

## RL78 ファミリ

### SIS を用いた PMBus 通信サンプルソフトウェア

---

#### 要旨

本アプリケーションノートは、PMBus Master モジュール SIS (Software Integration System)、及び PMBus Slave モジュール SIS (Software Integration System) を用いて PMBus 通信を行うサンプルアプリケーションについて説明します。対応規格は以下のとおりです。

- ・ PMBus Specification Rev. 1.4 Part I
- ・ PMBus Specification Rev. 1.4 Part II
- ・ System Management Bus (SMBus) Specification Version 3.2

PMBus 通信規格の詳細については上記の各規格書を参照してください。

#### 動作確認デバイス

RL78/G24

## 目次

1. 仕様概要 .....	4
2. 動作確認条件 .....	5
3. ハードウェア説明 .....	6
3.1 システム構成 .....	6
3.2 使用端子一覧 .....	7
4. ソフトウェア説明 .....	8
4.1 環境構築 .....	8
4.1.1 ソフトウェアインストール .....	8
4.1.2 ファームウェアの書き込み .....	8
4.1.3 サンプルアプリケーションのハードウェア接続方法 .....	9
4.2 動作概要 .....	11
4.2.1 通信フォーマット仕様 .....	12
4.2.1.1 Write Byte/Word .....	12
4.2.1.2 Read Byte/Word .....	13
4.2.1.3 Host Notify .....	13
4.2.1.4 その他の通信フォーマット .....	13
4.2.2 PMBus コマンド仕様 .....	14
4.2.2.1 PAGE .....	14
4.2.2.2 VOUT_COMMAND .....	14
4.2.2.3 READ_VOUT .....	14
4.2.3 アプリケーション動作仕様 .....	15
4.2.3.1 LED 調光動作 .....	15
4.2.3.2 LED 出力値の読み出し .....	15
4.2.3.3 Control Signal .....	15
4.2.3.4 Write Protect .....	16
4.2.3.5 PMBus 通信ログ .....	16
4.3 動作手順 .....	17
4.3.1 共通手順 .....	17
4.3.2 Write 通信による LED 調光動作確認 .....	17
4.3.3 Read 通信による LED 現在出力値の確認 .....	17
4.3.4 Control Signal 動作の確認 .....	18
4.3.5 Write Protect 動作の確認 .....	18
4.4 機能概要(PMBus Master) .....	19
4.4.1 機能ブロック図 .....	19
4.4.2 SIS (Software Integration System) .....	20
4.4.2.1 PMBMDRV (PMBus Master Driver) .....	20
4.4.2.2 PMBMCTL (PMBus Master Controller) .....	20
4.4.3 アプリケーション層 .....	21
4.4.3.1 PMBMAPP (PMBus Master Application) .....	21
4.5 機能概要(PMBus Slave) .....	22
4.5.1 機能ブロック図 .....	22

## RL78 ファミリ SIS を用いた PMBus 通信サンプルソフトウェア

---

4.5.2	ドライバ層.....	23
4.5.2.1	調光ドライバ.....	23
4.5.3	SIS (Software Integration System).....	23
4.5.3.1	PMBSDRV (PMBus Slave Driver).....	23
4.5.3.2	PMBSCTL (PMBus Slave Controller).....	23
4.5.4	アプリケーション層.....	24
4.5.4.1	PMBSAPP (PMBus Slave Application).....	24
4.6	ソフトウェア構成.....	26
4.6.1	フォルダ構成.....	26
4.6.2	オプション・バイトの設定一覧.....	28
5.	注意事項.....	29
6.	参考ドキュメント.....	29
	改訂記録.....	30

## 1. 仕様概要

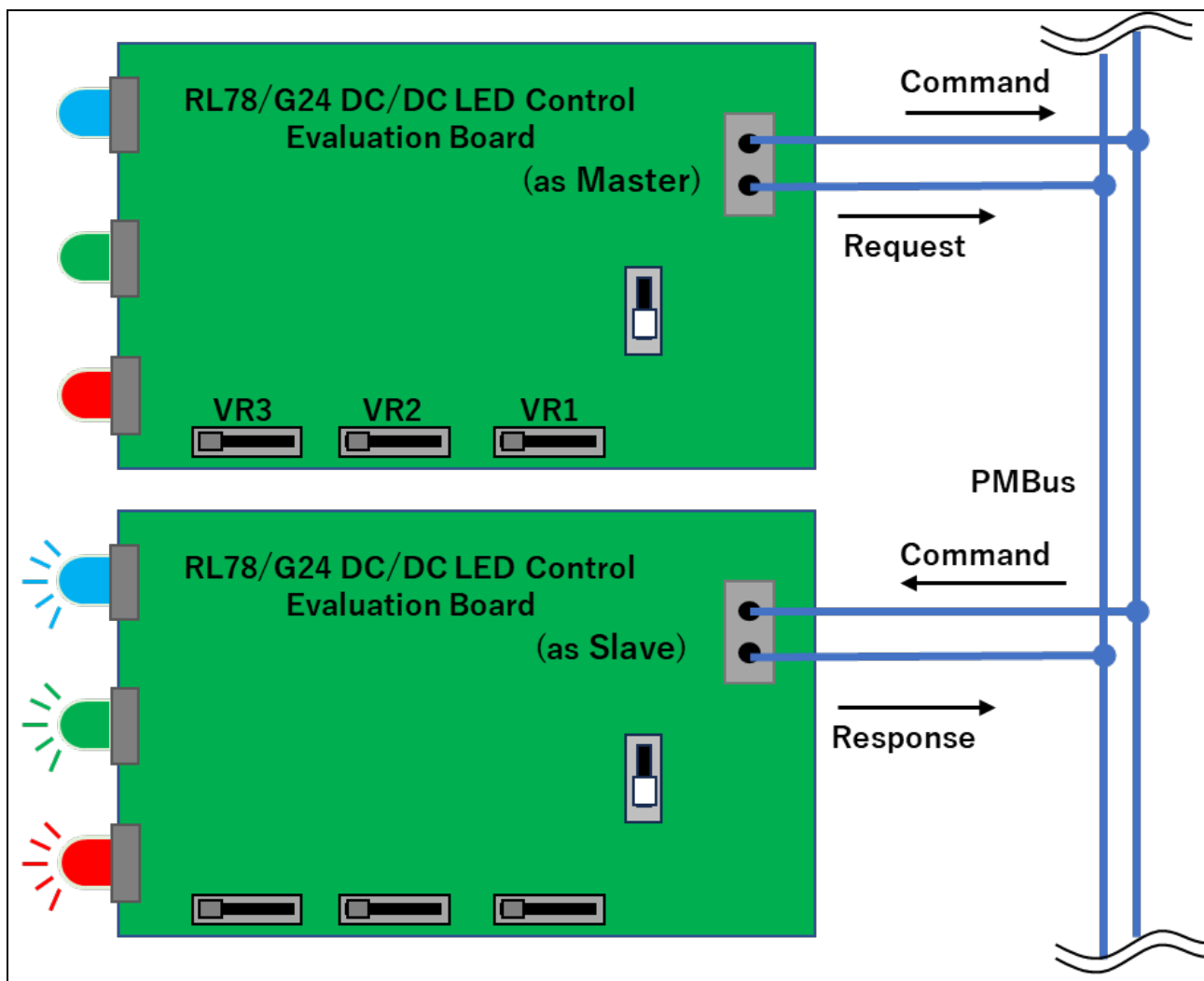
本アプリケーションノートは、PMBus 通信におけるマスタ機能、及びスレーブ機能を実現するサンプルアプリケーションについて記載しています。PMBus (Power Management Bus)とは電源管理を目的としたオープンな標準プロトコルであり、電力装置との通信手段が定義されています。

サンプルアプリケーション内の PMBus 通信プロトコルスタックには PMBus Master モジュール SIS 、及び PMBus Slave モジュール SIS が使用されています。PMBus Master モジュール SIS 、及び PMBus Slave モジュール SIS の詳細情報については以下のドキュメントを参照してください。

- RL78 ファミリ PMBus Master モジュール Software Integration System (R01AN7188)
- RL78 ファミリ PMBus Slave モジュール Software Integration System (R01AN6985)

以下にサンプルアプリケーションの動作概要図を示します。サンプルアプリケーションは RL78/G24 DC/DC LED Control Evaluation Board にて動作します。スレーブボード上の 3 つの LED を電源ユニットに見立てます。マスタボード上のボリュームスイッチ (VR1, VR2, VR3) に連動し PMBus コマンドを送信します。マスタボードから受信したコマンドおよびデータに応じて、スレーブボード上の LED の調光を行います。

