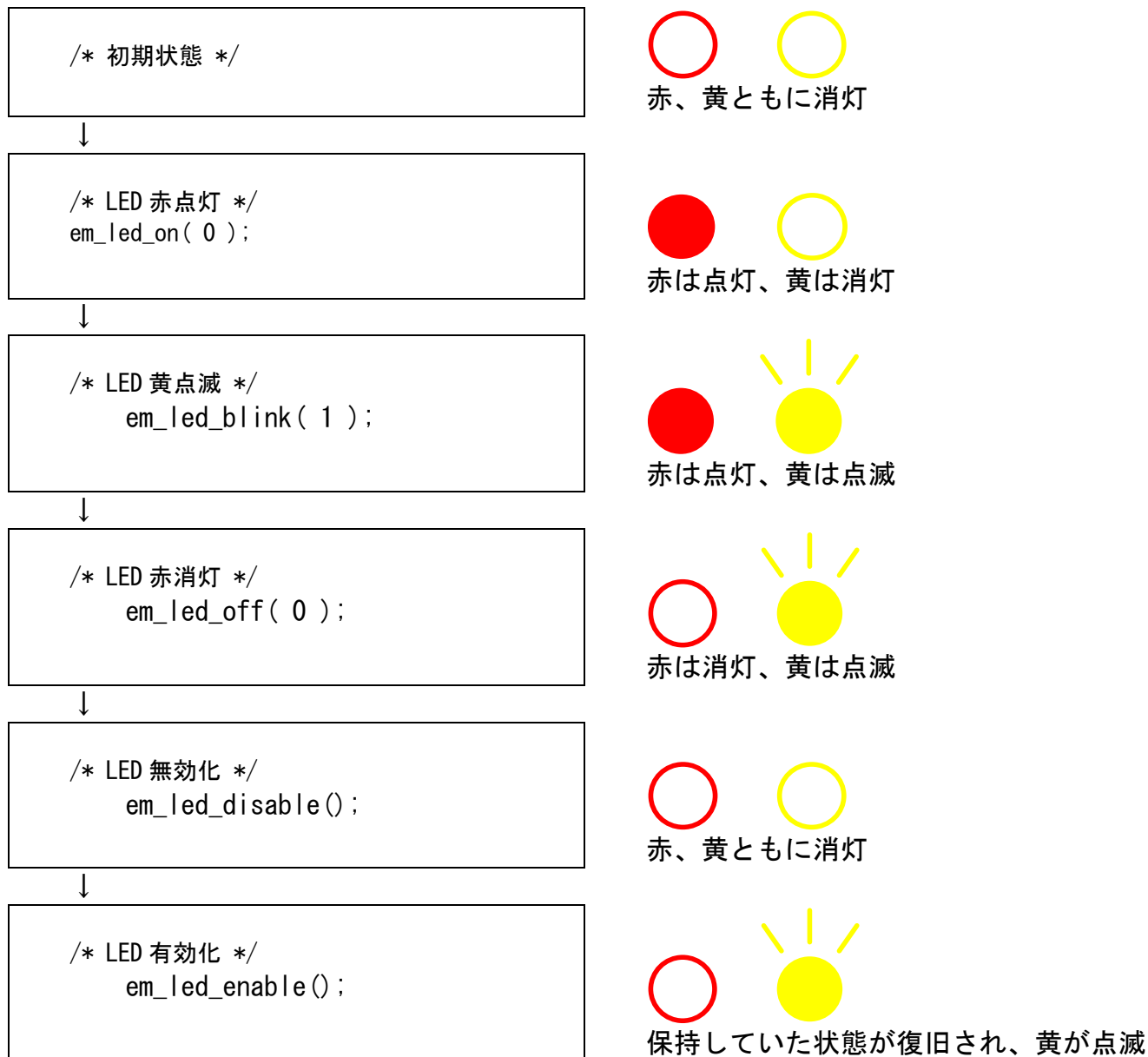
[関数名] em_led_enable

概要	LED 点灯/点滅有効化
説明	LED の点灯/点滅を有効にします。em_led_disable で無効化された点灯/点滅を内部に保持された点灯/点滅状態をもとに復旧します。初期状態で有効になっています。
引数	なし
リターン値	なし
備考	なし

[関数名] em_led_disable

概要	LED 点灯/点滅無効化
説明	LED の点灯/点滅を無効にします。LED の点灯/点滅状態は内部で保持され、em_led_enable を呼ぶ事で LED の点灯/点滅状態が復旧します。em_led_enable を呼ぶ前に em_led_on などの関数で LED を操作した場合は、操作した側の LED の無効化が自動的に解除されます。
引数	なし
リターン値	なし
備考	なし

以下に LED を点灯/消灯/点滅させる関数のサンプルコードを示します。



3. 5. EEPROM の使用方法

本製品に搭載されている EEPROM は RL78/I1B のシリアル・インタフェース IICA に接続されています。EEPROM のライトプロテクトはポート P62 で制御しています。またアドレスは使用しないため GND 固定となっており、IIC バス上のデバイスアドレスは 0 になります。図 3-3 に RL78/I1B と EEPROM の接続の概略図を示します。

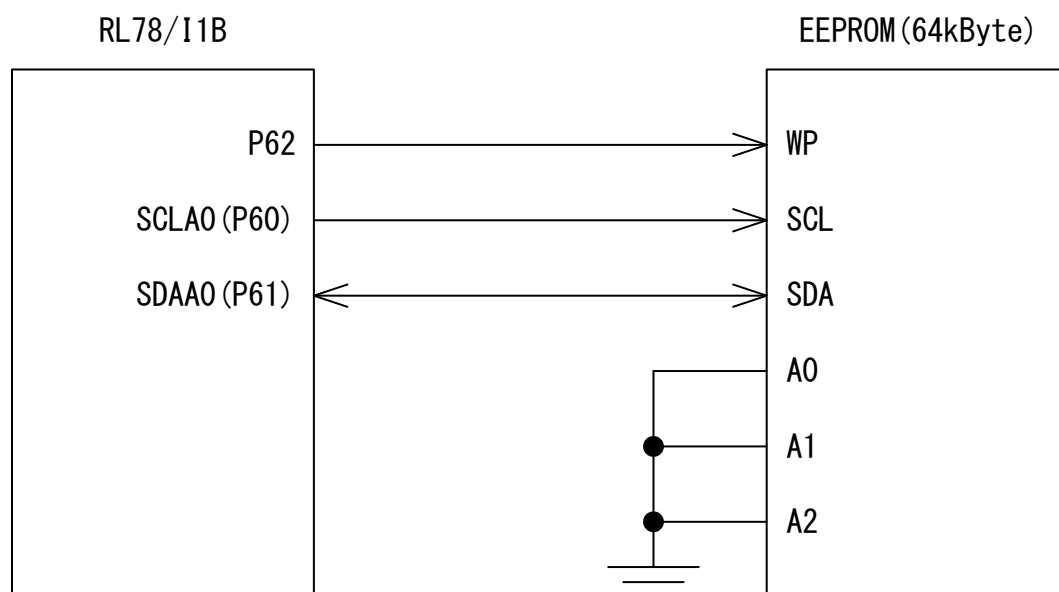


図 3-3 EEPROM 接続概略図

EV-200-USB-3 (RL78/I1B 評価キット)	2016/08/01	SJAA-150010-01	18/54
アプリケーションノート 基本編			

本ソフトウェアでは、シリアル・インタフェース IICA を制御して EEPROM に対して読み書きを行うための関数を提供しています。IIC の転送クロックは 100kHz に設定してあります。

[関数名] em_read_byte_eeprom

概要	EEPROM 読み出し (1Byte)
説明	IICA を用いて EEPROM から 1Byte 読み出します。
引数	unsigned short addr : 読み出すアドレス unsigned char *c : 読み出したデータを格納するポインタ
リターン値	0 : 正常終了 -1 : エラー
備考	なし

[関数名] em_read_eeprom

概要	EEPROM 読み出し (連続)
説明	IICA を用いて EEPROM から引数で指定されたバイト数だけ読み出します。
引数	unsigned short addr : 読み出し開始アドレス unsigned char *s : 読み出したデータを格納するポインタ unsigned int num : 読み出すバイト数 (最大 128 バイト)
リターン値	0 : 正常終了 -1 : エラー
備考	なし

[関数名] em_write_byte_eeprom

概要	EEPROM 書き込み (1Byte)
説明	IICA を用いて EEPROM へ 1Byte 書き込みます。
引数	unsigned short addr : 書き込むアドレス unsigned char *c : 書き込むデータ
リターン値	0 : 正常終了 -1 : エラー
備考	なし

[関数名] em_write_eeprom

概要	EEPROM 書き込み (連続)
説明	IICA を用いて EEPROM へ引数で指定されたバイト数だけ書き込みます。
引数	unsigned short addr : 書き込み開始アドレス unsigned char *s : 書き込むデータが格納されたポインタ unsigned int num : 書き込むバイト数 (最大 128 バイト)
リターン値	0 : 正常終了 -1 : エラー
備考	なし

以下に EEPROM を読み書きする関数のサンプルコードを示します。

```
/* 初期状態 */
```

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
10	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
20	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
30	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
40	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
50	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
60	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
70	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF
80	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF

初期状態の EEPROM

```
/* 0 番地から 128Byte 書き込み */
char s[128]
int i, ret;
for (i = 0; 128 > i; ++i) {
    s[i] = (char)i;
}
ret = em_write_eeprom( 0x00, s, 128 );
```

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
10	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
20	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
30	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F
40	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
50	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F
60	60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F
70	70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F
80	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF

0~127 番地のデータが更新

```
/* 任意の番地から読み出し */
char r[4]
int ret;
ret = em_read_byte_eeprom(0x01, &(r[0]));
ret = em_read_byte_eeprom(0x37, &(r[1]));
ret = em_read_byte_eeprom(0x6F, &(r[2]));
ret = em_read_byte_eeprom(0x80, &(r[3]));

/* r[0] = 0x01, r[1] = 0x37, r[2] = 0x6F
   r[3] = 0xFF になる */
```

