

お客様各位

---

## カタログ等資料中の旧社名の扱いについて

---

2010年4月1日を以ってNECエレクトロニクス株式会社及び株式会社ルネサステクノロジが合併し、両社の全ての事業が当社に承継されております。従いまして、本資料中には旧社名での表記が残っておりますが、当社の資料として有効ですので、ご理解の程宜しくお願い申し上げます。

ルネサスエレクトロニクス ホームページ (<http://www.renesas.com>)

2010年4月1日

ルネサスエレクトロニクス株式会社

【発行】ルネサスエレクトロニクス株式会社 (<http://www.renesas.com>)

【問い合わせ先】 <http://japan.renesas.com/inquiry>

## ご注意書き

1. 本資料に記載されている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。当社製品のご購入およびご使用にあたりましては、事前に当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、当社ホームページなどを通じて公開される情報に常にご注意ください。
2. 本資料に記載された当社製品および技術情報の使用に関連し発生した第三者の特許権、著作権その他の知的財産権の侵害等に関し、当社は、一切その責任を負いません。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。
3. 当社製品を改造、改変、複製等しないでください。
4. 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。お客様の機器の設計において、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には、お客様の責任において行ってください。これらの使用に起因しお客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
5. 輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続を行ってください。本資料に記載されている当社製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途の目的で使用しないでください。また、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に使用することができません。
6. 本資料に記載されている情報は、正確を期すため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本資料に記載されている情報の誤りに起因する損害がお客様に生じた場合においても、当社は、一切その責任を負いません。
7. 当社は、当社製品の品質水準を「標準水準」、「高品質水準」および「特定水準」に分類しております。また、各品質水準は、以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認ください。お客様は、当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途に当社製品を使用することができません。また、お客様は、当社の文書による事前の承諾を得ることなく、意図されていない用途に当社製品を使用することができません。当社の文書による事前の承諾を得ることなく、「特定水準」に分類された用途または意図されていない用途に当社製品を使用したことによりお客様または第三者に生じた損害等に関し、当社は、一切その責任を負いません。なお、当社製品のデータ・シート、データ・ブック等の資料で特に品質水準の表示がない場合は、標準水準製品であることを表します。  
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  
高品質水準： 輸送機器（自動車、電車、船舶等）、交通用信号機器、防災・防犯装置、各種安全装置、生命維持を目的として設計されていない医療機器（厚生労働省定義の管理医療機器に相当）  
特定水準： 航空機器、航空宇宙機器、海底中継機器、原子力制御システム、生命維持のための医療機器（生命維持装置、人体に埋め込み使用するもの、治療行為（患部切り出し等）を行うもの、その他直接人命に影響を与えるもの）（厚生労働省定義の高度管理医療機器に相当）またはシステム等
8. 本資料に記載された当社製品のご使用につき、特に、最大定格、動作電源電圧範囲、放熱特性、実装条件その他諸条件につきましては、当社保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障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おりますが、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、当社製品は耐放射線設計については行っておりません。当社製品の故障または誤動作が生じた場合も、人身事故、火災事故、社会的損害など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、機器またはシステムとしての出荷保証をお願いいたします。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、お客様が製造された最終の機器・システムとしての安全検証をお願いいたします。
10. 当社製品の環境適合性等、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。お客様がかかる法令を遵守しないこと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。
11. 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固くお断りいたします。
12. 本資料に関する詳細についてのお問い合わせその他お気付きの点等がございましたら当社営業窓口までご照会ください。

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とその総株主の議決権の過半数を直接または間接に保有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。