図 1-1 PMBus サンプルアプリケーション動作概要図



## 2. 動作確認条件

サンプルアプリケーションは以下の環境で動作確認をしております。

表 2.1 動作確認環境

項目	内容
使用マイコン	RL78/G24 (R7F101GLG2DFB)
使用ボード	ルネサスエレクトロニクス製 RL78/G24 DC/DC LED Control Evaluation Board
動作周波数	<ul style="list-style-type: none"><li>• 高速オンチップ・オシレータ・クロック (f<sub>HOCO</sub>) : 8MHz</li><li>• PLL クロック (f<sub>PLL</sub>) : 96MHz</li><li>• CPU/周辺ハードウェア・クロック (f<sub>CLK</sub>) : 48MHz</li></ul>
動作電圧	5.0V
統合開発環境 (CS+)	ルネサスエレクトロニクス製 CS+ for CC 8.10.00
C コンパイラ (CS+)	ルネサスエレクトロニクス製 CC-RL V1.12.01
統合開発環境 (IAR)	IAR Systems 製 IAR Embedded Workbench IDE V5.10.3
C コンパイラ (IAR)	IAR Systems 製 IAR C/C++ Compiler for Renesas RL78 V5.10.3
スマート・コンフィグレータ (SC)	V1.8.0
エミュレータ	E2 エミュレータ Lite

### 3. ハードウェア説明

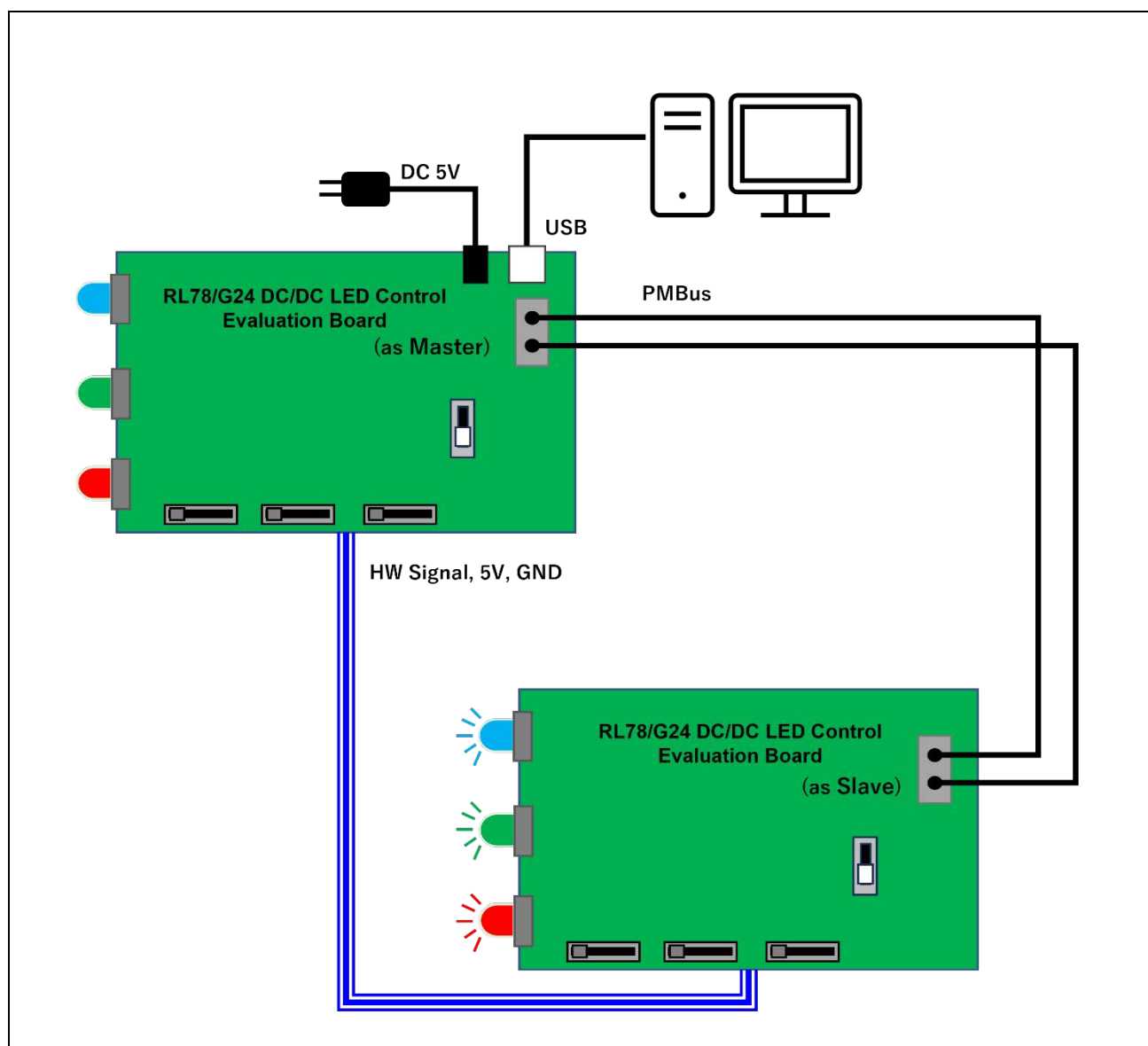
#### 3.1 システム構成

システムの構成を以下に示します。

下図のようにマスタ(コントローラ)デバイスおよびスレーブデバイスを接続します。マスタおよびスレーブデバイスは RL78/G24 DC/DC LED Control Evaluation Board にて動作します。

マスタデバイスと PC を USB ケーブルにて接続することで、マスタとスレーブ間の PMBus 通信ログを PC 画面上に表示することができます。

図 3-1 システム構成図



### 3.2 使用端子一覧

サンプルアプリケーションで使用するスレーブデバイスの端子と機能を以下に示します。

表 3.1 使用端子と機能(マスタ)

端子名	入出力	内容
P60 / SCLA0	入出力	I <sup>2</sup> C シリアル・クロック
P61 / SDAA0	入出力	I <sup>2</sup> C シリアル・データ・バス
P10	出力	Control Signal 出力
P140	入力	Control Signal 値切り替え(SW2)

表 3.2 使用端子と機能(スレーブ)

端子名	入出力	内容
P60 / SCLA0	入出力	I <sup>2</sup> C シリアル・クロック
P61 / SDAA0	入出力	I <sup>2</sup> C シリアル・データ・バス
P10	入力	Control Signal 入力
P140	入力	Write Protect 入力

※本サンプルでは使用されませんが、PMBus Slave モジュール SIS では、SMBALERT#端子によるアラート処理もサポートしております。詳細については下記ドキュメントを参照してください。

- RL78 ファミリ PMBus Master モジュール Software Integration System (R01AN7188)
- RL78 ファミリ PMBus Slave モジュール Software Integration System (R01AN6985)

## 4. ソフトウェア説明

### 4.1 環境構築

#### 4.1.1 ソフトウェアインストール

ファームウェア書き込みを行うために、フラッシュ書き込みソフト Renesas Flash Programmer (以降、RFP と略します)が必要となります。ルネサスエレクトロニクスのホームページからダウンロードしてください。

RFP の基本操作については下記のユーザーズマニュアルを参照してください。

- Renesas Flash Programmer V3.13 フラッシュ書き込みソフトウェア ユーザーズマニュアル (R20UT5352)

#### 4.1.2 ファームウェアの書き込み

マスタ、スレーブそれぞれのボードに対象のファームウェアを書き込みます。プロジェクトフォルダ内にモトローラ・Sタイプ・フォーマット形式のファイルが格納されておりますので、RFP を使用してフラッシュ書き込みを行ってください。格納フォルダについては、4.6.1 フォルダ構成よりご確認ください。

マスタ : PMBusMaster\_Sample.mot

スレーブ : PMBusSlave\_Sample.mot

RL78/G24 DC/DC LED Control Evaluation Board にファームウェアを書き込む際は接続方法に応じ、ボード上の SW1 の設定を行う必要があります。

