tinyMicon MatisseCORE™

Peripheral Window ユーザーズガイド

ペリフェラルレジスタの値読み書きツール

© 2021 ROHM Co., Ltd.

No. 67UG032J Rev.003
2022.12

変更履歴

日付	バージョン	内容
2021/01/26	Rev.001	新規作成
2021/11/09	Rev.002	Ver.2.0 の機能追加
2022/12/02	Rev.003	下記を追記 ・キーボード操作への対応 ・レジスタの配列定義への対応

© 2021 ROHM Co., Ltd.

目次

1. はじめに 4
2. 概要 5
3. 用語集 5
4. セットアップ 6
4.1. 設定項目 6
4.1.1. ポート番号 6
4.1.2. 設定ファイル 6
5. 起動方法 7
5.1. VS Code から起動 7
5.1.1. デバッグモード開始時 7
5.1.2. デバッグモード中 8
5.2. コマンドライン(cmd)から起動 8
6. ユーザインタフェース 9
6.1. ウィンドウ 9
6.1.1. 画面構成 9
6.2. 使用ファイル 12
6.2.1. 設定ファイル 12
6.2.2. XML ファイル 12
7. エラーメッセージ一覧 14

© 2021 ROHM Co., Ltd.

はじめに

Peripheral Window はロームオリジナル 8bit CPU(Matisse)の開発環境で使用可能な、ペリフェラルレジスタの値を読み書きできる GUI アプリケーションです。ペリフェラルレジスタの構造と値を視覚的に表示し、それらの値を編集することが可能です。

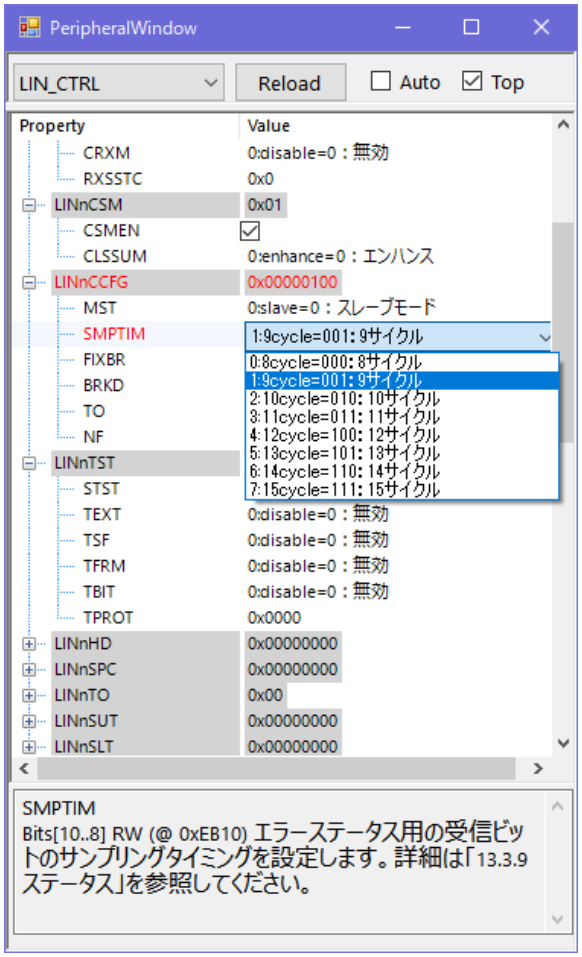

Figure 1. Peripheral Window 画面

© 2021 ROHM Co., Ltd.