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
10	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
20	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
30	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F
40	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F
50	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F
60	60	61	62	63	64	65	66	67	68	69	6A	6B	6C	6D	6E	6F
70	70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	7F
80	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF

0x01, 0x37, 0x6F, 0x80 番地の
データが読み出される

3.6. WDT 処理

本ソフトウェアでは WDT が起動しているため、WDT のタイムアウトが発生する前にクリアする必要があります。WDT のタイムアウト時間は 3799.18ms に設定されていますが、オプションバイトで変更可能です。

また、関数 `em_user_wdt` 内に WDT が発生したときに実行したい処理を記載することができます。

[関数名] `em_clear_wdt`

概要	WDT カウンタクリア
説明	ウォッチドッグ・タイマのカウントをクリアします。
引数	なし
リターン値	なし
備考	なし

[関数名] `em_user_wdt`

概要	WDT 処理 I/F
説明	ユーザの WDT タイムアウト処理を記述します。
引数	なし
リターン値	なし
備考	なし

3.7. 電力測定結果の参照

本ソフトウェアは RL78/I1B 搭載の $\Delta\Sigma$ ADC を用いて電圧と電流をサンプリングし、電力計算を行っています。表 3-4 に $\Delta\Sigma$ ADC の設定項目と対応する電力測定 API のグローバル変数を示します。

表 3-4 $\Delta\Sigma$ ADC と電力測定 API の対応

$\Delta\Sigma$ ADC 設定項目	電力測定 API	概要
サンプリング周波数	em_g_sampling_freq	4kHz サンプリング・モードと 2kHz サンプリング・モードを選択可能
電流チャンネル選択	em_g_current_device	CT が接続されているチャンネル 0 とシャント抵抗が接続されているチャンネル 2 を選択可能
プリアンプ・ゲイン 選択	em_g_volt_gain	1 倍, 2 倍, 4 倍, 8 倍, 16 倍から選択可能 プリサンプリングにより、自動的に適切なゲインを選択するオートゲインモードもソフトウェアで実装
	em_g_current_gain_ct	1 倍, 2 倍, 4 倍, 8 倍, 16 倍, 32 倍から選択可能 プリサンプリングにより、自動的に適切なゲインを選択するオートゲインモードもソフトウェアで実装
	em_g_current_gain_s	

また、図 3-4 に示す本ソフトウェアの電力測定タイミング概略図の各要素に対応する電力測定 API の設定項目を表 3-5 に示します。

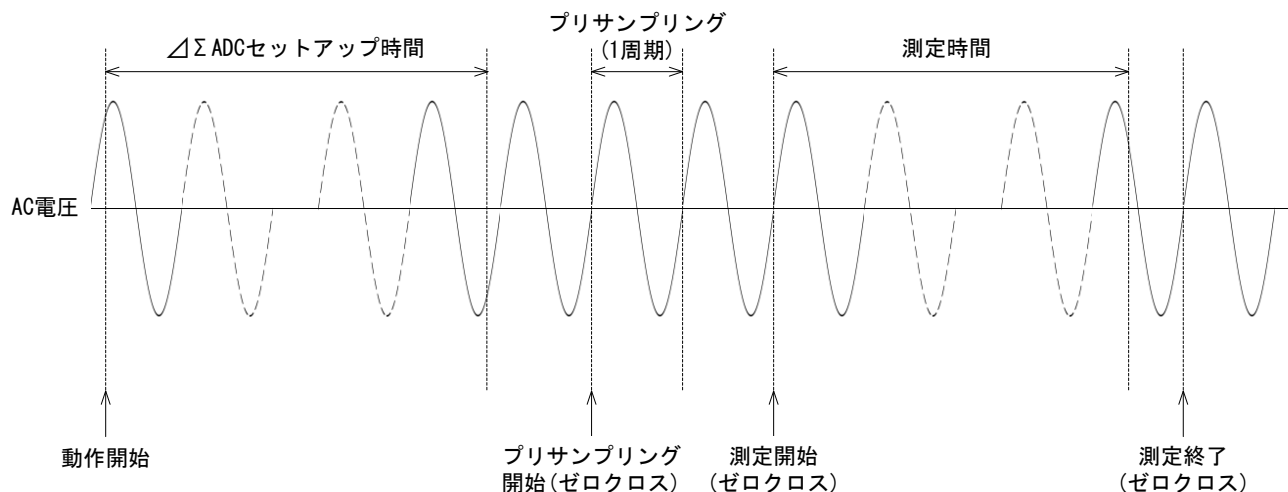


図 3-4 電力測定タイミング概略図

表 3-5 タイミング概略図と電力測定 API の対応

タイミング概略図 要素	電力測定 API	概要
動作開始	em_g_measure	電力測定の開始を設定します
△Σ ADC セットアップ時間	em_g_dsadc_init_wait	△Σ ADC のセットアップ時間(初期値は 150ms)
プリサンプリング	em_g_volt_gain em_g_current_gain_ct em_g_current_gain_s	ゲイン設定が全て固定ゲインモードの場合は、プリサンプリングは実行されません
ゼロクロス	em_g_zerocross_trigger	ゼロクロス検出をソフトウェアで行うか、ハードウェア (RL78/I1B のコンパレータ) で行うかを選択可能
	em_g_zerocross_th	ゼロクロス検出をソフトウェアで行う場合の閾値(電圧瞬時値)
測定時間	em_g_measure_time	電力の測定時間を設定します。0 を設定すると、1 周期分の測定になります。

電力測定 API の詳細に関しては、EV-200-USB-3 (RL78/I1B 評価キット) ユーザーズ・マニュアル ソフトウェア編を参照して下さい。

	2016/08/01	SJAA-150010-01	23/54
EV-200-USB-3 (RL78/I1B 評価キット)	アプリケーションノート 基本編		

電力測定処理の設定は電力測定の前に呼ばれるユーザプログラム I/F1~3 で行います。
以下に電力測定 API の設定処理のサンプルコードを示します。

```

/* ユーザプログラム 1~3 に実装することを想定しています。 */
void em_user_program1(unsigned char gs) {
    /* スリープモードの場合は処理をしない */
    if (EM_IS_SLEEP) {
        return;
    }

    /* 電源電圧復帰判定 */
    if (EM_IS_AC_RETURN(gs)) {
        /* 電源電圧復帰に関する処理がある場合はここに記述 */
        EM_CLEAR_AC_RETURN(gs);
    }

    /*  $\Delta\Sigma$  ADC のサンプリング周波数設定 */
    EM_SAMPLING_FREQ_4K; /* 4kHz に設定 */

    /*  $\Delta\Sigma$  ADC のセットアップ時間設定 */
    em_g_dsadc_init_wait = 15; /* 150ms に設定 */

    /* 電力測定時間を設定 */
    em_g_measure_time = 10; /* 1.0 秒に設定 */

    /* 電流測定デバイス選択 */
    EM_DEVICE_CT; /* CT を選択 */

    /* 各入力ゲインを設定 */
    EM_VGAIN_AUTO; /* 電圧入力ゲインを自動に設定 */
    EM_CGAIN_CT_AUTO; /* 電流 (CT) 入力ゲインを自動に設定 */
    EM_CGAIN_S_AUTO; /* 電流 (シャント抵抗) 入力ゲインを自動に設定 */

    /* ゼロクロス設定 */
    em_g_zerocross_trigger = 0x01; /* ゼロクロストリガをソフトウェアに設定 */
    em_g_zerocross_th = 11.3; /* ゼロクロストリガ閾値を 11.3V に設定 */

    /* 過電圧・過電流設定 */
    em_g_over_volt_th = 382.0; /* 過電圧閾値を 382.0V に設定 */
    em_g_over_current_th = 22.0; /* 過電流閾値を 22.0A に設定 */

    /* 電力測定処理開始 */
    em_g_measure = 0x01; /* 電力測定開始 */
}

```

電力の測定を実行した後、電力測定の結果を対応するグローバル変数から読み出すことができます。

表 3-6 測定結果と電力測定 API の対応

測定結果	電力測定 API	概要
有効電力	em_g_active_power	有効電力の測定結果が設定されています。そのまま読み出して使用して下さい。単位は W です。
無効電力	em_g_reactive_power	無効電力の測定結果が設定されています。そのまま読み出して使用して下さい。単位は var です。
皮相電力	em_g_apparent_power	皮相電力の測定結果が設定されています。そのまま読み出して使用して下さい。単位は VA です。
電圧	em_g_volt	電圧の測定結果が設定されています。そのまま読み出して使用して下さい。単位は V です。
電流	em_g_current	電流の測定結果が設定されています。そのまま読み出して使用して下さい。単位は A です。

また、電力測定処理中にエラーが発生した場合はエラーフラグにエラーの種類が設定されます。電力測定の結果を読み出す前に必ずエラーフラグを確認し、電力測定が正常終了したことを確認して下さい。エラーフラグは電力測定を実行する毎に更新されます。

表 3-7 電力測定 API エラーフラグ

エラーフラグ	値	意味	概要
em_g_error	0x0000 0000	正常	電力測定は正常に終了しました
	0x0000 0001	ゼロクロス未検出	ゼロクロスを 200ms 間検出できませんでした
	0x0000 0002	過電圧検出	過電圧閾値(em_g_over_volt_th)を超える電圧が入力されました
	0x0000 0004	過電流検出	過電流閾値(em_g_over_current_th)を超える電流が入力されました
	0x0000 0008	AC 電源断検出	電力測定中に AC 電源が遮断されました
	0x0000 0010	電力測定異常検出	電力測定中にデバイスの異常を検出しました(割り込みが発生しなくなった、 $\Delta\Sigma$ ADC の入力電圧範囲を超える電圧が入力された等)
	0x0000 0020	磁気検出	本製品に搭載されている磁気センサが磁気を検出しました
	0x0000 0040	ケース開放検出	本製品に搭載されているケース開閉検出スイッチがケースの開放を検出しました