図 4-1 RL78/G24 DC/DC LED Control Evaluation Board SW1

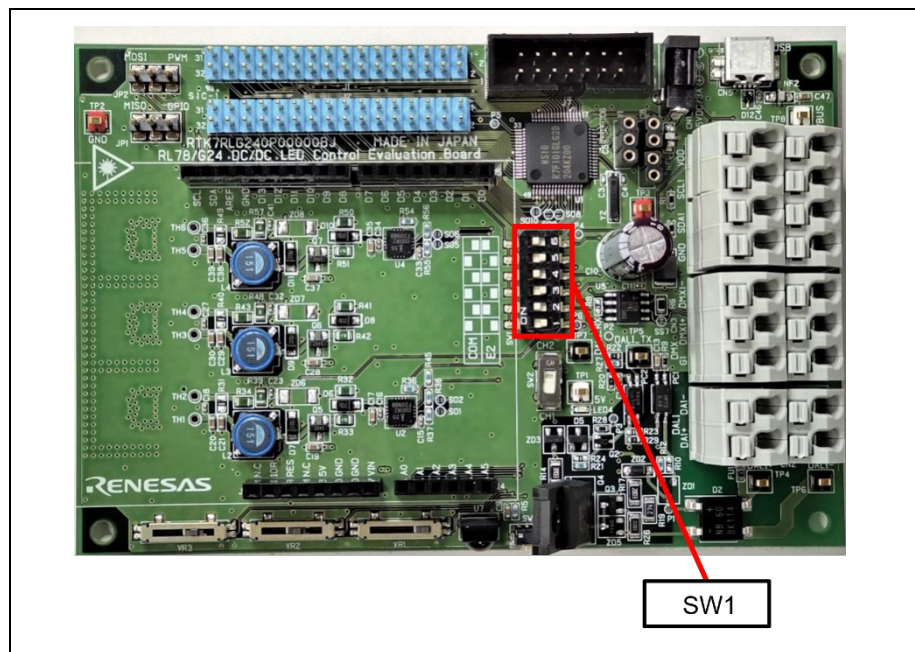




表 4.1 RL78/G24 DC/DC LED Control Evaluation Board SW1 設定

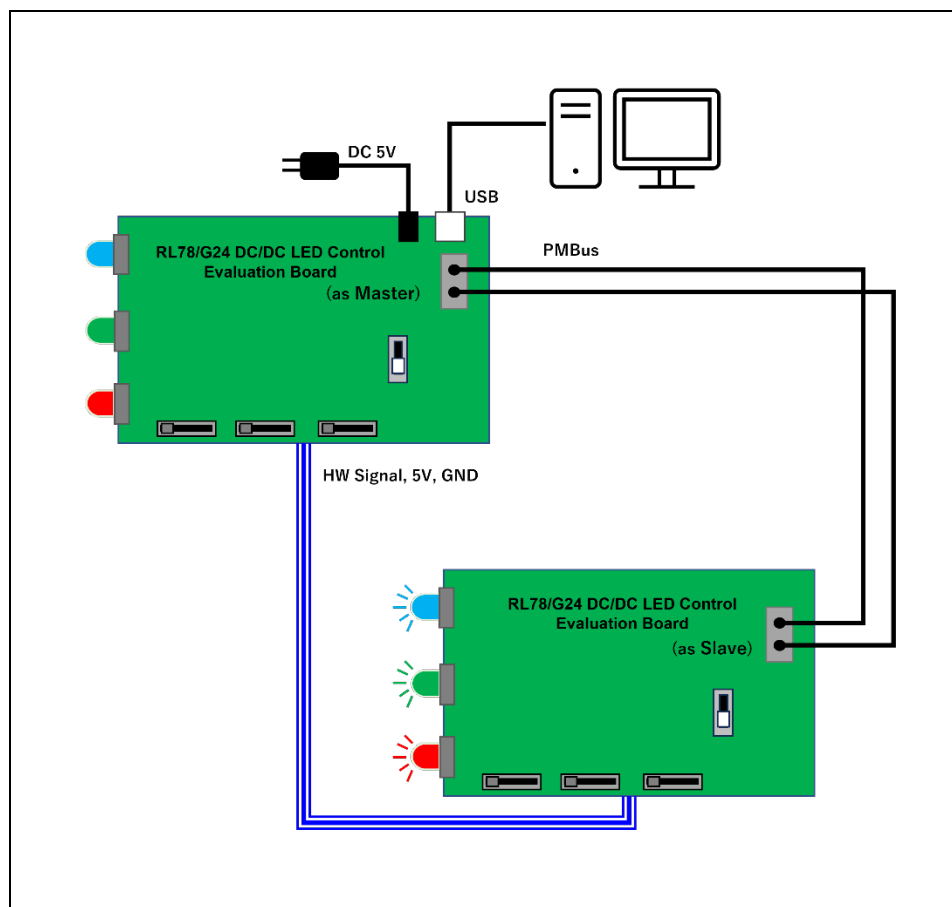
SW1	COM port	E2
1	ON	OFF
2	OFF	ON
3	ON	OFF
4	OFF	ON
5	ON	OFF
6	OFF	ON
7	ON	OFF
8	OFF	ON

#### 4.1.3 サンプルアプリケーションのハードウェア接続方法

ハードウェア接続方法について説明します。マスターボード側またはスレーブボード側のいずれかに AC アダプタを接続し、電源供給を行ってください。また、マスターボード上の USB 端子と PC を接続することで、PMBus の通信ログをモニタすることができます。

マスターボードとスレーブボードの接続方法については次頁にて説明します。

図 4-2 ハードウェア構成図



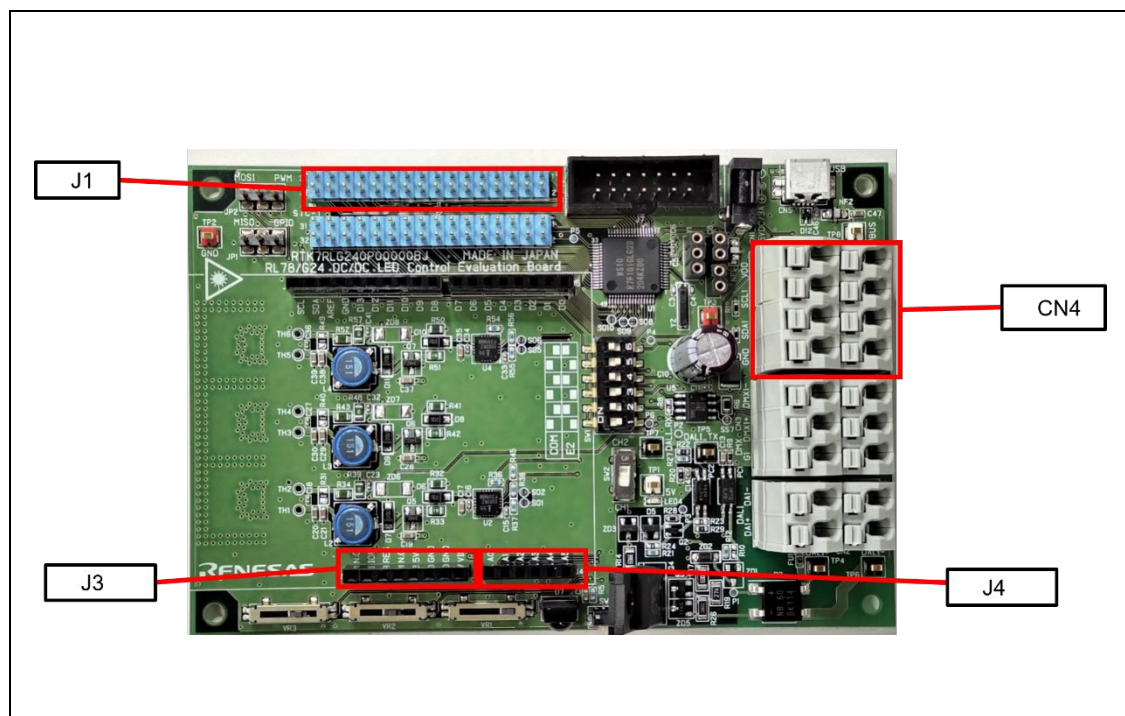
## RL78 ファミリ SIS を用いた PMBus 通信サンプルソフトウェア

マスタボードとスレーブボードの接続内容を以下に記載します。

表 4.2 マスタ/スレーブボード接続

項目	マスタボード	スレーブボード
SMBCLK	CN4-5 (SCLA0)	CN4-5 (SCLA0)
SMBDAT	CN4-3 (SDAA0)	CN4-3 (SDAA0)
Control	J4-6 (P10)	J4-6 (P10)
DC5V	J3-5	J3-5
GND	J3-7	J3-7