概要

以下の図は Peripheral Window と Matisse 開発環境との関係図です。

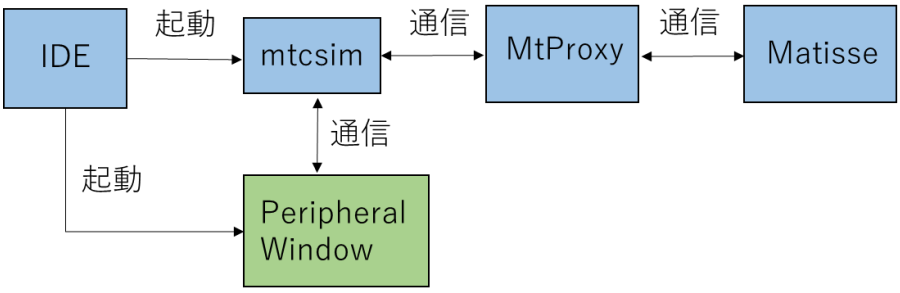

Figure 2. Peripheral Window と Matisse 開発環境との関係図

以下に本アプリケーションの機能の概要を示します。

- Matisse 開発用 IDE(matiseye™-studio)から起動します。
- ペリフェラルレジスタの読み書き(レジスタフィールドの定義により 3 種類の編集パターンで表示・編集)ができます。
- ペリフェラルレジスタ(多バイト)の書き込み順(上位⇔下位バイト順)の設定ができます。
- ペリフェラルレジスタの名称、構造をツリー形式で画面表示します。
- レジスタに関する追加情報を表示します。

用語集

Table 1. 用語集

用語	説明
Matisse	小型 8 ビット CPU コア
mtcsim	Matisse 用 C 言語シミュレータ兼デバッガツール
MtProxy	matiseye™-adapter と matiseye™-adapter Pro の通信処理の差異を吸収するためのツール
IDE	統合開発環境
Visual Studio Code (VS Code)	Microsoft 社が開発した多機能かつ軽量なコードエディタ
matiseye™–studio	Visual Studio Code をベースにした Matisse 向け C 言語開発ツール
ペリフェラル	CPU の周辺回路。タイマなど。

© 2021 ROHM Co., Ltd.

セットアップ

Peripheral Window の起動に必要な設定を以下に示します。

1.1.設定項目

VS Code の設定エディタ(キー入力・・・Ctrl+,)を開きます。検索スペースに「matisse debug」と入力してください。

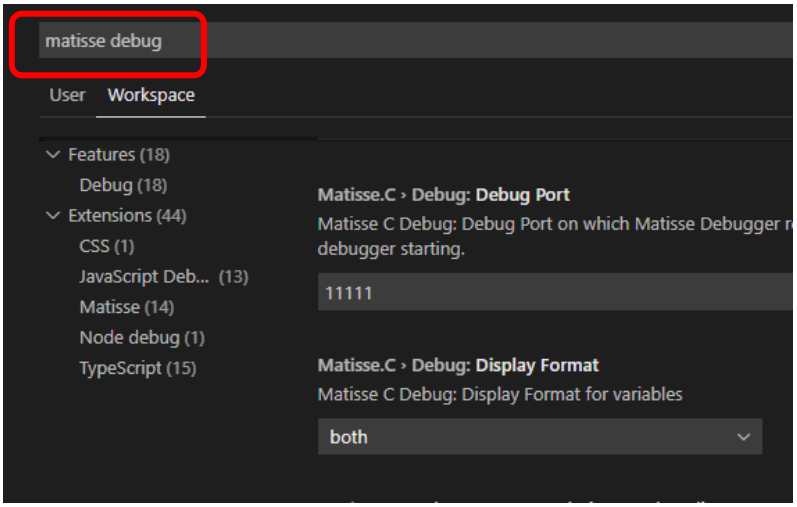

Figure 3. VS Code の設定エディタ

User と Workspace のタブがあり、それぞれグローバル設定・プロジェクト設定をすることができます。Workspace で以下の２つの項目の設定をします。