	2016/08/01	SJAA-150010-01	25/54
EV-200-USB-3 (RL78/I1B 評価キット)	アプリケーションノート 基本編		

電力測定処理の結果の取得は電力測定の後に呼ばれるユーザプログラム I/F4~6 で行います。

以下に電力測定結果の取得処理のサンプルコードを示します。

```

/* ユーザプログラム 4~6 に実装することを想定しています。 */
void em_user_program4(unsigned char gs ) {
    char str[64];

    /* スリープモードの場合は処理をしない */
    if (EM_IS_SLEEP) {
        return;
    }

    /* エラー判定 */
    if (EM_IS_SUCCESS) {
        /* 正常時の処理 */
        sprintf( str, "%5f, %5f, %5f, %5f, %5f\n", em_g_active_power, /* 有効電力 */
            em_g_reactive_power, /* 無効電力 */
            em_g_apparent_power, /* 皮相電力 */
            em_g_volt, /* 電圧 */
            em_g_current ); /* 電流 */

    }
    else if (EM_IS_ZEROCROSS_ERR) {
        /* ゼロクロスエラー時の処理 */
    }
    else if (EM_IS_OVER_VOLT_ERR) {
        /* 過電圧エラー時の処理 */
    }
    else if (EM_IS_OVER_CURRENT_ERR) {
        /* 過電流エラー時の処理 */
    }
    else if (EM_IS_AC_OFF_ERR) {
        /* AC 電源断エラー時の処理 */
    }
    else if (EM_IS_DEVICE_ERR) {
        /* 電力測定エラー時の処理 */
    }
    else if (EM_IS_MAGNETIC_ERR) {
        /* 磁気センサエラー時の処理 */
    }
    else if (EM_IS_CASE_OPEN_ERR) {
        /* ケース開放時の処理 */
    }
}

```

4. 拡張シリアル I/F

本製品に搭載されている拡張シリアル I/F は RL78/I1B の UART2 を使用して通信を行うことができます。他に、制御信号として RTS と CTS、汎用 I/O として使えるポートを一つ備えています。

また、電源は外部より供給する必要があります。

4.1. I/F 仕様

以下に、拡張シリアル I/F の規格について記述します。

表 4-1 拡張シリアル I/F コネクタの規格

項目	説明
メーカー	日本航空電子
品名	PS-8PLB-D4LT1-FL1E

※相手側の推奨コネクタは、以下になります。

ハウジング : PS-8SLA-D4C2

コンタクト : PS-SLA-C2-1-5000

表 4-2 拡張シリアル I/F 信号

ピン 番号	本製品		
	信号名	入出力	RL78/I1B 端子
1	VIN	—	
2	GND	—	
3	SERIAL_TXD	Out	P01/TxD2
4	SERIAL_RXD	In	P00/RxD2
5	SERIAL_GPIO	InOut	P32
6	SERIAL_RTS	Out	P33
7	SERIAL_CTS	In	P137/INTP0
8	GND	—	

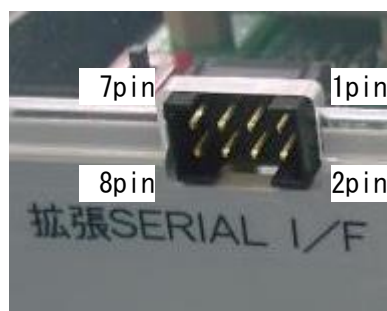


図 4-1 拡張シリアル I/F コネクタのピン配列

4.2. ブロック図

以下に、拡張シリアル I/F のブロック図を記載します。

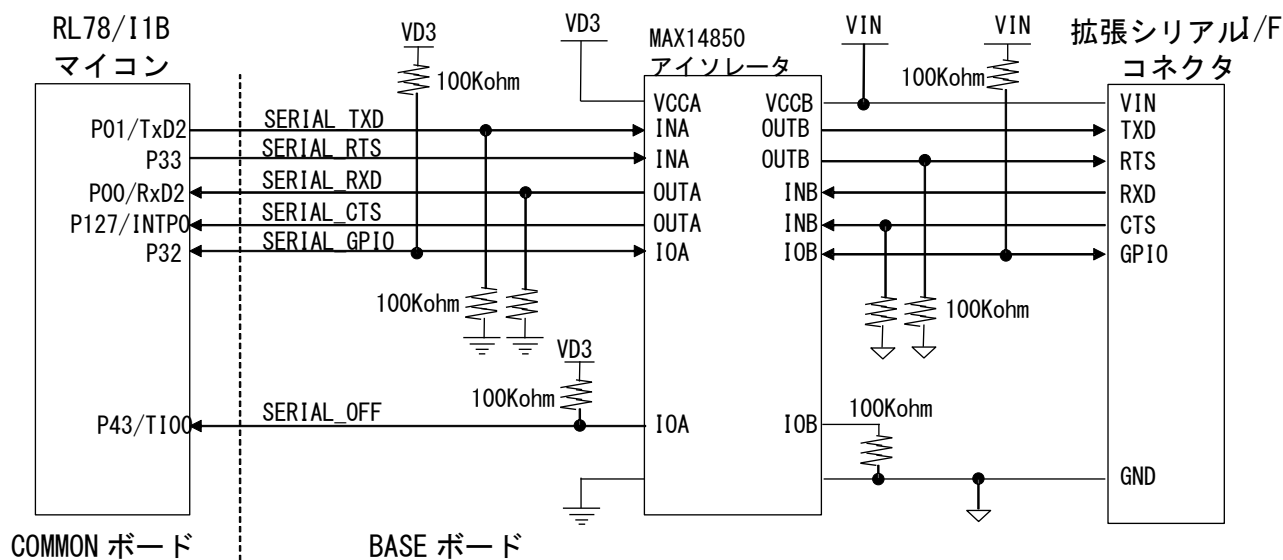


図 4-2 拡張シリアル I/F のブロック図

・拡張シリアル I/F と RL78/I1B はアイソレータ MAX14850 (Maxim 製) で絶縁されています。

・SERIAL_OFF 信号は拡張シリアル I/F が有効であるか確認する信号です。拡張シリアル I/F を使用しないとき High になり、拡張シリアル I/F から電源が供給されると Low になります。

RL78/I1B のタイマ・アレイ・ユニットのチャンネル 0 のタイマ入力端子に接続しています。タイマ割り込み機能で拡張シリアル I/F が有効/無効を検出して下さい。

・SERIAL_CTS 信号は、RL78/I1B の外部割り込み 0 に接続しています。割り込み機能で相手側の送信可能/不可能を検出して下さい。

4.3. 電気的特性

以下に、拡張シリアル I/F の電気的特性について記述します。

[入力電源電圧特性]

項目	略号	条件	MIN.	TYP.	MAX.	単位
入力電圧レベル	VIN	–	3.0	–	5.5	V
供給電流	–	データ転送速度 2Mbps VIN=+5V	–	7.2	11	mA
		データ転送速度 2Mbps VIN=+3.3V	–	6.2	9.5	mA
		データ転送最大速度 VIN=+5V	–	15	22	mA
		データ転送最大速度 VIN=+3.3V	–	10	16	mA

以下のスペックは、アイソレータ MAX14850[※]のスペック表記になっています。

[入力出力端子の DC スペック]

項目	略号	条件	MIN.	TYP.	MAX.	単位
ハイ・レベル入力電圧	VIH	全ての端子	$0.7 \times V_{IN}$	–	VIN	V
ロウ・レベル入力電圧	VIL	端子 : RXD, CTS	0	–	0.8	V
		端子 : GPIO	0	–	$0.3 \times V_{IN}$	V
ハイ・レベル出力電圧	VOH	全ての端子 IOH = –4.0mA	VCCB –0.4	–	–	V
ロウ・レベル出力電圧	VOL	端子 : TXD, RTS IOH = 4.0mA	–	–	0.8	V
		端子 : GPIO IOH = 30.0mA	–	–	0.4	V

※詳細仕様は MAX14850 のマニュアルを参照して下さい。

<https://www.maximintegrated.com/jp.html>

EV-200-USB-3 (RL78/I1B 評価キット)	2016/08/01	SJAA-150010-01	29/54
アプリケーションノート 基本編			

以下のスペックは、RL78/I1B のマニュアルに対して動作電圧を 3.3V 固定としたスペック表記になっています。その他詳細仕様は RL78/I1B のマニュアルを参照して下さい。

[通信レートの AC スペック]

項目	略号	条件	HS (高速メイン) モード		LS (低速メイン) モード		単位
			MIN.	MAX.	MIN.	MAX.	
転送レート	-	最大転送レート理論値 fMCK = fCLK	-	4.0	-	1.3	Mbps

5. Pmod モジュール I/F

本製品には Pmod 規格インターフェースを持ったモジュールが接続可能な Pmod モジュール I/F をご用意しています。Pmod 規格は Type 2A と Type 4A に対応しています。本製品に添付されている Pmod ケーブルの 12 ピン側をモジュールと接続可能です。モジュールを本製品に固定するための方法は、十分検討してから行って下さい。また、組み込み例を 5.3 に記載してますので、参照して下さい。

本製品に搭載されている Pmod モジュール I/F は、RL78/I1B の UART0、UART1 または CSIO を使用して通信を行うことができます。他に、汎用 I/O として使えるポートを備えています。

5.1. I/F 仕様

図 5-1 のように、Pmod モジュール I/F のコネクタは COMMON ボードに実装されています。図 5-2 のように、Pmod モジュール I/F のコネクタと Pmod ケーブルの 13 ピン側を接続し、Pmod ケーブルの 12 ピン側とモジュールを接続します。