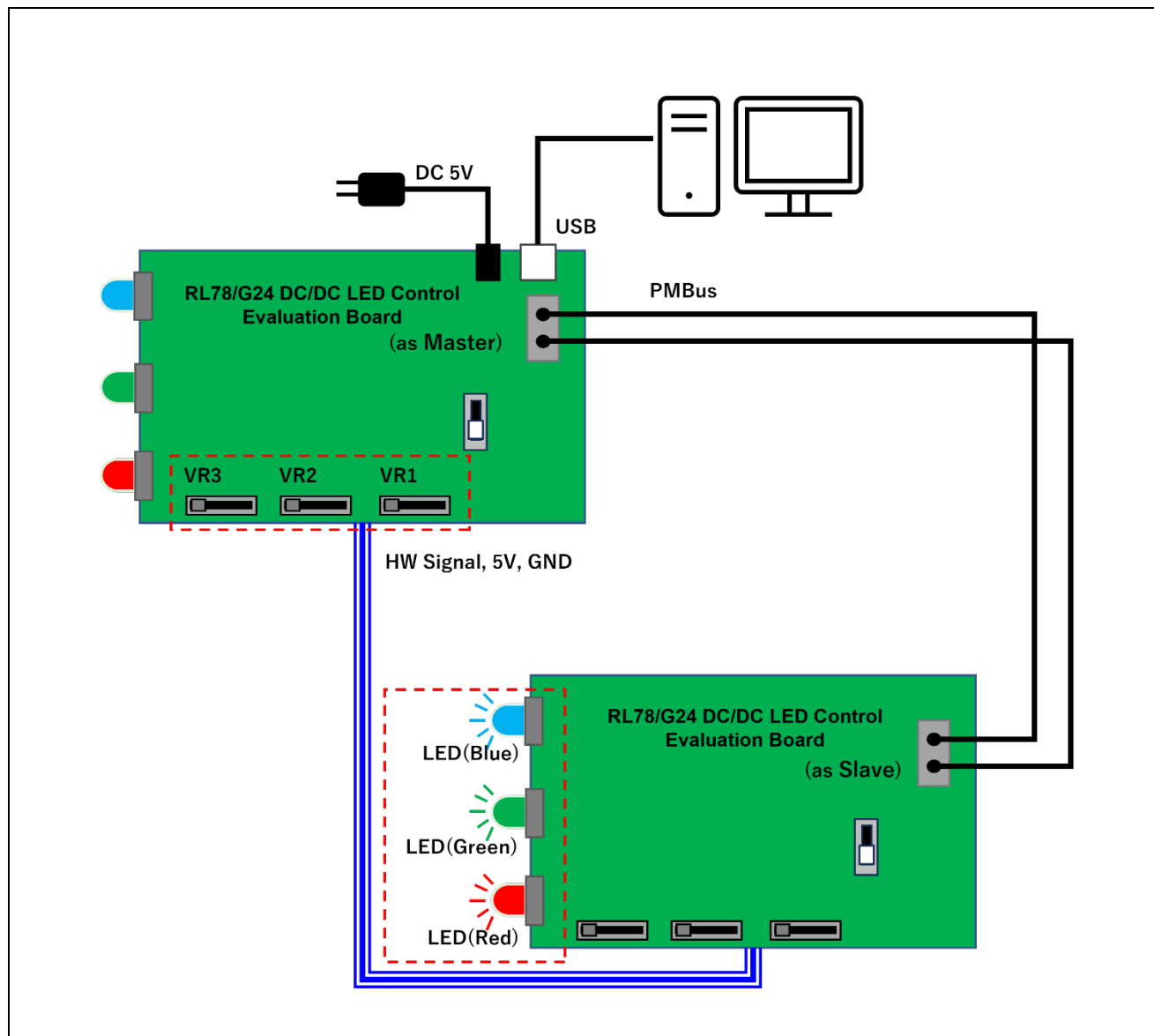
図 4-3 RL78/G24 DC/DC LED Control Evaluation Board 上の各端子



## 4.2 動作概要

本サンプルアプリケーションの動作概要を記載します。

図 4-4 システム構成



マスタボード上のボリュームスイッチ(VR1, VR2, VR3)に連動し、スレーブボード上の3色のLED(赤、緑、青)の調光レベルが切り替わります。マスタはボリュームスイッチレベルに変化があればスレーブに対しPMBusコマンドを送信し、LED出力レベル(目標値)の書き込みを行います。これを受けて、スレーブはLEDの出力レベルを切り替えます。

また、マスタはスレーブのLED出力レベル(実際の出力値)を知るために、定期的にコマンドを送信し、スレーブからデータの読み出しを行います。

PCとマスタボードをUSBケーブルで接続することで、これらのコマンド通信ログをPC画面に表示させることができます。

#### 4.2.1 通信フォーマット仕様

本サンプルアプリケーションにて使用する PMBus 通信フォーマットについて説明します。

凡例：

S : スタートコンディション

Sr : リピートスタートコンディション

Rd : リード(1)

Wr : ライト(0)

ACK : アクノリッジ

NACK : Not アクノリッジ

PEC : Packet Error Code

P : ストップコンディション

: マスタ (or ホスト) → スレーブ

: スレーブ → マスタ (or ホスト)

##### 4.2.1.1 Write Byte/Word

任意のターゲットアドレスに対し、コマンドコードとそれに対応する書き込みデータを送信します。

図 4-5 通信フォーマット Write (Single byte)

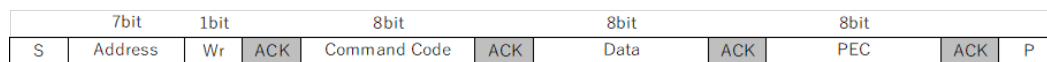
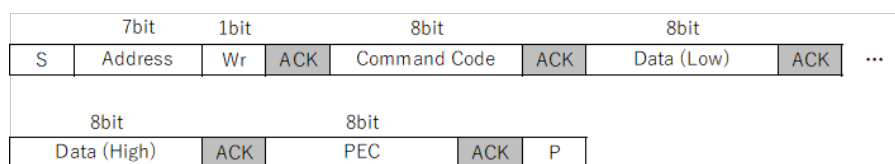


図 4-6 通信フォーマット Write (Multi byte)



#### 4.2.1.2 Read Byte/Word

任意のターゲットアドレスに対し、コマンドコードを送信します。次にターゲットデバイスからコマンドコードに対応する応答データを読み出します。

図 4-7 通信フォーマット Read (Single byte)

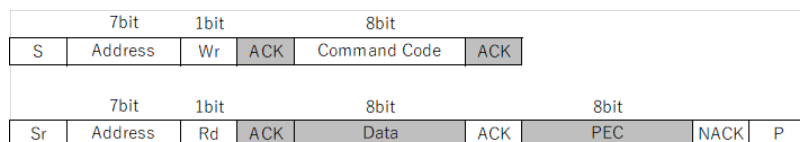
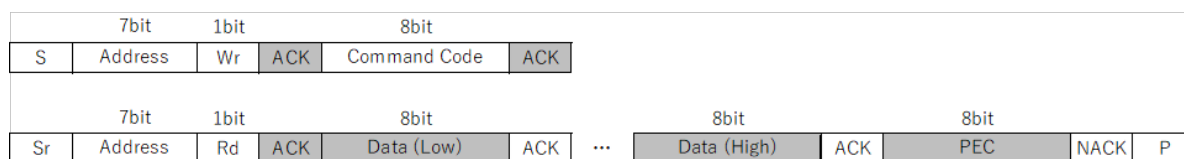


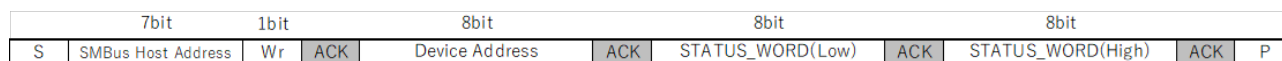
図 4-8 通信フォーマット Read (Multi byte)



#### 4.2.1.3 Host Notify

スレーブデバイスは通信異常やデバイスの故障を検出した際に、自身のアドレス情報とステータス情報 (STATUS\_WORD) を下記の Host Notify 通信フォーマットにて送信します。

図 4-9 通信フォーマット Host Notify



#### 4.2.1.4 その他の通信フォーマット

本サンプルでは使用されませんが、PMBus Slave モジュール SIS では、上記以外の通信フォーマットによるコマンド受信もサポートしております。詳細については下記ドキュメントを参照してください。

- RL78 ファミリ PMBus Master モジュール Software Integration System (R01AN7188)
- RL78 ファミリ PMBus Slave モジュール Software Integration System (R01AN6985)

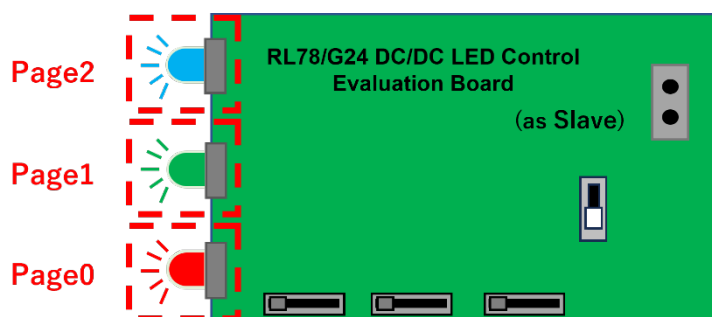
#### 4.2.2 PMBus コマンド仕様

本サンプルアプリケーションにて使用する PMBus コマンドについて説明します。

##### 4.2.2.1 PAGE

PAGE コマンドは 1 つのデバイス内で複数の出力を割り当てるための機能を提供します。本サンプルアプリケーションではスレーブボード上の各 LED にページの割り当てを行い、PAGE コマンドにて制御対象の LED の切り替えを行います。コマンドは前述の Write Byte/Word 通信フォーマットにて送信され、1byte のデータ(ページ値)が書き込まれます。