1.1.1.ポート番号

「Debug Port」でポート番号の設定をすることができます。

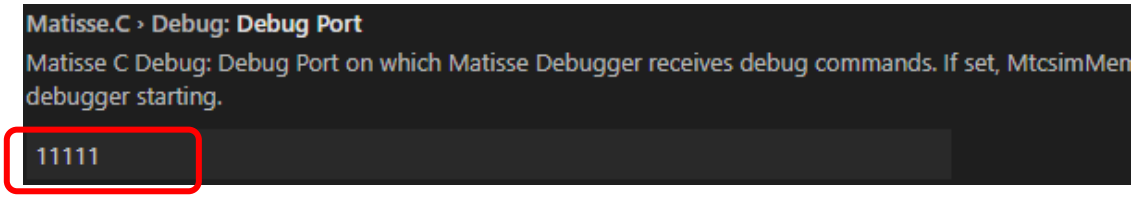

Figure 4. ポート番号の設定

1.1.2.設定ファイル

「Peripheral Window Setting File」で設定ファイルのパスを設定することができます。

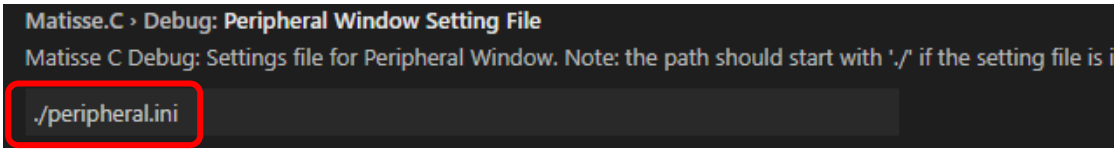

Figure 5. 設定ファイルのパスの設定

設定ファイルの詳細はユーザインタフェースの使用ファイルに記載しています。

© 2021 ROHM Co., Ltd.

起動方法

Peripheral Window の起動方法は VS Code から起動・コマンドラインから起動の 2 つの方法があります。
以下にそれぞれの起動方法を説明します。

1.2.VS Code から起動

Peripheral Window の起動方法は以下に示す VS Code のデバッグモード開始時とデバッグモード中の 2 通りあります。

1.2.1.デバッグモード開始時

設定エディタの「Show Peripheral Window On Start」を true に設定します。

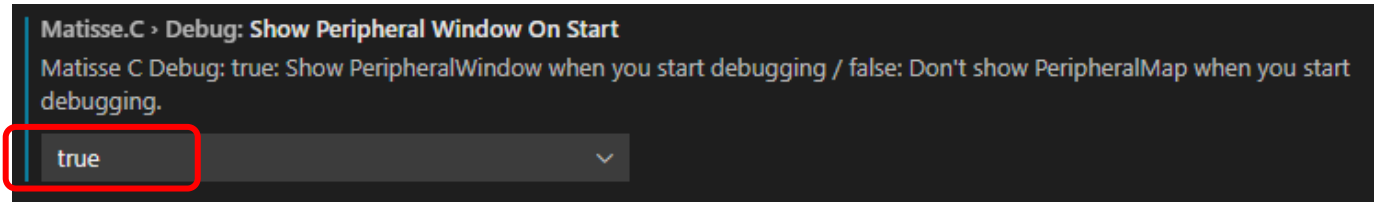

Figure 6. 設定エディタの Peripheral Window 起動設定

次にデバッグタブでデバッグ開始ボタンをクリックします。

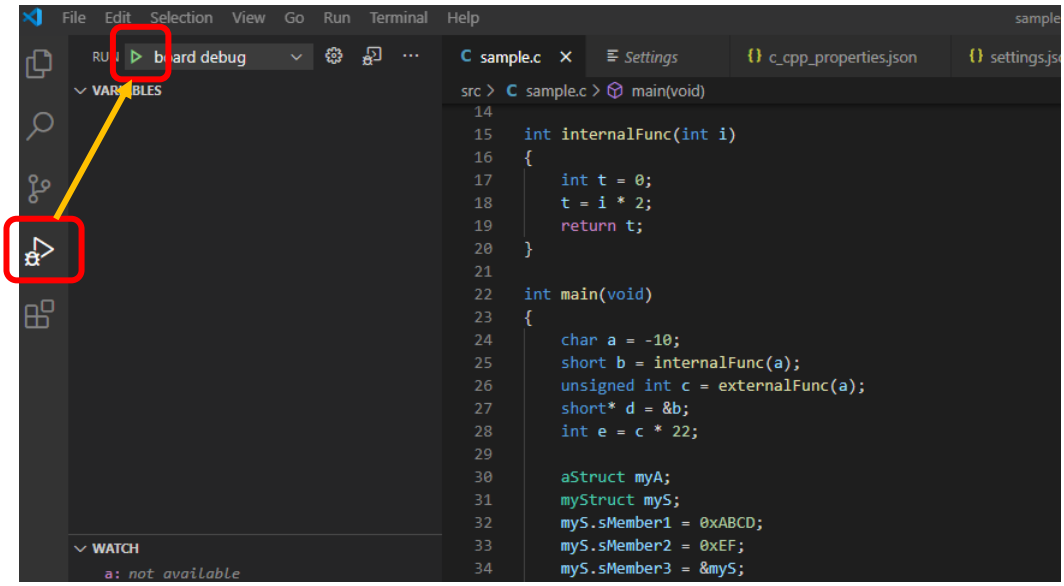

Figure 7. デバッグ実行による Peripheral Window の起動

デバッグモードになり Peripheral Window が起動します。

© 2021 ROHM Co., Ltd.