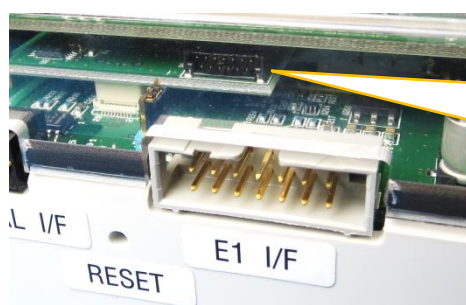


図 5-1 Pmod モジュール I/F コネクタの外観



図 5-2 Pmod ケーブルの接続

以下に、Pmod ケーブル(モジュール側)の規格について記述します。

表 5-1 Pmod ケーブル(モジュール側)の規格

項目	説明
メーカー	日本航空電子
品名	ハウジング : PS-12SLA-D4C2 コンタクト : PS-SLA-C2-1-5000

※相手側の推奨コネクタは、ストレートタイプの PS-12PLB-D4T1-FL1E、又はライトアングルタイプの PS-12PLB-D4LT1-FL1E です。

(2.54mm ピッチのピンヘッダでも代替え可能です。)

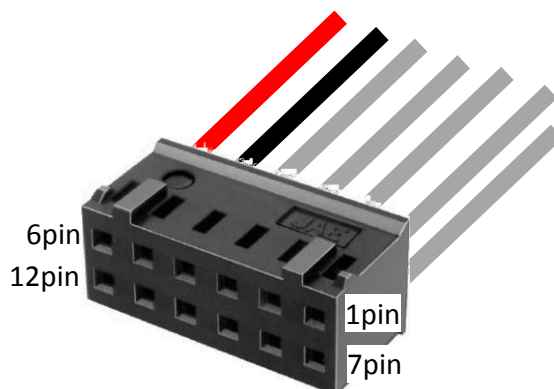


図 5-3 Pmod ケーブルのコネクタ側のピン配列

信号名について記述します。

表 5-2 Pmod ケーブル(モジュール側)の信号一覧

ピン 番号	Type 2A	Type 4A	本製品		
	信号名	信号名	信号名	入出力	RL78/I1B 端子
1	SS	CTS	PMOD_SS/CTS	In	P125/INTP1
2	MOSI	TXD	PMOD_MOSI/TXD	Out	P07/S000/TxD0
3	MISO	RXD	PMOD_MISO/RXD	In	P06/SI00/RxD0
4	SCK	RTS	PMOD_SCK/RTS	Out	P05/_SCK00
5	GND	GND	GND	—	
6	VCC	VCC	VD3	—	
7	INT	INT	PMOD_INT	In	P02/INTP5
8	RESET	RESET	PMOD_RESET	Out	P130
9	N/S	N/S	PMOD_RXD	—	P03/RxD1
10	N/S	N/S	PMOD_TXD	—	P04/TxD1
11	GND	GND	GND	—	
12	VCC	VCC	VD3	—	

※N/S : 空き端子

※RL78/I1B の端子とコネクタはダイレクトに接続されています。

EV-200-USB-3 (RL78/I1B 評価キット)	2016/08/01	SJAA-150010-01	32/54
	アプリケーションノート 基本編		

Type2A の場合、

- 1 ピンは出力ポートとして SS 信号を P125/INTP1 で制御します。
- 2～4 ピンは CSI00 として使用して下さい。
- 7～10 ピンは Type2A と Type4A で共通の使用方法になります。
- 7 ピンは外部割込み機能を使用して割り込み (INTP5) を発生させることができます。割り込みが必要ない場合は入力ポートとして使用し、信号レベルを確認することも可能です。
- 8 ピンは出力ポートとして RESET 信号を P130 で制御します。
- 9 ピン、10 ピンは Pmod 規格としては空き端子ですが、それぞれ P03、P04 に接続されています。そのまま汎用 I/O ポートとして使用するか、UART1 としても使用可能です。その場合は、9 ピンが RXD1、10 ピンが TXD1 となります。

Type4A の場合、

- 1 ピン CTS 信号は外部割込み機能を使用して割り込み (INTP1) を発生させることができます。割り込みが必要ない場合は入力ポートとして使用し、信号レベルを確認することも可能です。
- 2 ピン、3 ピンを UART0 として使用して下さい。
- 4 ピンは出力ポートとして SCK 信号を P05/_SCK00 で制御します。
- 7～10 ピンは Type2A と Type4A で共通の使用方法になります。
- 7 ピンは外部割込み機能を使用して割り込み (INTP5) を発生させることができます。割り込みが必要ない場合は入力ポートとして使用し、信号レベルを確認することも可能です。
- 8 ピンは出力ポートとして RESET 信号を P130 で制御します。
- 9 ピン、10 ピンは Pmod 規格としては空き端子ですが、それぞれ P03、P04 に接続されています。そのまま汎用 I/O ポートとして使用するか、UART1 としても使用可能です。その場合は、9 ピンが RXD1、10 ピンが TXD1 となります。

5.2. 電氣的特性

以下に、Pmod I/F の電氣的特性について記述します。

[出力電圧特性]

項目	略号	条件	MIN.	TYP.	MAX.	単位
出力電圧レベル	VD3	-	3.135	3.3	3.465	V
供給電流	-	-	-	-	300	mA

以下のスペックは、RL78/I1B のマニュアルに対して動作電圧を 3.3V 固定としたスペック表記になっています。その他詳細仕様は RL78/I1B のマニュアルを参照して下さい。

[入力出力端子の DC スペック]

項目	略号	条件	MIN.	TYP.	MAX.	単位
ハイ・レベル入力電圧	VIH1	P02-P07, P125 通常入力バッファ	2.64	-	3.3	V
	VIH2	P03, P05, P06 TTL 入力バッファ	2.0	-	3.3	V
ロウ・レベル入力電圧	VIL1	P02-P07, P125, P130 通常入力バッファ	0	-	0.66	V
	VIL2	P03, P05, P06 TTL 入力バッファ	0	-	0.5	V
ハイ・レベル出力電圧	VOH1	P02-P07, P125, P130 IOH1 = -2.0mA	2.7	-	-	V
ロウ・レベル出力電圧	VOL1	P02-P07, P125, P130 IOL1 = 3.0mA	-	-	0.6	V

EV-200-USB-3 (RL78/I1B 評価キット)	2016/08/01	SJAA-150010-01	34/54
アプリケーションノート 基本編			

[TYPE 2A (CSI モード、マスタ・モード) の AC スペック]

項目	略号	条件	HS (高速メイン) モード		LS (低速メイン) モード		単位
			MIN.	MAX.	MIN.	MAX.	
SCKp サイクル・タイム	tKCY1	-	167	-	500	-	ns
SCKp ハイ、ロウ・レベル幅	tKH1, tKL1	-	tKCY1/2 -18	-	tKCY1/2 -50	-	ns
Slp セットアップ時間 (対 SCKp ↑)	tSIK1	-	42 ※	-	108 ※	-	ns
Slp ホールド時間 (対 SCKp ↑)	tKSI1	-	21 ※	-	21 ※	-	ns
SCKp ↓ → S0p 出力遅延時間	tKS01	-	-	25	-	25	ns

※tSIK1 と tKSI1 はケーブル遅延を考慮していますが、環境によって異なりますので、お客様にてご確認下さい。

[TYPE 4A (UART モード) の AC スペック]

項目	略号	条件	HS (高速メイン) モード		LS (低速メイン) モード		単位
			MIN.	MAX.	MIN.	MAX.	
転送レート	-	最大転送レート理論値 fMCK = fCLK	-	4.0	-	1.3	Mbps

5.3. モジュールの取り付け

モジュールを取り付ける場合の参考にお使い下さい。

2) 取り付け例にて、スペーサで固定する方法を記載しますが、
取り付け方法についてはお客様の責任で行って下さい。

1) エリア制限

本製品に取り付ける Pmod モジュールは下記の範囲に収まるようにして下さい。

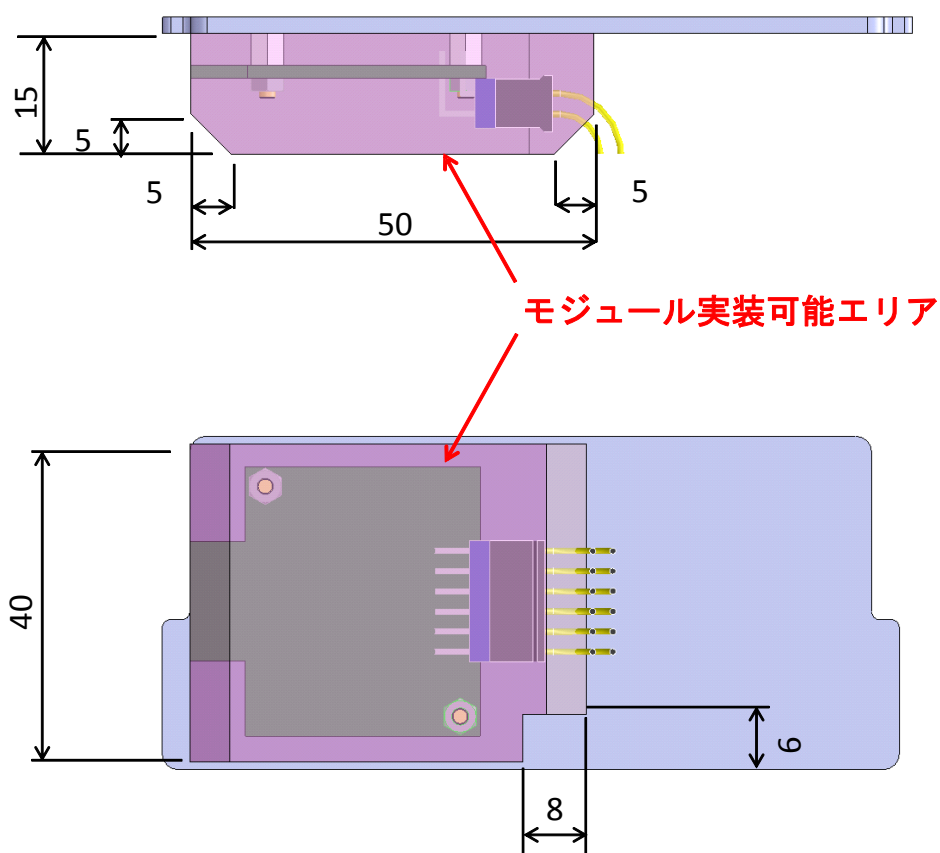


図 5-4 Pmod モジュールの組み込みエリア

2) 取り付け例

- a) 電源コードをコンセントから抜いて下さい。
- b) ケースの蓋を外して、内部に組み込まれている Pmod 固定板を取り外します。
- c) Pmod 固定板に穴を開けて、Pmod 固定板にモジュールを取り付けます。