図 4-10 Page 割り当て



##### 4.2.2.2 VOUT\_COMMAND

VOUT\_COMMAND は、デバイスの出力電圧を指定された値に設定するためのコマンドです。本サンプルアプリケーションでは、制御対象の LED の出力レベルの設定を行います。コマンドは前述の Write Byte/Word 通信フォーマットにて送信され、2byte のデータ(出力目標値)が書き込まれます。

##### 4.2.2.3 READ\_VOUT

READ\_VOUT、実際に測定された出力電圧を応答するためのコマンドです。本サンプルアプリケーションでは制御対象の LED の実際の出力レベルを応答します。コマンドは前述の Read Byte/Word 通信フォーマットにて送信され、スレーブより 2byte のデータ(現在の出力レベル値)を読み出します。

#### 4.2.3 アプリケーション動作仕様

本サンプルアプリケーションの動作仕様について説明します。

##### 4.2.3.1 LED 調光動作

マスタボード上のボリュームスイッチ(VR1, VR2, VR3) 入力値に変化があった場合、マスタからスレーブへ下記コマンドを送信し、制御対象の LED の切り替えおよび LED 出力指示を行います。

- PAGE
- VOUT\_COMMAND

スレーブは受信した PAGE コマンドより対象の LED を決定し、VOUT\_COMMAND にて設定された出力値を制御対象の LED へ反映します。

##### 4.2.3.2 LED 出力値の読み出し

マスタは一定周期(5 秒間隔)で下記コマンドをスレーブへ送信し、LED の実際の出力値を取得します。

- PAGE
- READ\_VOUT

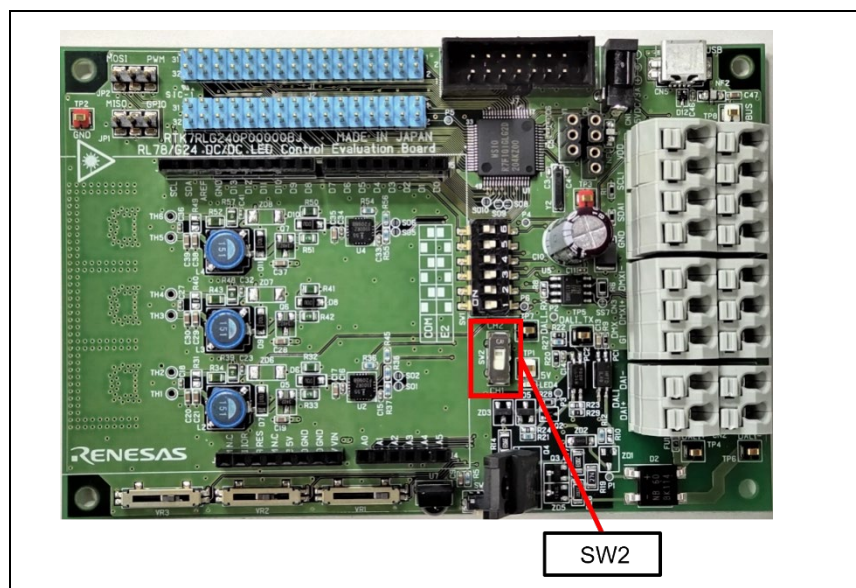
スレーブは受信した PAGE コマンドより対象の LED を決定し、READ\_VOUT コマンドに LED の現在出力値を応答します。

##### 4.2.3.3 Control Signal

マスタボード上の SW2 を ON/OFF することで、マスタ/スレーブ間で接続されている Control Signal の出力レベルが切り替わります。

スレーブは Control Signal が OFF のとき、LED を消灯します。

図 4-11 RL78/G24 DC/DC LED Control Evaluation Board SW2



#### 4.2.3.4 Write Protect

スレーブボード上の SW2 を ON/OFF することで、Write Protect 信号が切り替わります。Write Protect 信号が ON のときはコマンドによる内部情報の書き換えが禁止され、PAGE や VOUT\_COMMAND を受信してもこれを反映しません。また、内部情報の書き換え禁止中に PAGE や VOUT\_COMMAND などの Write コマンドを受信した場合は、通信異常として前述した Host Notify がスレーブから送信されます。

#### 4.2.3.5 PMBus 通信ログ

マスタボードと PC を接続することで、PMBus 通信ログをモニタすることができます。

シリアル通信ソフトウェア (Tera Term 等) を使用し、RL78/G24 DC/DC LED Control Evaluation Board と接続されている USB の COM ポートを選択し接続してください。接続設定は以下となります。

表 4.3 シリアル通信接続設定

項目	設定内容
通信ボーレート	9600
データビット	8bit
パリティ	none
ストップビット	1bit
フロー制御	none

通信ログの表示フォーマットについて説明します。

##### (1) Write

Write 通信フォーマットのコマンドを送信したとき、下記の通信ログを出力します。送信元、送信先のアドレス情報やコマンドコード、書き込みデータ、PEC Byte 情報が表示されます。

```
Master( 0x** ) -> Slave( 0x** ): Write [ Command: 0x**, Data: 0x** 0x**, PEC: 0x** ]
```

##### (2) Read

Read 通信フォーマットのコマンドを送信したとき、下記の通信ログを出力します。送信元、送信先のアドレス情報やコマンドコード、応答データ、PEC Byte 情報が表示されます。

```
Master( 0x** ) -> Slave( 0x** ): Read [ Command: 0x**, Response: 0x** 0x**, PEC: 0x** ]
```

##### (3) Host Notify

Host Notify メッセージを検出したとき、下記の通信ログを出力します。送信元、送信先のアドレス情報や送信データ (STATUS\_WORD 情報) が表示されます。

```
Slave( 0x** ) -> Host( 0x** ) : Host Notify [ Data: 0x** 0x** 0x** ]
```



### 4.3 動作手順

本アプリケーションの動作手順例を記載します。

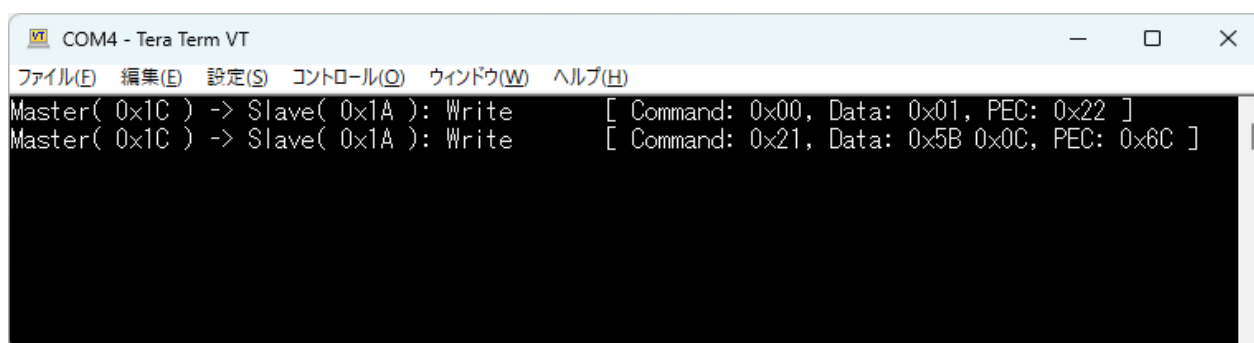
#### 4.3.1 共通手順

1. マスタボード上の SW2 を ON(CH2 側)としてください。
2. スレーブボード上の SW2 を OFF(CH1 側)としてください。
3. 4.1 環境構築に従い、各ボードの接続や電源供給を行います。
4. 4.2.3.5 PMBus 通信ログに従い、マスタボードと PC をシリアル接続します。

#### 4.3.2 Write 通信による LED 調光動作確認

1. マスタボード上のボリューム SW (VR1, VR2, VR3)を任意のレベルに設定してください。
2. SW 操作に連動してスレーブボード上の LED の光量に変化することを確認します。
3. さらに、シリアル通信画面にて、PAGE コマンド、VOUT\_COMMAND コマンドが送信されていることを確認します。