1.2.2.デバッグモード中

デバッグモード中にコマンドパレットを起動(キー入力・・・Ctrl + Shift + P or F1) し、コマンドパレットに以下のコマンドの入力で起動できます。

「Matisse: Start Peripheral Window」

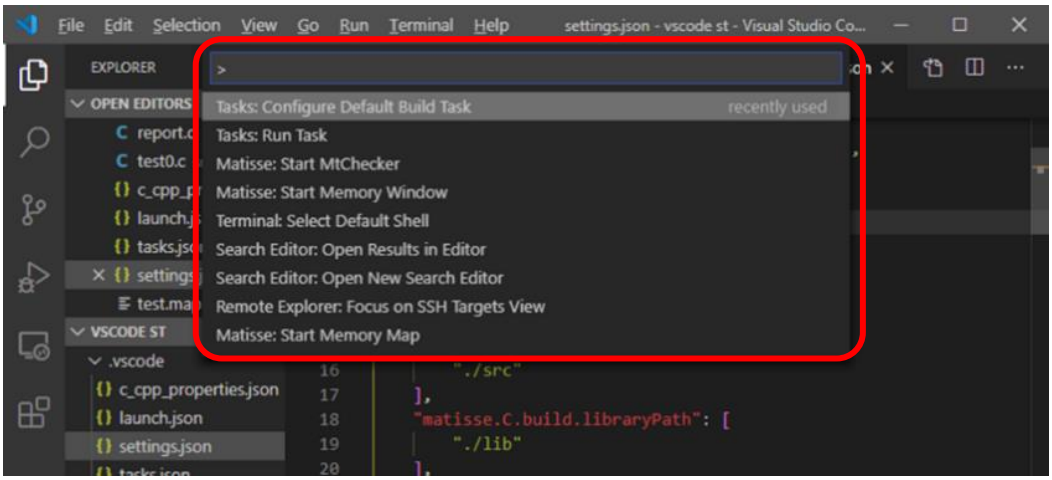

Figure 8. 起動コマンドの入力

1.3.コマンドライン(cmd)から起動

cmd から Peripheral Window 実行ファイル(PeripheralWindow.exe)を起動する場合は、VS Code のデバッグモード中に以下のコマンドラインオプションの入力で起動できます。

```
PeripheralWindow.exe --port=11111 --configPath=..¥..¥..¥PeripheralWindow¥settings.ini
```

Table 2. コマンドラインオプション

フォーマット	説明
--port=	ポート番号を入力してください。
--configPath=	設定ファイルのパスを入力してください。

© 2021 ROHM Co., Ltd.

ユーザインタフェース

1.4.ウィンドウ

1.4.1.画面構成

表示画面の所望のレジスタの Value の値を変更すると、対象のレジスタの書き込みを行い、値をリードバックします。プロパティが読み込み専用の場合、表示がグレーアウトされます。