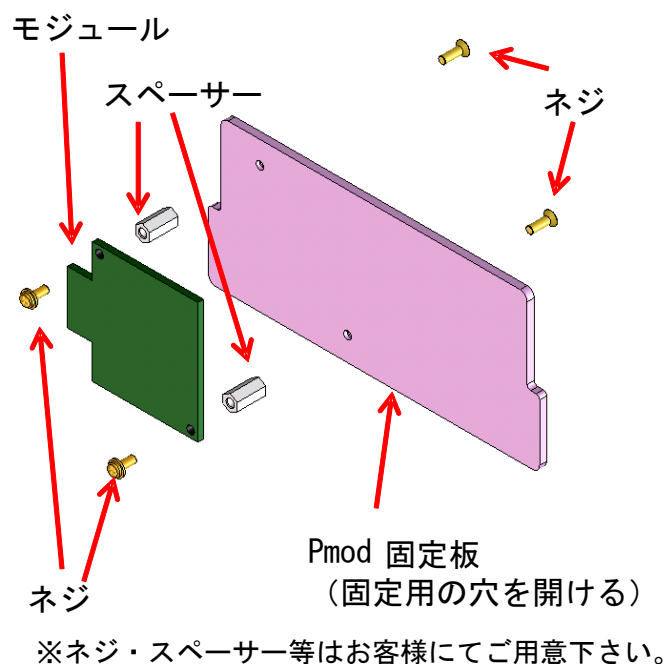


図 5-5 Pmod モジュール取り付け例

- d) モジュールと COMMON ボードを Pmod ケーブルで接続して下さい。
Pmod ケーブルの 1pin 向きには注意して下さい。
- e) モジュールと Pmod 固定板をケースに戻して下さい。その際に、Pmod ケーブルは Pmod 固定板などに挟まれていないか十分確認して下さい。



図 5-6 モジュール取り付け例

- f) ケースの蓋を取り付けて下さい。
締め付けトルクは $1\text{N}\cdot\text{m} \sim 1.5\text{N}\cdot\text{m}$ ($10\text{kgf}\cdot\text{cm} \sim 15\text{kgf}\cdot\text{cm}$) です。

6. 電池の交換

6.1. 概要

以下に、電源構成図を示します。

通常は AC 電源から動作電圧を生成していますが、AC 電源が切断されたときの予備バッテリーが搭載されています。

VBAT1 は RL78/I1B の VDD が供給されないとき、RL78/I1B をバッテリー・バックアップ・モードで動作させることができます。VBAT2 は AC 電源が切断されたときの予備バッテリーで、データ待避などの目的で使用して下さい。

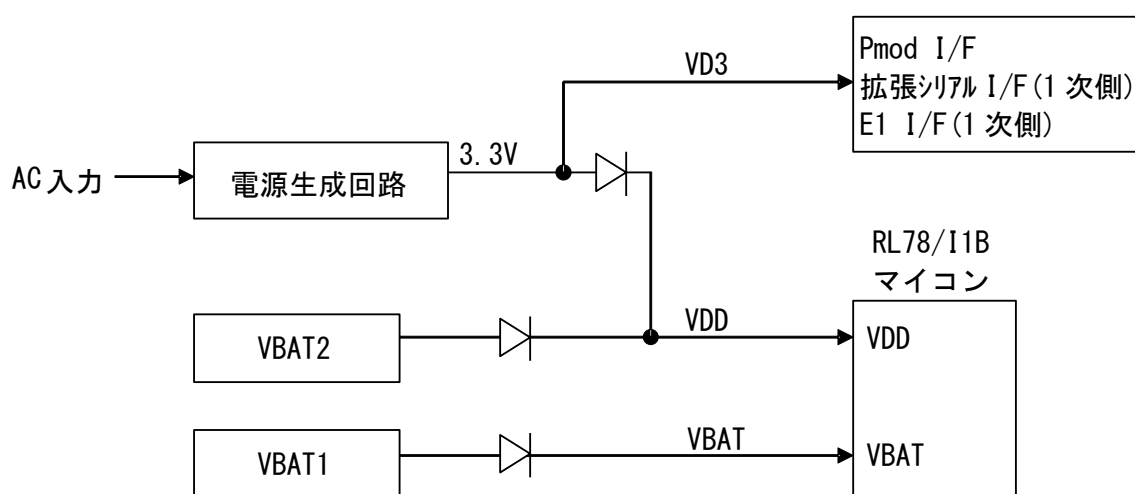


図 6-1 電源構成図

電池は、CR2032 が内蔵されております。電池が消耗した場合は、市販のものをご用意して下さい。

また、CR2032 の連続標準負荷は 0.2mA 程度ですので、AC 電源が切断されたら、すぐに RL78/I1B をスタンバイモードに移行するようにして下さい。

6.2. 電池の交換方法

- a) 電源コードをコンセントから外して下さい。
- b) 電池スイッチが OFF になっていることを確認して下さい。
- c) ケースの裏面にあるネジを以下の 6 カ所を外して下さい。

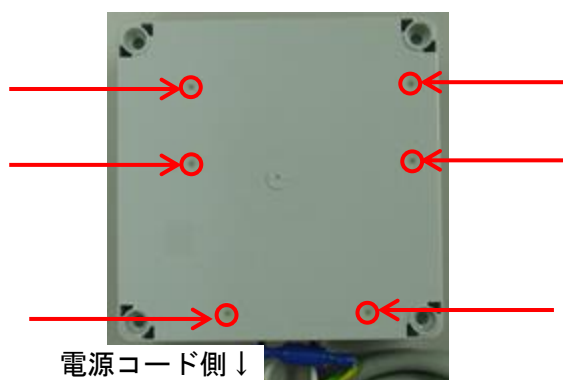


図 6-2 ネジの取り外し

- d) ケースを外して電池を交換して下さい。
VBAT1 と VBAT2 の位置は以下のようにになっています。

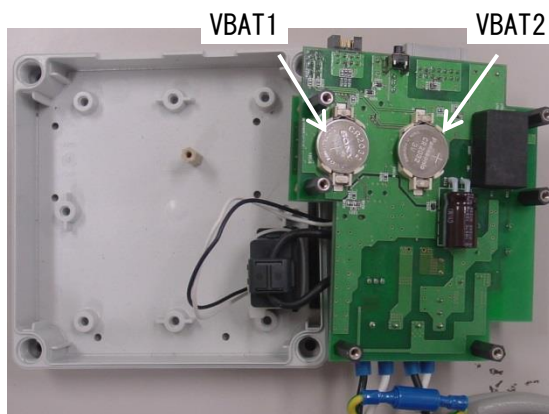


図 6-3 電池交換

- e) ケースに戻して、裏面からネジを止めて下さい。
- f) 電池スイッチを ON にして、電源コードをコンセントに接続して下さい。AC 電源が供給されていないときは、電池スイッチは極力 OFF にしておいて下さい。

7. 電力測定処理の速度改善

本ソフトウェアでは RL78/I1B を高速オンチップ・オシレータで動作させる場合に限り、電力測定処理前に高速オンチップ・オシレータ・クロック周波数補正機能を使用してクロック周波数の補正を行っています。

この高速オンチップ・オシレータ・クロック周波数補正機能はクロック周波数の補正に 5 秒以上の時間がかかる場合があります、電力測定処理の遅延の原因となっています。表 7-1 に電力測定のそれぞれの処理における最長の処理時間を示します。

表 7-1 電力測定処理の最長時間

処理	最長時間	備考
クロック補正	最長 5s 以上	通常は約 30ms ほどで終了
プリサンプリング	$\Delta\Sigma$ ADC セットアップ時間 + ゼロクロス検出タイムアウト時間 (200ms) + 電圧 1 周期分 (50Hz で約 20ms)	$\Delta\Sigma$ ADC セットアップ時間、 測定時間は電力測定 API で 任意の長さに変更可能
本サンプリング	$\Delta\Sigma$ ADC セットアップ時間 + ゼロクロス検出タイムアウト時間 (200ms) + 測定時間	

$\Delta\Sigma$ ADC セットアップ時間を初期値である 150 ミリ秒で電力測定処理を実行した場合、クロック補正にかかる時間は電力測定処理全体の中でもかなりの比率を占める事になります。

このクロック補正時間を短縮する二つの方法とプリサンプリングを省略し処理速度を短縮する方法を以下に示します。

7.1. 外部クロックを使用する

本製品は 12MHz の外部クロックを搭載しています。これを RL78/I1B のメイン・システム・クロックに設定することで、電力測定処理前にクロック周波数の補正は実行されなくなり、処理速度が向上します。

本ソフトウェアでは、メイン・システム・クロックの変更をするための関数を提供しています。

[関数名] em_select_clock_ext

概要	外部クロック選択
説明	RL78/I1B を外部クロック (12MHz) で動作させます。
引数	なし
リターン値	0 : 成功
備考	なし

〔関数名〕 em_select_clock_hoco

概要	高速オンチップ・オシレータ・クロック選択
説明	RL78/I1B を高速オンチップ・オシレータ・クロックで動作させます。動作周波数は引数 div に従います。
引数	unsigned char div: クロック分周比 (動作クロック = $24/2^{\text{div}}$)、 3 以上 : 設定禁止
リターン値	0: 成功 -1: 失敗 (引数エラー)
備考	なし