図 4-12 Write 通信ログ

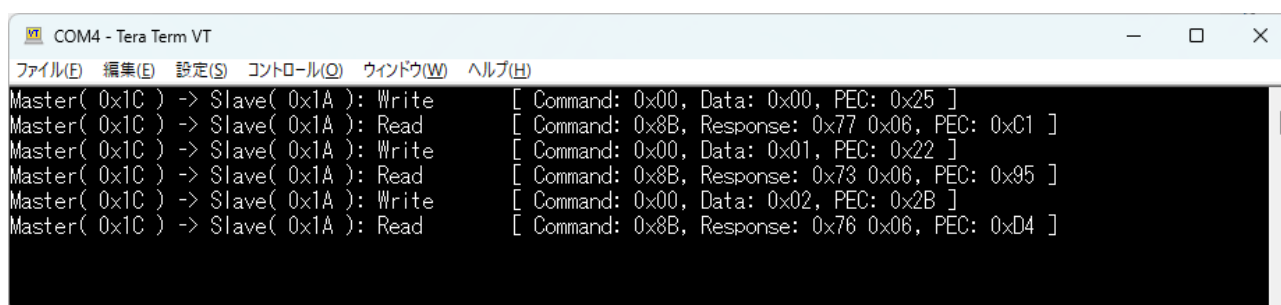


```
COM4 - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
Master( 0x1C ) -> Slave( 0x1A ): Write [ Command: 0x00, Data: 0x01, PEC: 0x22 ]
Master( 0x1C ) -> Slave( 0x1A ): Write [ Command: 0x21, Data: 0x5B 0x0C, PEC: 0x6C ]
```

#### 4.3.3 Read 通信による LED 現在出力値の確認

1. 5s 間隔で PAGE コマンド、READ\_VOUT コマンドが定期送信されていることを、シリアル通信画面にて確認します。
2. シリアル通信画面の Read 通信において、LED の現在の出力レベル値が確認できます。
3. マスタボード上のボリューム SW (VR1, VR2, VR3)にて LED の出力レベル(目標値)を変化させると、LED の現在出力値もそれに応じて変化します。

図 4-13 Read 通信ログ



```
COM4 - Tera Term VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
Master( 0x1C ) -> Slave( 0x1A ): Write [ Command: 0x00, Data: 0x00, PEC: 0x25 ]
Master( 0x1C ) -> Slave( 0x1A ): Read [ Command: 0x8B, Response: 0x77 0x06, PEC: 0xC1 ]
Master( 0x1C ) -> Slave( 0x1A ): Write [ Command: 0x00, Data: 0x01, PEC: 0x22 ]
Master( 0x1C ) -> Slave( 0x1A ): Read [ Command: 0x8B, Response: 0x73 0x06, PEC: 0x95 ]
Master( 0x1C ) -> Slave( 0x1A ): Write [ Command: 0x00, Data: 0x02, PEC: 0x2B ]
Master( 0x1C ) -> Slave( 0x1A ): Read [ Command: 0x8B, Response: 0x76 0x06, PEC: 0xD4 ]
```

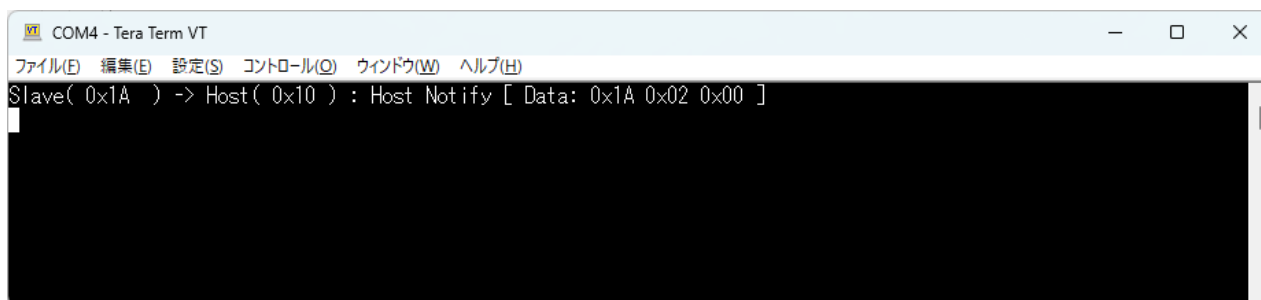
#### 4.3.4 Control Signal 動作の確認

1. マスタボード上のボリューム SW (VR1, VR2, VR3)を最大レベルとし、スレーブボード上の LED が点灯していることを確認します。
2. マスタボード上の SW2 を OFF すると、マスタ/スレーブ間で接続された Control Signal が非アクティブとなります。このとき、スレーブボード上の LED が消灯することを確認します。
3. マスタボード上の SW2 を ON すると、スレーブボード上の LED が再び点灯することを確認します。

#### 4.3.5 Write Protect 動作の確認

1. スレーブボード上の SW2 を ON してください。ボード上で接続された Write Protect 信号がアクティブとなります。
2. マスタボード上のボリューム SW (VR1, VR2, VR3)を変化させても、スレーブボード上の LED の光量が変わらないことを確認してください。
3. また、通信異常としてスレーブボードから Host Notify が発行されていることをシリアル通信画面にて確認してください。

図 4-14 Host Notify 通信ログ

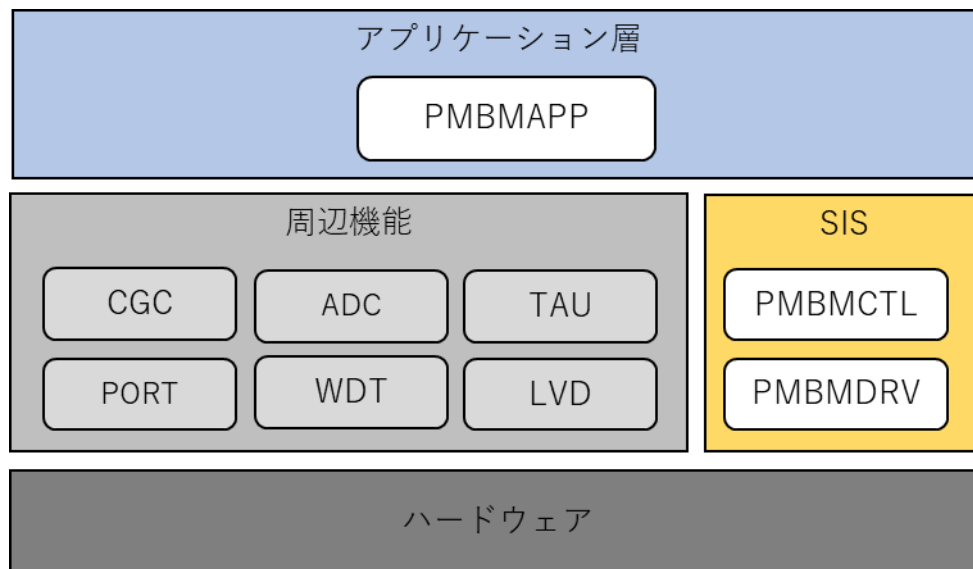


## 4.4 機能概要(PMBus Master)

### 4.4.1 機能ブロック図

サンプルアプリケーションの機能ブロック図を以下に示します。

図 4-15 機能ブロック図



## RL78 ファミリ SIS を用いた PMBus 通信サンプルソフトウェア

### 4.4.2 SIS (Software Integration System)

#### 4.4.2.1 PMBMDRV (PMBus Master Driver)

PMBus Master モジュール SIS のドライバ部です。シリアル・インターフェース IICA を利用して PMBus のマスタ送受信動作を実現します。

モジュールの詳細情報については下記ドキュメントを参照してください。

- RL78 ファミリ PMBus Master モジュール Software Integration System (R01AN7188)

#### 4.4.2.2 PMBMCTL (PMBus Master Controller)

PMBus Master モジュール SIS のミドルウェア部です。アプリケーションからコマンドおよびデータ送信要求を受け付けます。

モジュールの詳細情報については下記ドキュメントを参照してください。

- RL78 ファミリ PMBus Master モジュール Software Integration System (R01AN7188)

本サンプルアプリケーションでのコンフィグレーション設定内容を以下に示します。

表 4-4 PMBus Master Driver 設定内容一覧

項目	設定値	説明
Bus Speed	100kHz	バス速度を指定します。
IICA Input Mode	SMBus	IICA 入力モードを指定します。
SDAA and SCLA signal falling times (tF)[ns]	0	SDAA および SCLA 信号の立ち下がり時間を設定します。(注1)
SDAA and SCLA signal rising times (tR)[ns]	0	SDAA および SCLA 信号の立ち上がり時間を設定します。(注1)
Digital filter	OFF	デジタル・フィルタを有効にするかどうかを指定します。(注2)
Interrupt level for INTIICA0	Level 3(Lowest)	INTIICA0 割り込みの優先度を設定します。
Pin for Control Signal	P10	Control Signal 端子を選択します。
Pin for SMBALERT#	Unused	SMBALERT#端子を選択します。
Timer resource for device timeout measurement	TAU0_0	デバイスタイムアウトを計測するためのタイマリソースを選択します。
Host Notify Supported	Supported	SMBus Host Notify をサポートするかどうかを選択します。