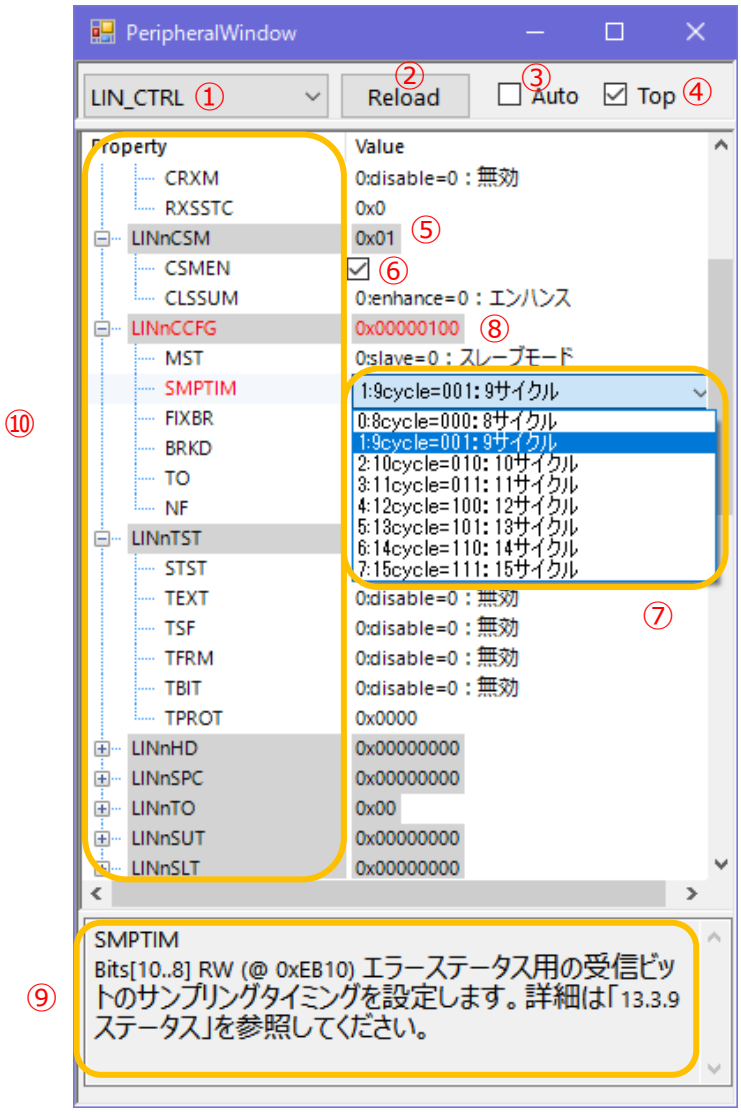

Figure 9. ユーザインタフェース

（次ページへ続く）

© 2021 ROHM Co., Ltd.

以下では、各 UI コンポーネントの機能について説明しています。また、それぞれのコンポーネントを操作するためのキーも、「対応キー」として示しています。キーボード操作は、対象となるコンポーネントにフォーカスが当たっている状態で、指定されたキーを押すことにより実行できます。

① ペリフェラル選択リスト
XML ファイルで定義されたペリフェラル一覧をプルダウンメニュー形式で表示します。
リストから選択したペリフェラルの情報をウィンドウ表示します。
（対応キー"Alt" + "↓"）

②リロードボタン
現在のレジスタ値を読み込み表示します。
（対応キー"Enter"）

③オートリロードチェックボックス
オートリロードのオン・オフ切替ができます。
チェックを入れると 1s 毎にリロードをします。
（対応キー"space"）

④最前面表示チェックボックス
Peripheral Window の最前面表示オン・オフ切替ができます。
チェックを入れると常に最前面表示となります。
（対応キー"space"）

⑤レジスタ値設定テキストボックス
説明表示パネルで示すビット幅に応じて、キーボード入力でレジスタに値の書き込みができます。
5 バイト以上のレジスタの場合、表示値が 4 バイト分の値+"…"になります。例)0x…12345678
（対応キー"F2"）

⑥レジスタ値設定チェックボックス
説明表示パネルに示すビットのオン・オフ切替ができます。
（対応キー"space"）

⑦レジスタ値設定リスト
プルダウンリストのフィールド値を選択すると、値の設定ができます。
（対応キー"Alt + ↓"）

⑧最新変化レジスタ
最新の変化したレジスタ名・値を赤字で表示します。

（次ページへ続く）

© 2021 ROHM Co., Ltd.

⑨説明表示パネル

レジスタの詳細情報を以下のフォーマットで表示します。

[名前 ビット幅 R/W アドレス 説明]

以下に説明表示パネルの例とその内容の説明を示します。



Figure 10. 説明表示パネル

Table 3. 説明表示パネルの説明

表示	内容
UART_CTRL	レジスタ/フィールド名
[Bits 7..0]	レジスタ/フィールドのビット幅
RW	レジスタ/フィールドが読み書き可能(Read/Write)。 読み取り専用は RO、書き込み専用は WO。
(@ 0x9904)	レジスタアドレス値
Control Register	レジスタ/フィールドの説明