上記関数のうち、em_select_clock_ext 関数をユーザ初期化処理 I/F で呼び出せば、本製品を外部クロックで動作させることができます。
以下にメイン・システム・クロックの変更用関数のサンプルコードを示します。

```
/* 外部クロックを使用*/
em_select_clock_ext();
```

```
/* 高速オンチップ・オシレータ・クロックを使用 */
em_select_clock_hoco(1) /* 動作クロック = 24MHz / 2^1 = 12MHz */
```

関数 em_select_clock_hoco を使用すると、高速オンチップ・オシレータ・クロックを分周し、メイン・システム・クロックの周波数を、24MHz, 12MHz, 6MHz, 3MHz から選択することが出来ます。ただし、メイン・システム・クロックを 3MHz で動作させた場合、 $\Delta\Sigma$ ADC のサンプリング周波数が 4kHz の設定で電力測定処理を動作させると処理が間に合わず、電力測定異常のエラーとなってしまいます。
表 7-2 にメイン・システム・クロックと $\Delta\Sigma$ ADC のサンプリング周波数設定別の電力測定処理について示します。

表 7-2 電力測定処理

種別	周波数	$\Delta\Sigma$ ADC サンプリング周波数	
		2kHz	4kHz
外部クロック	12MHz	動作可能	動作可能
高速オンチップ・オシレータ・クロック	24MHz	動作可能	動作可能
	12MHz	動作可能	動作可能
	6MHz	動作可能	動作可能
	3MHz	動作可能	動作不可

本製品を高速オンチップ・オシレータ・クロック 3MHz で動作させる場合は、 $\Delta\Sigma$ ADC のサンプリング周波数を必ず 2kHz で動作させて下さい。

7.2. クロック補正処理を無効にする

クロック補正処理の無効化は RL78/I1B の仕様外での動作となりますので、ご注意ください。
クロック補正処理は `em_clock_corection` という関数に実装されています。

クロック補正処理を無効にする時は、他の部分まで変更しないように注意して下さい。
他の部分に手を加えた場合、電力測定処理が動作しなくなる可能性があります。また、
RL78/I1B の仕様では $\Delta\Sigma$ ADC を動作させる前に必ず高速オンチップ・オシレータ・クロックの周波数補正を行う事になっています。

[関数名] `em_clock_corection`

概要	高速オンチップ・オシレータ・クロック周波数補正
説明	高速オンチップ・オシレータ・クロックの周波数補正を行います。
引数	なし
リターン値	なし
備考	長い時間 (約 5 秒) 制御が戻らない可能性があります。本関数を呼ぶ場合は注意して下さい。

この関数を呼び出している `em_ep_measure_if` 関数の処理を変更してクロック補正処理を無効にし、処理速度を向上させます。表 7-3 に示す修正を行って下さい。

表 7-3 クロック補正処理の無効化 修正箇所

修正ファイル	修正箇所	備考
<code>em\em_ep_measure_if.c</code>	134 行目、『 <code>em_clock_corection();</code> 』をコメントアウト	修正して下さい

7.3. 固定ゲインモードを使用する

表 7-1 で示した処理の内、「プリサンプリング」は電力測定 API で $\Delta \Sigma$ ADC のプリアンプ・ゲインにオートゲインモードが設定されている場合に実行される処理です。

プリサンプリングでは、1 周期分のサンプリングを行い、測定された電圧・電流から適切なゲイン値を自動で設定するという処理を行っています。

この処理は電圧、電流 (CT)、電流 (シャント) の 3 つのゲイン設定が全て固定ゲインモードの場合は実行されません。

よって、3 つのゲイン設定を全て固定ゲインモードに設定することにより、プリサンプリング分の処理時間を短縮することができます。

以下に全てのゲインを固定ゲインモード (1 倍) に設定するサンプルコードを示します。

```
/* 全てのゲインを固定ゲインモード (1 倍) に設定することでプリサンプリング処理を省略 */
EM_VGAIN_1;          /* 電圧入力ゲインを 1 倍に設定 */
EM_CGAIN_CT_1;        /* 電流 (CT) 入力ゲインを 1 倍に設定 */
EM_CGAIN_S_1;         /* 電流 (シャント抵抗) 入力ゲインを 1 倍に設定 */
```

ユーザプログラムにおいて固定ゲインモードでゲイン値を設定する場合は、測定対象の電圧・電流に適したゲイン値を設定して下さい。

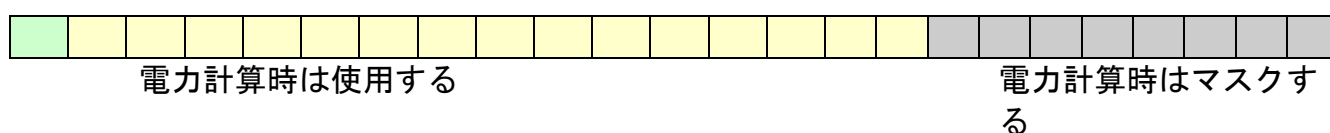
大きな倍率のゲイン値を設定すると電力測定時に過電圧検出・過電流検出や電力測定異常検出のエラーが発生する場合があります。

8. サンプリング値の変換係数

本ソフトウェアで使用している $\Delta\Sigma$ ADC のサンプリング結果から実際の電圧値・電流値に変換するための係数は以下のように算出します。

8.1. $\Delta\Sigma$ ADC の特性

RL78/I1B の 24bit $\Delta\Sigma$ ADC の下位 8bit は誤差の影響で値が不安定のため、実際の電力計算時はマスクしています。符号ビットを除いたサンプリング結果の全てのビットが 1 になった場合 (0x7FFF) の RL78/I1B に入力される電圧は 0.8V (基準電圧) になります。



0x7FFF (16bit のとき) = 0.8V

図 8-1 $\Delta\Sigma$ ADC サンプリング結果

このサンプリング結果 16 ビットの 1LSB あたりの電圧を計算すると 24.4148uV になります。

$$1\text{LSB あたりの電圧 (16bit)} = \frac{0.8\text{V}}{0x7FFF} = 24.4148\text{uV}$$

しかし、サンプリング結果は 24bit の下位 8bit をマスクしており、24bit レジスタとして扱っていますので、24bit 時の 1LSB あたりの電圧を計算すると 95.3703nV になります。

$$1\text{LSB あたりの電圧 (24bit)} = 24.4148\text{uV} \div 256 = 95.3703\text{nV}$$

算出された 1LSB あたりの電圧 (24bit) の値を使用して、実際の電圧と電流を求めるための係数を計算します。

8.2. 電圧用係数

本製品の電圧の入力部は図 8-2 のようになっています。

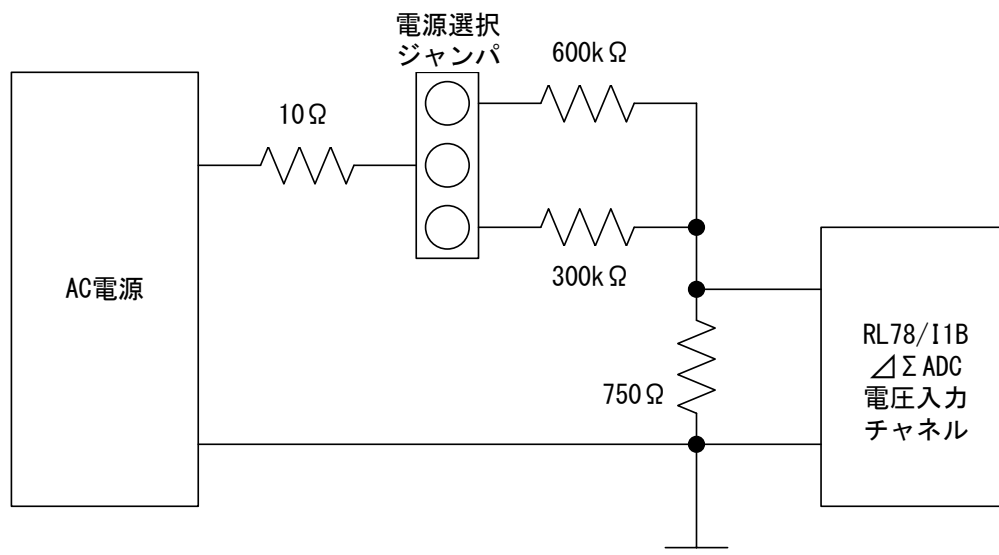


図 8-2 電圧入力部 概略図

電源選択ジャンパは出荷時の設定では、AC100V 入力用の 300kΩ が選択されるようにショートされていますので、RL78/I1B の ΔΣ ADC 電圧入力チャネルには 10Ω、300kΩ、750Ω の抵抗で分圧された電圧が入力されます。

$$\text{分圧比} = \frac{750\Omega}{10\Omega + 300\text{k}\Omega + 750\Omega} \div 0.0024937$$

AC 電源からの入力電圧の瞬時値が 1V であるとき、ΔΣ ADC に入力される電圧は、

$$\Delta\Sigma \text{ ADC の入力電圧} = 1\text{V} \times 0.0024937 = 2.4937\text{mV}$$