表 4-5 PMBus Master Controller 設定内容一覧

項目	設定値	説明
Support the SMBus Packet Error Checking (PEC)	Supported	PEC をサポートするかどうかを選択します。

#### 4.4.3 アプリケーション層

##### 4.4.3.1 PMBMAPP (PMBus Master Application)

PMBus 通信機能におけるアプリケーション部です。一定周期でスレーブにコマンド送信をし LED の出力値を取得します。ボリュームスイッチの入力値に変化があった場合、スレーブにコマンド送信をし LED 出力レベルを指示します。また、PMBus 通信ログをシリアル通信画面に出力します。

#### (1) アプリケーションタスク処理

##### (a) LED 調光動作

LED 調光動作を 1000ms 間隔で実施します。ボリュームスイッチ(VR1, VR2, VR3)の値を取得し値に変化があれば、PMBMCTL より提供される API 関数"RM\_PMBMCTL\_WriteData"にてコマンドとページ値を指定し PAGE コマンドを送信し、PMBus 通信ログをシリアル画面に出力します。次に API 関数"RM\_PMBMCTL\_WriteData"にてコマンドと LED 出力レベルを指定し VOUT\_COMMAND コマンドを送信し、PMBus 通信ログをシリアル画面に出力します。

##### (b) LED 出力値の読み出し

LED 出力値の読み出しを 5000ms 間隔で実施します。PMBMCTL より提供される API 関数"RM\_PMBMCTL\_WriteData"にてコマンドとページ値を指定し PAGE コマンドを送信し、PMBus 通信ログをシリアル画面に出力します。次に API 関数"RM\_PMBMCTL\_ReadData"にてコマンドを指定し READ\_VOUT コマンドを送信し LED の出力値を取得します。PMBus 通信ログをシリアル通信画面に出力します。

##### (c) Control Signal の設定

Control Signal の設定を 100ms 間隔で実施します。SW2 の状態を取得しポート Hi レベルの場合はアクティブを、ポート Low レベルの場合は非アクティブを PMBMCTL より提供される API 関数"RM\_PMBMCTL\_SetControl"にて指定し Control Signal を設定します。

#### (2) PMBus Master モジュールからのコールバック時の処理

##### (a) HostNotifyReceived コールバック

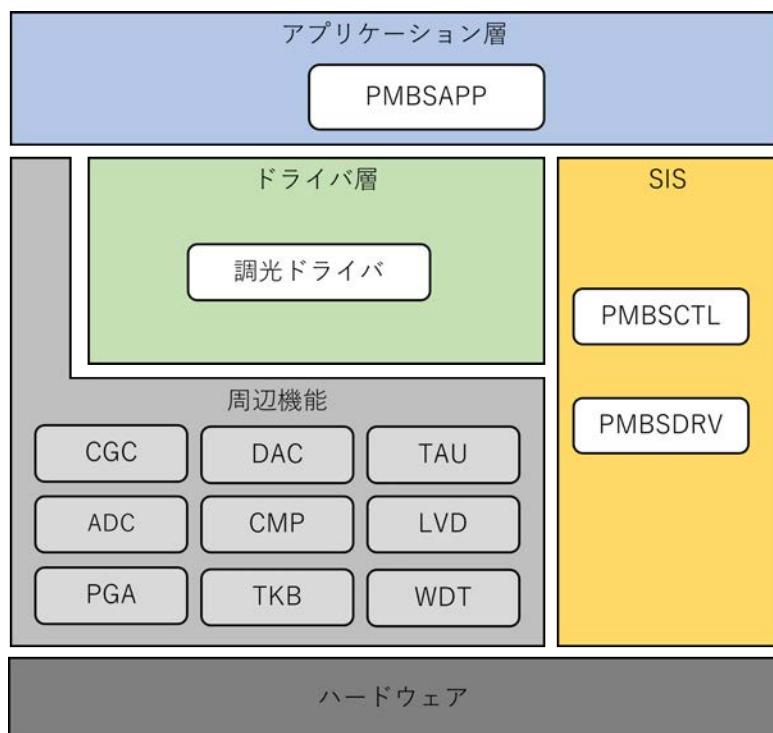
Host Notify を受信したタイミングで PMBMCTL よりコールバックされます。通知された STATUS\_WORD 情報を取得し PMBus 通信ログをシリアル画面に出力します。

## 4.5 機能概要(PMBus Slave)

### 4.5.1 機能ブロック図

サンプルアプリケーションの機能ブロック図を以下に示します。

図 4-16 機能ブロック図



## 4.5.2 ドライバ層

### 4.5.2.1 調光ドライバ

3色のLED(赤色、緑色、青色)を対象に調光を行うドライバです。

下記周辺機能を使用して各LEDの調光制御を行います。

- ・ADC
- ・PGA
- ・DAC
- ・CMP
- ・TKB

調光制御の詳細に関しては、下記ドキュメントを参照してください。

- RL78/G24 による LED 制御 (R01AN6673)

## 4.5.3 SIS (Software Integration System)

### 4.5.3.1 PMBSDRV (PMBus Slave Driver)

PMBus Slave モジュール SIS のドライバ部です。シリアル・インターフェース IICA を利用して PMBus のスレーブ送受信動作を実現します。

モジュールの詳細情報については下記ドキュメントを参照してください。

- RL78 ファミリ PMBus Slave モジュール Software Integration System (R01AN6985)

### 4.5.3.2 PMBSCTL (PMBus Slave Controller)

PMBus Slave モジュール SIS のミドルウェア部です。PMBus 通信仕様に基いたパケットデータの構築や通信フォーマットの解釈を行います。

モジュールの詳細情報については下記ドキュメントを参照してください。

- RL78 ファミリ PMBus Slave モジュール Software Integration System (R01AN6985)

本サンプルアプリケーションでのコンフィグレーション設定内容を以下に示します。

表 4-6 PMBus Slave Driver 設定内容一覧(1/2)

項目	設定値	説明
Bus Speed	100kHz	バス速度を指定します。
IICA Input Mode	SMBus	IICA 入力モードを指定します。
SDAA and SCLA signal falling times (tF)[ns]	0	SDAA および SCLA 信号の立ち下がり時間を設定します。(注1)
SDAA and SCLA signal rising times (tR)[ns]	0	SDAA および SCLA 信号の立ち上がり時間を設定します。(注1)
Digital filter	OFF	デジタル・フィルタを有効にするかどうかを指定します。(注2)

## RL78 ファミリ SIS を用いた PMBus 通信サンプルソフトウェア

表 4-7 PMBus Slave Driver 設定内容一覧(2/2)

項目	設定値	説明
Interrupt level for INTIICA0	Level 3(Lowest)	INTIICA0 割り込みの優先度を設定します。
How to set the device address	Setting by GUI	デバイスアドレスの設定方法を指定します。
Device address	0x1A	デバイスアドレスを設定します。
Pin for setting device address 1	P11	デバイスアドレス(bit1)端子を選択します。
Pin for setting device address 2	P12	デバイスアドレス(bit2)端子を選択します。
Pin for setting device address 3	P13	デバイスアドレス(bit3)端子を選択します。
Pin for setting device address 4	P14	デバイスアドレス(bit4)端子を選択します。
Pin for setting device address 5	P15	デバイスアドレス(bit5)端子を選択します。
Pin for setting device address 6	P16	デバイスアドレス(bit6)端子を選択します。
Pin for setting device address 7	P17	デバイスアドレス(bit7)端子を選択します。

表 4-8 PMBus Slave Controller

項目	設定値	説明
Support the SMBus Packet Error Checking (PEC)	Supported	PEC をサポートするかどうかを選択します。

### 4.5.4 アプリケーション層

#### 4.5.4.1 PMBSAPP (PMBus Slave Application)

PMBus 通信機能におけるアプリケーション部です。受信コマンドを解釈し、コマンド受信時の動作を行います。また、Control Signal や Write Protect 信号を受け、デバイス出力の ON/OFF やデータの更新判定を行います。

#### (1) PMBus Slave モジュールからのコールバック時の処理

PMBus コマンド受信時や通信異常の検出時に PMBus Slave モジュール SIS のミドルウェア部(以降、PMBUSCTL と略します)より、コールバック通知を受けます。各コールバック時の処理について説明します。

##### (a) CheckCommandSupported コールバック

コマンドコードを受信したタイミングで PMBUSCTL よりコールバックされます。通知されたコマンドがアプリケーションでサポートされているかを判定し、サポートコマンドであればコールバック関数の戻り値として PMBUSCTL\_CBK\_RTN\_OK を返します。サポート外コマンドであればコールバック関数の戻り値として PMBUSCTL\_CBK\_RTN\_CMD\_NOT\_SUPPORTED を返します。