⑩レジスタツリー

表示レジスタ名をダブルクリックするとフィールド名の一覧をオープンします。再度ダブルクリックするとフィールド名をクローズします。

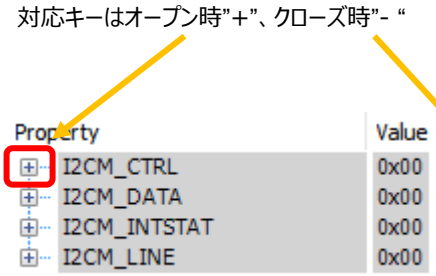

Figure 11. 表示レジスタ名一覧
(クローズ)

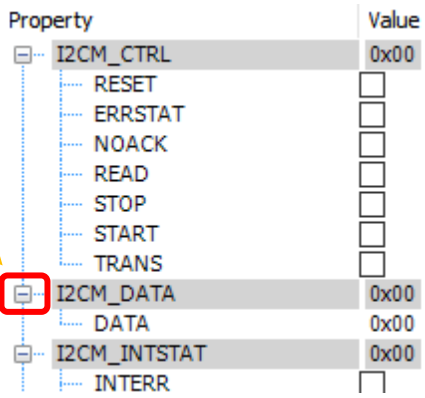

Figure 12. 表示レジスタ名一覧
(オープン)

© 2021 ROHM Co., Ltd.

1.5.使用ファイル

1.5.1.設定ファイル

設定ファイルは、Peripheral Windowを起動する際に必要な情報を記述する.ini形式のファイルです。設定ファイルは起動時に読み込まれます。以下のフォーマットで構成されています。

Table 4. 設定ファイルの項目と説明

設定項目	説明
[Peripheral] SelectedPeripheral=	Peripheral Window の起動時に表示するペリフェラル名を指定します。 何も設定されていない場合は、1 番目のペリフェラルを表示します。 Peripheral Window の終了直前に表示中のペリフェラル名が自動的に設定されます。
[FilePath] XMLFilePath=	機種に合わせたペリフェラル情報が定義されている XML ファイルのパスを指定します。 **注意** ・指定したパスが正しくない場合、Peripheral Window は起動しません。 ・ペリフェラル情報は CMSIS-SVD 形式で記述された XML ファイルのパスを指定してください。

例)

```
[Peripheral]
SelectedPeripheral=APB_GPIO
[FilePath]
XMLFilePath="Matisse_svd.xml"
```

1.5.2.XML ファイル

XML ファイルは、ペリフェラルのレジスタ情報を XML 形式で定義したものです。フォーマットに不備があるとエラーメッセージが表示され、Peripheral Window は起動しません。また、ファイル内でキーワードの追記をすると、以下のような設定ができます。

①多バイトレジスタのバイト単位の書き込み順

多バイトレジスタの場合、キーワードを記述することで書き込み順の指定(上位⇔下位バイト順)ができます。以下に設定方法と XML への記述例を示します。<register>要素の<description>内文字列の先頭に以下のいずれかのキーワードを記述します。

Table 5. XML ファイルのキーワードと説明

キーワード	説明
[WH]	上位→下位
[WL]	下位→上位

例)

```
<register>
        <name>OUT_PRM1</name>
        <description>[WH]Output Control Parameter1</description>
        ・・・
</register>
```

© 2021 ROHM Co., Ltd.

②レジスタの配列表示

レジスタが規定のキーワードと XML タグにより、配列定義されている場合、一つのレジスタとして配列で表示します。配列表示をするためには、下記の 2 つの条件を満たす必要があります。

条件 1 ：<register>要素の<name>内文字列の最後尾にキーワード"[%s]"を記述

例）

```
<register>
        <name> ArrayRegister[%s]</name>
        ・・・
</register>
```

条件 2 ：<register>要素に下記の表のタグを記述

Table 6. レジスタのタグと説明

タグ	説明
<dim>	配列要素の個数
<dimIncrement>	配列要素間のアドレスインクリメント値

例）

```
<register>
        <name> ArrayRegister[%s]</name>
        ・・・
        <dim>4</dim>
        <dimIncrement>1</dimIncrement>
        ・・・
</register>
```

また、画面表示の例を右に示します。

例）・フィールド数 2 の 32 ビットのレジスタ："ArrayRegister"