となります。この ΔΣ ADC の入力電圧をサンプリング結果のデジタル値に変換します。

$$1\text{V 入力時のサンプリング結果} = \frac{2.494\text{mV}}{95.3707\text{nV}} \div 26147$$

この 1V 入力時のサンプリング結果の数値 26147 が電圧を計算するときの係数となります。

8.3. 電流 (CT) 用係数

本製品の電流 (CT) の入力部は図 8-3 のようになっています。

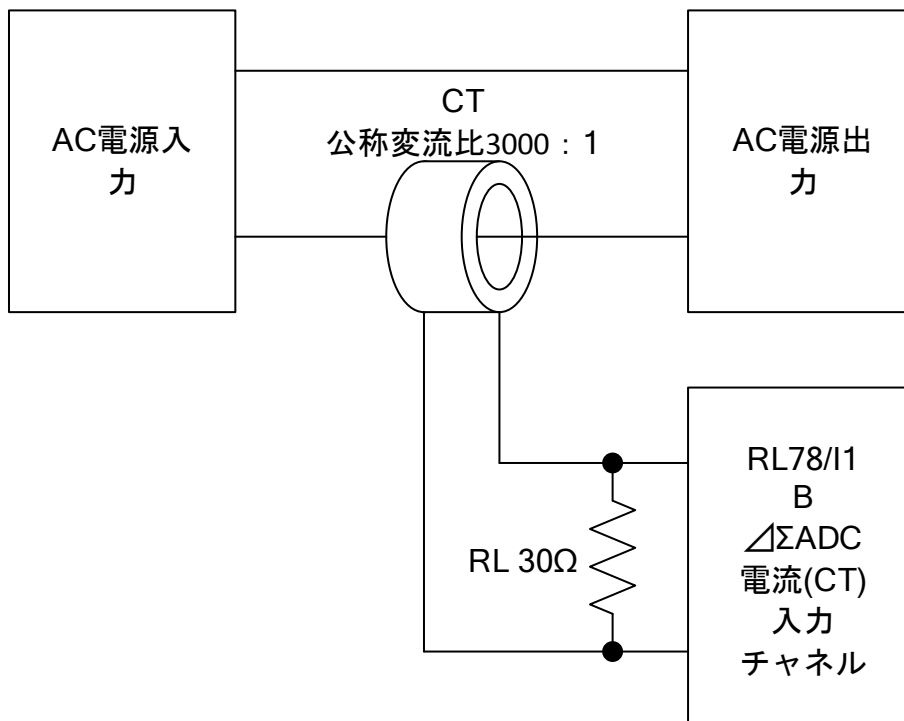


図 8-3 電流 (CT) 入力部 概略図

本製品に搭載している CT の公称交流比は 3000、RL は 30Ω となっています。AC 電源入力から AC 電源出力に流れる電流の瞬時値が 1A であるとき、 $\Delta\Sigma$ ADC に入力される電圧は

$$1\text{A 入力時の}\Delta\Sigma\text{ADC の入力電圧} = \frac{1\text{A} \times 30\Omega}{3000} = 10\text{mV}$$

となります。この $\Delta\Sigma$ ADC の入力電圧をサンプリング結果のデジタル値に変換します。

$$1\text{A 入力時のサンプリング結果} = \frac{10.0\text{mV}}{95.3674\text{nV}} \div 104854$$

この 1A 入力時のサンプリング結果の数値 104854 が電流 (CT) を計算するときの係数となります。

8.4. 電流(シャント)用係数

本製品の電流(シャント)の入力部は図 8-4 のようになっています。

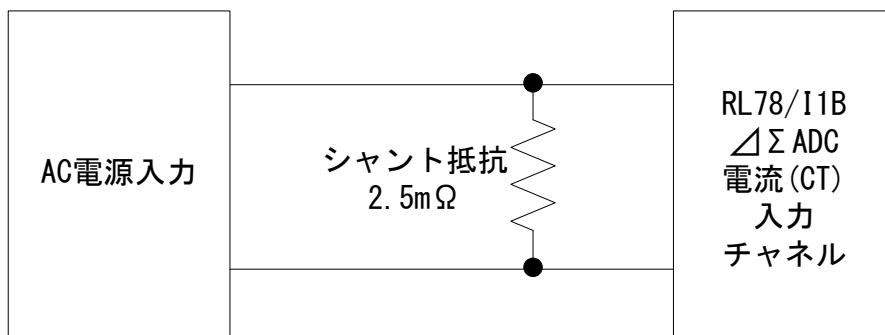


図 8-4 電流(シャント)入力部 概略図

本製品に搭載されているシャント抵抗は $2.5\text{m}\Omega$ となっています。AC 電源入力から流れる電流の瞬時値が 1A であるとき、 $\Delta\Sigma$ ADC に入力される電圧は

$$1\text{A 入力時の}\Delta\Sigma\text{ ADC の入力電圧} = 1\text{A} \times 2.5\text{m}\Omega = 2.5\text{mV}$$

となります。この $\Delta\Sigma$ ADC の入力電圧をサンプリング結果のデジタル値に変換します。

$$1\text{A 入力時のサンプリング結果} = \frac{2.5\text{mV}}{95.3674\text{nV}} \doteq 26213.6$$

この 1A 入力時のサンプリング結果の数値 26213.6 が電流(シャント)を計算するときの係数となります。

8.5. 誤差補正

ここまでで算出した変換係数を表 8-1 に示します。

表 8-1 変換係数

測定対象	係数
電圧	26147
電流 (CT)	104854
電流 (シャント)	26212.5

電圧、電流 (CT)、電流 (シャント) 用の係数はソースコードの `em_api.h` にて定義しています。58 行目 `EM_COEFF_VOLT` が電圧用係数、60 行目 `EM_COEFF_CT` が電流 (CT) 用係数、61 行目 `EM_COEFF_SHUNT` が電流 (シャント) 用係数になります。以下にサンプルコードを示します。

```

55 : #if defined(EM_MAX_VOLTAGE_200)
56 : #define      EM_COEFF_VOLT      (13090.0/(1<<EM_INEFFECTIVE_SHIFT))
57 : #else
58 : #define      EM_COEFF_VOLT      (26147.0/(1<<EM_INEFFECTIVE_SHIFT))
59 : #endif
60 : #define      EM_COEFF_CT      (104854.0/(1<<EM_INEFFECTIVE_SHIFT))
61 : #define      EM_COEFF_SHUNT
      (26212.5/(1<<EM_INEFFECTIVE_SHIFT))

```

これらの係数は部品バラツキなどを考慮してないため、実際の製品では個体ごとに誤差が発生しています。ここでは製品ごとの誤差を補正する方法を記載します。

例えば、校正された測定機器では 100V と表示されるのに、本製品では 97V と表示された場合、電圧の誤差率は-3%となります。そこで電圧用係数 26147 を 26955 に変更すると、誤差率が小さくすることができます。計算式は下記のように求めます。

$$\text{補正後の電圧用係数} = 26147 \times \frac{\text{校正された機器の測定結果}}{\text{本製品の測定結果}}$$

同様に電流 (CT) 用係数と電流 (シャント) 用係数も補正することができます。

9. 定格電源の変更

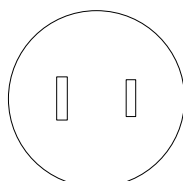
9.1. 規格について

本製品に添付されている電源コードの定格は 15A、125V です。本製品で定格 15A、250V を測定する場合は、電源コードを変更する必要があります。電源コードはお客様にてご用意して下さい。

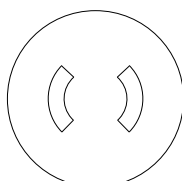
<参考資料>

参考として、以下に JIS 規格（JIS C 8303 配線用差込接続器）の中で本製品に対応している 2 極 15A、125V または 2 極 15A、250V のコンセント形状について記載します。

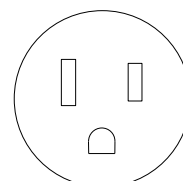
15A 125Vのコンセント形状



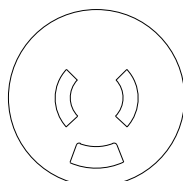
2極差込接続器



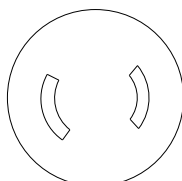
2極抜止形差込接続器



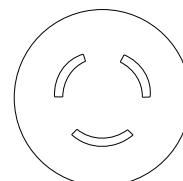
2極接地極付差込接続器



2極接地極付抜止形差込接続器

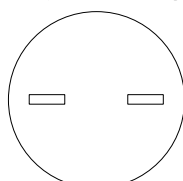


2極引掛形差込接続器

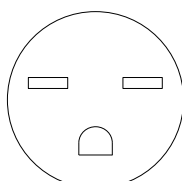


2極接地極付引掛形差込接続器

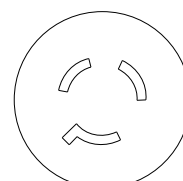
15A 250Vのコンセント形状



2極差込接続器



2極接地極付差込接続器



2極接地極付引掛形差込接続器

図 9-1 コンセント形状(参考)

9.2. ケーブルの交換と設定変更

- 電源コードをコンセントから抜いて下さい。
- 既存の電源コードを取り外します。
- CT 用電線の下にある動作電圧選択ジャンパを 1-2 ショートに切り替えます。

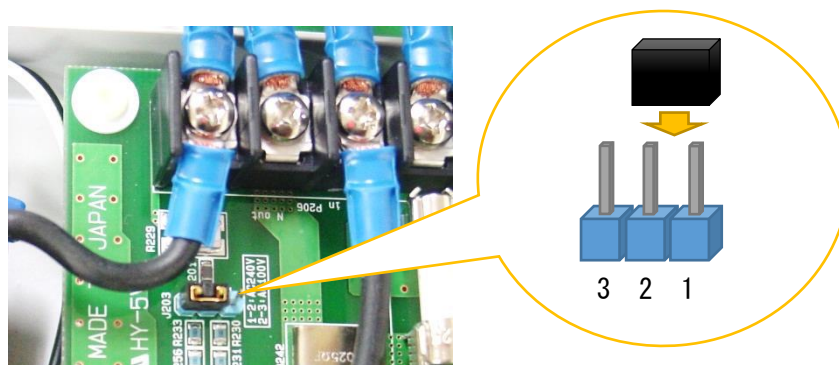


図 9-2 動作電圧選択ジャンパ (1-2 ショート時)

- 作成した定格 15A、250V の電源コードと取り付けます。AC 端子台の L 側、N 側と IN、OUT を注意しながら接続して下さい。
また、AC 端子台の推奨締付トルクは 0.5~0.8Nm です。