##### (b) WriteDataReceived コールバック

Write コマンドを受信したタイミングで PMBUSCTL よりコールバックされます。通知された受信データやデータ長が有効値であるかを判定し、有効値であれば内部データを更新し、コールバック関数の戻り値として PMBUSCTL\_CBK\_RTN\_OK を返します。無効値であれば内部データの更新を行わず、戻り値として PMBUSCTL\_CBK\_RTN\_DATA\_NG を返します。また、Write Protect 信号がアクティブの場合も内部データの更新を行わず、PMBUSCTL\_CBK\_RTN\_DATA\_NG を返します。



(c) ReadDataReceived コールバック

Read コマンドを受信したタイミングで PMBSCTL よりコールバックされます。応答対象のデータをセットし、PMBSCTL\_CBK\_RTN\_OK を返します。

(d) FaultNotification コールバック

通信異常を検出したタイミングで PMBSCTL よりコールバックされます。通知された Fault 情報より、STATUS\_WORD 情報を更新し、Host Notify を発行します。

各コールバック関数の詳細やシーケンス情報については下記ドキュメントを参照してください。

- RL78 ファミリ PMBus Slave モジュール Software Integration System (R01AN6985)

(2) アプリケーションタスク処理

アプリケーションのタスク処理は 5ms 間隔で実施され、以下の処理を行います。

(a) HW シグナル情報の取得

PMBSCTL より提供される API 関数”RM\_PMBSCTL\_GetHWSignal”にて HW シグナル情報を取得し、内部データ(Control Signal および Write Protect 信号レベル)を更新します。

(b) LED 現在出力値の取得

調光ドライバより各 LED の現在の出力値を取得し、内部データを更新します。この内部データは READ\_VOUT コマンド受信時に前述の ReadDataReceived コールバックにより応答データとして設定されます。

(c) LED 目標値の設定

調光ドライバに対し、各 LED の目標値として内部データ値を反映します。この内部データは VOUT\_COMMAND 受信時に前述の WriteDataReceived コールバックにより更新されたものです。

## 4.6 ソフトウェア構成

サンプルアプリケーションのソフトウェア構成を記載します。

### 4.6.1 フォルダ構成

本サンプルアプリケーションのフォルダ構成を以下に示します。

表 4.9 フォルダ構成(マスタ)

フォルダ、ファイル名	説明
PMBusMaster_Sample<DIR>	サンプルアプリケーションのフォルダ
\App<DIR>	アプリケーションプログラム格納フォルダ
r_main.c	メインソースファイル
r_pmbm_app.c	PMBusMaster アプリケーションのソースファイル
r_pmbm_app.h	PMBusMaster アプリケーションのヘッダファイル
\Utility<DIR>	共通プログラム格納フォルダ
r_timer16.c	タイマモジュールのソースファイル
r_timer16.h	タイマモジュールのヘッダファイル
r_usertype.h	タイプ定義ヘッダファイル
\src<DIR>	スマート・コンフィグレータ生成ファイル格納フォルダ
\smc_gen<DIR>	スマート・コンフィグレータ生成フォルダ (以下省略) ※配下に PMBus Master モジュール SIS コードが生成されます
\DefaultBuild<DIR> (CC-RL)	プロジェクト出力ファイル格納フォルダ (CC-RL 開発環境のみ)
PMBusMaster_Sample.mot	ファームウェアへの書き込みに使用するモトローラ・S タイプ・フォーマット形式のファイル
\Release\Exe<DIR> (IAR)	プロジェクト出力ファイル格納フォルダ (IAR 開発環境のみ)
PMBusMaster_Sample.mot	ファームウェアへの書き込みに使用するモトローラ・S タイプ・フォーマット形式のファイル

RL78 ファミリ  
SIS を用いた PMBus 通信サンプルソフトウェア

表 4.10 フォルダ構成(スレーブ)

フォルダ、ファイル名	説明
PMBusSlave_Sample<DIR>	サンプルアプリケーションのフォルダ
\App<DIR>	アプリケーションプログラム格納フォルダ
r_main.c	メインソースファイル
r_pmbs_app.c	PMBusSlave アプリケーションのソースファイル
r_pmbs_app.h	PMBusSlave アプリケーションのヘッダファイル
\Driver<DIR>	ドライバプログラム格納フォルダ
r_led.c	調光ドライバのソースファイル
r_led.h	調光ドライバのヘッダファイル
r_led1.c	LED1(Red)用調光ドライバのソースファイル
r_led1.h	LED1(Red)用調光ドライバのヘッダファイル
r_led2.c	LED2(Green)用調光ドライバのソースファイル
r_led2.h	LED2(Green)用調光ドライバのヘッダファイル
r_led3.c	LED3(Blue)用調光ドライバのソースファイル
r_led3.h	LED3(Blue)用調光ドライバのヘッダファイル
\Utility<DIR>	共通プログラム格納フォルダ
r_timer16.c	タイマモジュールのソースファイル
r_timer16.h	タイマモジュールのヘッダファイル
r_usertype.h	タイプ定義ヘッダファイル
\src<DIR>	スマート・コンフィグレータ生成ファイル格納フォルダ
\smc_gen<DIR>	スマート・コンフィグレータ生成フォルダ (以下省略) ※配下に PMBus Slave モジュール SIS コードが生成されます
\DefaultBuild<DIR> (CC-RL)	プロジェクト出力ファイル格納フォルダ (CC-RL 開発環境のみ)
PMBusSlave_Sample.mot	ファームウェアへの書き込みに使用するモトローラ・S タイプ・フォーマット形式のファイル
\Release\Exe<DIR> (IAR)	プロジェクト出力ファイル格納フォルダ (IAR 開発環境のみ)
PMBusSlave_Sample.mot	ファームウェアへの書き込みに使用するモトローラ・S タイプ・フォーマット形式のファイル

#### 4.6.2 オプション・バイトの設定一覧

サンプルアプリケーションのオプション・バイト設定を以下に示します。

表 4.11 オプション・バイト設定一覧

アドレス	設定値	内容
000C0H / 010C0H	01111101B	ウォッチドッグ・タイマ動作許可 (リセット解除後、カウント開始)
000C1H / 010C1H	11111011B	LVD リセット・モード (立ち下がり : 2.97V 立ち上がり : 2.91V)
000C2H / 010C2H	11101010B	HS モード、高速オンチップ・オシレータ : 8MHz
000C3H / 010C3H	10000100B	オンチップ・デバッグ許可

## 5. 注意事項

- 本サンプルアプリケーションはボード RL78/G24 DC/DC LED Control Evaluation Board に合わせた構成・動作となっています。
- 本サンプルアプリケーションが提供する動作はあくまでもサンプルです。アプリケーション層は、使用目的に合わせた設計変更・評価を行ってください。

## 6. 参考ドキュメント

RL78/G24 ユーザーズマニュアル ハードウェア編 (R01UH0961)

RL78 ファミリ PMBus Master モジュール Software Integration System (R01AN7188)

RL78 ファミリ PMBus Slave モジュール Software Integration System (R01AN6985)

Renesas Flash Programmer V3.12 フラッシュ書き込みソフトウェア ユーザーズマニュアル (R20UT5312)

(最新版をルネサスエレクトロニクスホームページから入手してください。)

テクニカルアップデート／テクニカルニュース

(最新版の情報をルネサスエレクトロニクスホームページから入手してください。)

SMBus, PMBus 規格書

PMBus Specification Rev. 1.4 Part I

PMBus Specification Rev. 1.4 Part II

System Management Bus (SMBus) Specification Version 3.2

RL78 ファミリ  
SIS を用いた PMBus 通信サンプルソフトウェア

---

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2023/8/28	-	初版発行
1.01	2024/4/18	-	4.4 機能概要(PMBus Master)を追加

## 製品ご使用上の注意事項

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

### 1. 静電気対策

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

### 2. 電源投入時の処置

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

### 3. 電源オフ時における入力信号

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

### 4. 未使用端子の処理

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

### 5. クロックについて

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

### 6. 入力端子の印加波形

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

### 7. リザーブアドレス（予約領域）のアクセス禁止

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

### 8. 製品間の相違について

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

## ご注意書き

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

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

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

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

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

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

(Rev.5.0-1 2020.10)

## 本社所在地

〒135-0061 東京都江東区豊洲 3-2-24（豊洲フォレストシア）

[www.renesas.com](http://www.renesas.com)

## お問合せ窓口

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

[www.renesas.com/contact/](http://www.renesas.com/contact/)

## 商標について

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