・レジスタ名：ArrayRegister

・フィールド：Field_A,Field_B

・dim：4

・dimIncrement：1

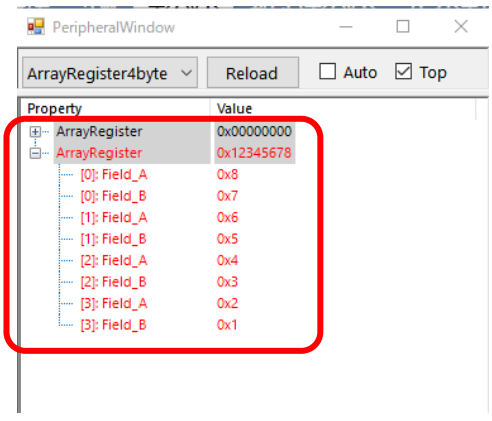

Figure 13. レジスタの配列表示

© 2021 ROHM Co., Ltd.

エラーメッセージ一覧

本アプリケーションで発生する可能性があるエラーメッセージを以下に示します。エラーが発生した場合には、対応するメッセージごとに推奨される対処法をお試しください。

Table 7. エラーメッセージ一覧

イベント	エラーメッセージ	対処方法
起動時	The input format is incorrect. Close the application.	セットアップの章で起動時に必要なポート番号、設定ファイルパスの正しいフォーマットを確認し入力してください。
	The format of the setting file is incorrect. Load the default value.	設定ファイルの記述内容がフォーマットと異なっています。 設定ファイルを確認して修正してください。
	The peripheral described in the setting file does not exist. Shows the default peripherals.	設定ファイルに指定されたペリフェラルは XML ファイルのペリフェラル情報には存在しません。 XML ファイルに記述されたペリフェラル名を指定してください。
	Enter the port number in decimal. Close the application.	ポート番号は 10 進数で入力してください。
	The XML file path is incorrect.	設定ファイルに指定された XML ファイルのパスを確認し、正しいパスを指定してください。
	The read value is not present in the field enumeration list. The XML file and the board's peripherals may not match.	XML ファイルのペリフェラル情報とボードのペリフェラルが異なる可能性があります。正しい XML ファイルまたはボードに変更してください。
	The format of the XML file is invalid.	読み込まれた XML ファイルのフォーマットに誤りがあるようです。フォーマットに必要なタグを削除してしまっていないか、ご確認ください。問題が解決しない場合は、担当者へお問い合わせをお願いします。
	The bitRange format of the XML file is invalid.	XML ファイルのビットサイズ情報<bitRange>の記述に不備がある可能性があります。"[○:○]"のフォーマットで記述されているか確認してください。
	The position of the write order keyword is incorrect. Please put it at the beginning of <description>.	XML ファイルに記述された書き込み順キーワードの位置が不正確です。<description>の先頭に記述してください。
レジスタ値変更時 (キーボード入力)	Input value is too large. Please check the bit size.	レジスタ値の入力はレジスタサイズの範囲内で入力してください。
	The input value is incorrect. Please enter in hexadecimal.	レジスタ値は 16 進数値で入力してください。
	The read value is not present in the field enumeration list. The XML file and the board's peripherals may not match	XML ファイルのペリフェラル情報とボードのペリフェラルが異なっている可能性があります。正しい XML ファイルまたはボードに変更してください。
mtcsim との 通信不良時	"Unable to communicate with mtcsim. Quit the application.	mtcsim との通信不良時に表示され、アプリケーションを終了します。 以下の事項を順に実施してください。 1. VS Code がデバッグモード中にブレイクポイントで停止しているか。 2. ボードの電源を入れ直し、MtProxy を立ち上げてください。その後 1 を確認してください。

© 2021 ROHM Co., Ltd.

オープンソースソフトウェアについて

本ソフトウェアは、ロームが権利を保有もしくは権利許諾を受けているソフトウェアに加えて、下記のライセンス条件の下で提供されているオープンソースソフトウェア（以下「オープンソースソフトウェアプログラム」といいます）を含んでいます。
オープンソースソフトウェアプログラムに関しては、 それぞれの使用許諾条件が適用されますので、万一、オープンソースソフトウェアプログラムのライセンス条件と本資料の間で矛盾が生じた場合は、オープンソースソフトウェアプログラムのライセンス条件が優先されます。

含まれているソフトウェアとそのライセンス条件
・treeviewadv(The BSD License)
・NLog(The BSD License)

商標について

「VS Code」は、マイクロソフト グループの企業の商標です。
「tinyMicon MatisseCORE™」「matiseye™」は、ローム株式会社の商標または登録商標です。

© 2021 ROHM Co., Ltd.