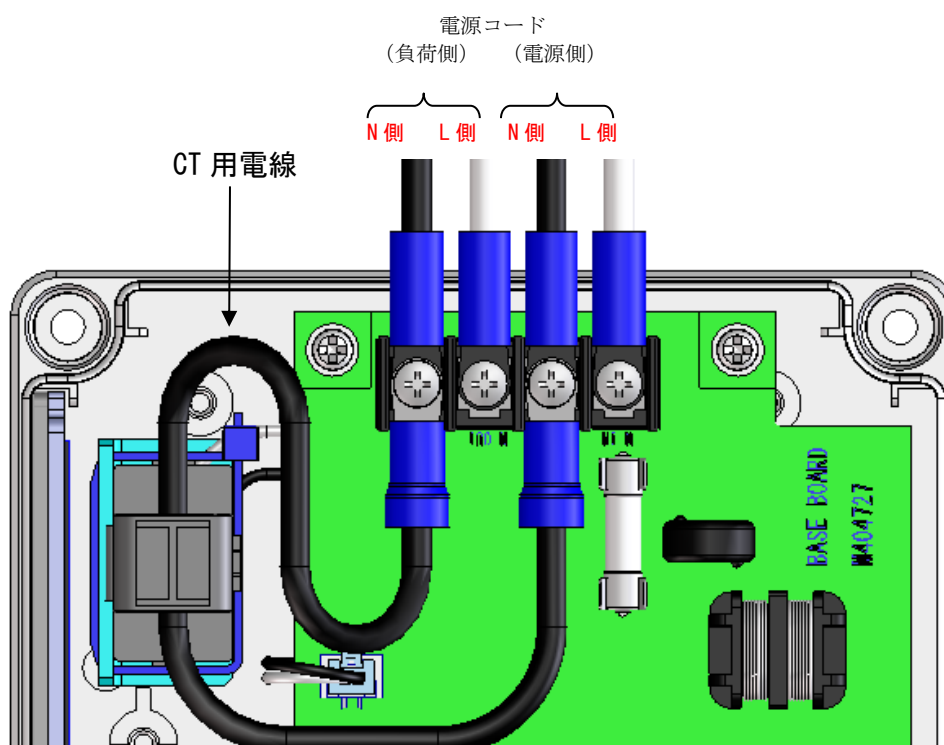


図 9-3 電源コードの取り付け図

EV-200-USB-3 (RL78/I1B 評価キット)	2016/08/01	SJAA-150010-01	50/54
アプリケーションノート 基本編			

9.3. ソフトウェアの変更箇所

本製品の動作電圧選択ジャンパを 1-2 ショートに変更し、入力電圧を AC180～250V に設定した場合、ソフトウェア側にも変更が必要になります。

入力電圧 AC180～250V 用にソフトウェアを変更するには、表 9-1 に示す修正を行って下さい。

表 9-1 入力電圧(100→200) 修正箇所

修正ファイル	修正箇所	備考
include¥em_config.h	46 行目、『#define EM_MAX_VOLTAGE_100』をコメントアウト	二箇所とも修正して下さい
	48 行目、『#define EM_MAX_VOLTAGE_200』を有効	

この変更により、em¥em_api.h で定義されている電圧用の係数が AC180～250V 用の値になります。

10. 付録

10.1. 各電力の計算式

以下に本ソフトウェアで使用している各電力の計算式を示します。

1) 有効電力

図 10-1 に本ソフトウェアの有効電力測定の流れを表したブロック図を示します。

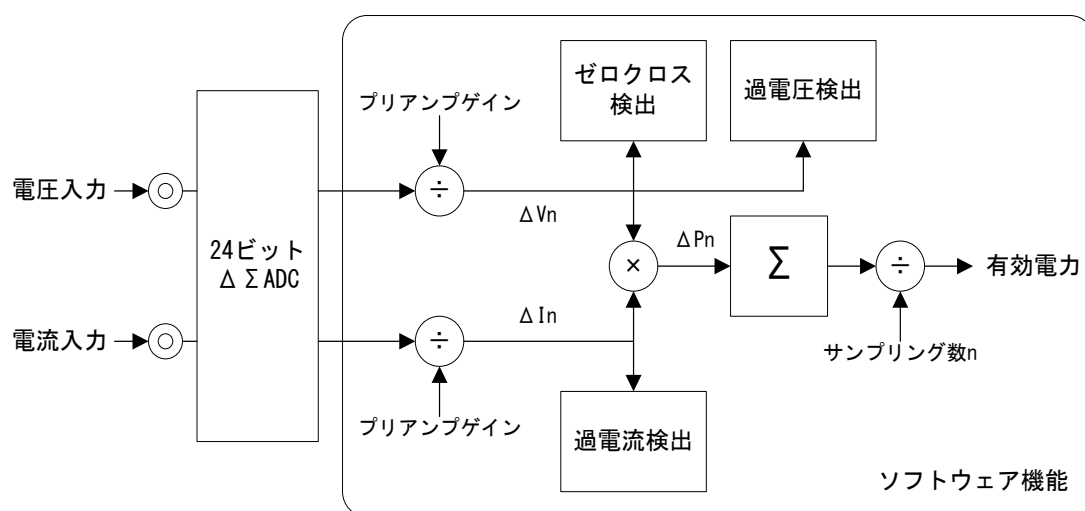


図 10-1 有効電力測定ブロック図

有効電力 P を計算するには、まず RL78/I1B の $\Delta\Sigma$ ADC でサンプリングされた瞬時電圧 ΔV_n と瞬時電流 ΔI_n の積から瞬時電力 ΔP_n を計算します。

$$\text{瞬時電力 } \Delta P_n = \Delta V_n \times \Delta I_n$$

計算された瞬時電力 ΔP_n をサンプリング毎に積算していきます。測定時間が経過しサンプリングが終了した後、瞬時電力の積算値をサンプリング数 n で割ることで有効電力 P が計算されます。

$$\text{有効電力 } P = \frac{\Delta P_1 + \Delta P_2 + \dots + \Delta P_n}{n}$$

2) 皮相電力

図 10-2 に本ソフトウェアの皮相電力測定の流れを表したブロック図を示します。

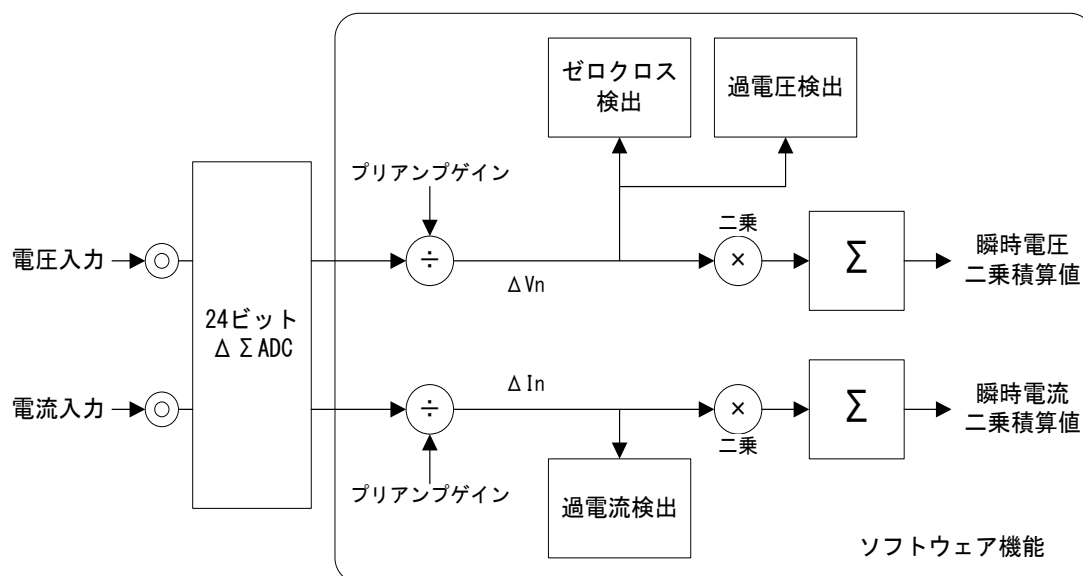


図 10-2 皮相電力測定ブロック図

皮相電力 S を求めるには、まず実効電圧 V と実効電流 I を計算する必要があります。

RL78/I1B の $\Delta\Sigma$ ADC でサンプリングされた瞬間電圧 ΔV_n の二乗値をサンプリング毎に積算していきます。測定時間が経過しサンプリングが終了した後、積算された瞬間電圧 ΔV_n の二乗値をサンプリング数 n で割って平方根をとることで実効電圧 V が計算されます。

$$\text{実効電圧 } V = \sqrt{\frac{\Delta V_1^2 + \Delta V_2^2 + \dots + \Delta V_n^2}{n}}$$

実効電流 I も実効電圧 V と同様の方法を使います。RL78/I1B の $\Delta\Sigma$ ADC でサンプリングされた瞬間電流 ΔI_n の二乗値をサンプリング毎に積算していきます。測定時間が経過しサンプリングが終了した後、積算された瞬間電流 ΔI_n の二乗値をサンプリング数 n で割って平方根をとることで実効電流 I が計算されます。

$$\text{実効電流 } I = \sqrt{\frac{\Delta I_1^2 + \Delta I_2^2 + \dots + \Delta I_n^2}{n}}$$

皮相電力 S は計算した実効電圧 V と実効電流 I を掛け合わせて計算します。

$$\text{皮相電力 } S = \text{実効電圧 } V \times \text{実効電流 } I$$

EV-200-USB-3 (RL78/I1B 評価キット)	2016/08/01	SJAA-150010-01	53/54
	アプリケーションノート 基本編		

3) 無効電力

無効電力 Q は有効電力 P と皮相電力 S をもとに以下の式で計算します。

$$\text{無効電力 } Q = \sqrt{\text{皮相電力 } S^2 - \text{有効電力 } P^2}$$

	2016/08/01	SJAA-150010-01	54/54
EV-200-USB-3 (RL78/I1B 評価キット)	アプリケーションノート 基本編		

11. 改版履歴

版数	日付	内容
1 版	2016/8/1	新規作成