ご注意

1）本資料に記載されているソフトウェア及びシステム（以下あわせて「本ソフトウェア」といいます）に関する情報、及び本資料の記載内容は、本資料発行時点のものであり、予告なく変更することがあります。

2）本製品又は本ソフトウェアを、極めて高度な信頼性が要求され、その故障や誤動作が人の生命、身体への危険若しくは損害、又はその他の重大な損害の発生に関わるような機器又は装置（医療機器、輸送機器、交通機器、航空宇宙機器、原子力制御装置、燃料制御、カーアクセサリを含む車載機器、各種安全装置等）（以下「特定用途」といいます）へのご使用を検討される際は事前にローム営業窓口までご相談くださいますようお願い致します。ロームの文書による事前の承諾を得ることなく、特定用途に使用したことによりお客様又は第三者に生じた損害等に関し、ロームは一切その責任を負いません。

3）ロームは常に品質・信頼性の向上に取り組んでいますが、半導体製品は種々の要因で故障・誤作動する可能性があります。万一、本ソフトウェアと使用するローム製品（以下「本製品」といいます）が故障・誤作動した場合であっても、その影響により人身事故、火災損害等が起こらないようご使用機器でのディレーティング、冗長設計、延焼防止、バックアップ、フェイルセーフ等の安全確保をお願いします。定格を超えたご使用や使用上の注意事項が守られていない場合、ロームは一切その責任を負いません。

4）ロームは、本ソフトウェアの使用または使用不能によってお客様に生じるいかなる直接的、間接的損害(情報、データ、プログラム等の無体財産の損失、利益の喪失、中断などによる損害を含む)に関して、一切の責任を負いません。

5）本ソフトウェア、本資料及び本製品には、ロームに帰属する機密情報、技術情報又はトレードシークレットが記載されておりますので、ロームの文書による事前の承諾を得ることなく、その全部又は一部について以下の行為をすることは固くお断り致します。
(ア) 第三者へ開示すること
(イ) 逆アセンブル、リバースエンジニアリングその他の解析をすること
(ウ) 転載、複写、複製すること
(エ) 本ソフトウェアに含まれている著作権表示を取り除くこと

6）本製品、本ソフトウェア及び本資料に記載の技術を輸出または国外へ提供する際には、「外国為替及び外国貿易法」、「米国輸出管理規則」など適用される輸出関連法令を遵守し、それらの定めにしたがって必要な手続きを行ってください。

7）ロームは、本ソフトウェア及び本資料に記載された情報に関し第三者の知的財産権及びその他の権利について権利侵害がないことを保証するものではありません。従いまして、上記権利侵害に関し、ロームは一切その責任を負いません。

8）ロームは、本資料に記載された情報についてローム若しくは第三者が所有又は管理している知的財産権その他の権利の実施又は利用を、本資料によって、明示的にも黙示的にも、お客様に許諾するものではありません。

9）お客様は、（１）お客様による第三者の権利の侵害（オープンソースソフトウェアプログラムのライセンス条件への違反を含むがこれに限定されない）や法令違反により生じる、または、（２）お客様が本資料に違反することやこれに関連して生じる賠償金等の請求による損失、罰金や費用（弁護士費用を含むがこれに限定されない）から、ローム及びロームの役員や従業員を免責することを保証し、損害を与えないようにし、また保護するものとします。

１０）本資料は、正確を期して慎重に作成したものですが、誤りがないことを保証するものではありません。万一、本資料に記載された内容の誤りに起因してお客様又は第三者に損害が生じた場合であっても、ロームは一切その責任を負いません。

ローム製品のご利用ありがとうございます。
製品に関するお問い合わせは、下記より願い致します。

ROHM Customer Support System

http://www.rohm.co.jp/contact/

© 2024 ROHM Co., Ltd. All rights reserved.