# Peripheral Driver Generator V.1.03

ユーザーズマニュアル

## 本資料ご利用に際しての留意事項

1. 本資料は、お客様に用途に応じた適切な弊社製品をご購入いただくための参考資料であり、本資料中に記載の技術情報について弊社または第三者の知的財産権その他の権利の実施、使用を許諾または保証するものではありません。
2. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例など全ての情報の使用に起因する損害、第三者の知的財産権その他の権利に対する侵害に関し、弊社は責任を負いません。
3. 本資料に記載の製品および技術を大量破壊兵器の開発等の目的、軍事利用の目的、あるいはその他軍事用途の目的で使用しないでください。また、輸出に際しては、「外国為替及び外国貿易法」その他輸出関連法令を遵守し、それらの定めるところにより必要な手続を行ってください。
4. 本資料に記載の製品データ、図、表、プログラム、アルゴリズムその他応用回路例などの全ての情報は本資料発行時点のものであり、弊社は本資料に記載した製品または仕様等を予告なしに変更することがあります。弊社の半導体製品のご購入およびご使用に当たりましては、事前に弊社営業窓口で最新の情報をご確認頂きますとともに、弊社ホームページ(<http://www.renesas.com>)などを通じて公開される情報に常にご注意下さい。
5. 本資料に記載した情報は、正確を期すため慎重に制作したのですが、万一本資料の記述の誤りに起因する損害がお客様に生じた場合においても、弊社はその責任を負いません。
6. 本資料に記載の製品データ、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流用する場合は、流用する情報を単独で評価するだけでなく、システム全体で十分に評価し、お客様の責任において適用可否を判断して下さい。弊社は、適用可否に対する責任を負いません。
7. 本資料に記載された製品は、各種安全装置や運輸・交通用、医療用、燃焼制御用、航空宇宙用、原子力、海底中継用の機器・システムなど、その故障や誤動作が直接人命を脅かしあるいは人体に危害を及ぼすおそれのあるような機器・システムや特に高度な品質・信頼性が要求される機器・システムでの使用を意図して設計、製造されたものではありません（弊社が自動車用と指定する製品を自動車に使用する場合を除きます）。これらの用途に利用されることをご検討の際には、必ず事前に弊社営業窓口へご照会下さい。なお、上記用途に使用されたことにより発生した損害等について弊社はその責任を負いかねますのでご了承願います。
8. 第7項にかかわらず、本資料に記載された製品は、下記の用途には使用しないで下さい。これらの用途に使用されたことにより発生した損害等につきましては、弊社は一切の責任を負いません。
  - 1) 生命維持装置。
  - 2) 人体に埋め込み使用するもの。
  - 3) 治療行為（患部切り出し、薬剤投与等）を行なうもの。
  - 4) その他、直接人命に影響を与えるもの。
9. 本資料に記載された製品のご使用につき、特に最大定格、動作電源電圧範囲、放熱特性、実装条件およびその他諸条件につきましては、弊社保証範囲内でご使用ください。弊社保証値を越えて製品をご使用された場合の故障および事故につきましては、弊社はその責任を負いません。
10. 弊社は製品の品質および信頼性の向上に努めておりますが、特に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。弊社製品の故障または誤動作が生じた場合も人身事故、火災事故、社会的損害など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、誤動作防止設計などの安全設計（含むハードウェアおよびソフトウェア）およびエージング処理等、機器またはシステムとしての出荷保証をお願いいたします。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、お客様が製造された最終の機器・システムとしての安全検証をお願い致します。
11. 本資料に記載の製品は、これを搭載した製品から剥がれた場合、幼児が口に入れて誤飲する等の事故の危険性があります。お客様の製品への実装後に容易に本製品が剥がれることがなきよう、お客様の責任において十分な安全設計をお願いします。お客様の製品から剥がれた場合の事故につきましては、弊社はその責任を負いません。
12. 本資料の全部または一部を弊社の文書による事前の承諾なしに転載または複製することを固くお断り致します。
13. 本資料に関する詳細についてのお問い合わせ、その他お気付きの点等がございましたら弊社営業窓口までご照会下さい。

## はじめに

Peripheral Driver Generator(以降、PDG と略語で記述します)はマイコン内臓周辺 I/O モジュールのドライバ作成を支援するツールです。

ユーザは、PDG に同梱する周辺 I/O モジュール用 API ライブラリを呼び出すための関数を、PDG の画面上で設定し、自動生成することができます。

PDGは、IBM PC互換機のMicrosoft Windows上で動作します。

サポート対象マイコンは、SH/Tinyシリーズ、H8/300H Tinyシリーズ、R8C/Tinyシリーズ、M16C/TinyシリーズおよびM16C/60シリーズ内の主要グループです。詳細は本マニュアルの概要を参照してください。

## ご使用上の注意事項

PDGが出力する関数及びAPIライブラリに関して評価を行っていますが、本ソフトウェアを使用して開発する場合は、お客様の責任のもと十分な動作確認を行うようお願いいたします。

IBM はInternational Business Machines Corporation の登録商標です。  
Microsoft, Windowsは米国Microsoft社の米国およびその他の国における登録商標です。  
その他、記載されている製品名は各社の商標または登録商標です。

製品内容及び本書についてのお問い合わせ先

株式会社ルネサス テクノロジ  
コンタクトセンタcsc@renesas.com  
ホームページ<http://japan.renesas.com/tools>

## 目次

<b>1. 概要</b> .....	<b>1-1</b>
1.1 PDGの機能.....	1-1
1.2 PDGのプロジェクト.....	1-1
1.3 PDGの役割.....	1-1
1.4 動作環境.....	1-2
1.5 コンパイラとの組み合わせ.....	1-3
1.6 同梱しているAPIライブラリ.....	1-3
1.7 メイン画面.....	1-4
1.7.1 設定内容表示ウィンドウ .....	1-4
1.7.2 設定パターン新規作成ウィンドウ .....	1-5
1.7.3 生成ファイル情報ウィンドウ.....	1-5
1.8 メニュー.....	1-7
1.9 ツールバー .....	1-10
<b>2. 使用前の準備</b> .....	<b>2-1</b>
2.1 インストール .....	2-1
2.2 エディタの設定 .....	2-1
2.3 HEWへのPDGの登録 .....	2-2
2.4 HEWのHEWTARGETSERVER設定.....	2-4
<b>3. PDGの操作方法</b> .....	<b>3-1</b>
3.1 PDGを使用したアプリケーションの開発 .....	3-1
3.2 PDGの操作の流れ .....	3-2
3.3 プロジェクトの作成/プロジェクトを開く.....	3-3
3.3.1 プロジェクトの新規作成.....	3-3
3.3.2 既存プロジェクトを開く.....	3-6
3.3.3 CPUクロックの設定 .....	3-7
3.4 周辺I/Oモジュールの選択・設定 .....	3-8
3.4.1 周辺I/Oモジュールの設定パターンの新規作成 .....	3-8
3.4.2 周辺I/Oモジュールの設定パターンの変更 .....	3-10
3.4.3 周辺I/Oモジュールの設定パターンの複製 .....	3-10
3.4.4 周辺I/Oモジュールの設定パターンの解除 .....	3-11
3.5 資源の割り当てと解除.....	3-11
3.5.1 資源の割り当て .....	3-11
3.5.2 割り当てた資源の解除.....	3-11
3.6 一括ソース生成 .....	3-12
3.7 出力関数情報をCSV形式で見る.....	3-12
3.8 出力関数情報を更新する .....	3-12
3.9 生成ファイルのHEWプロジェクトへの登録.....	3-12
3.9.1 登録機能 .....	3-12
3.9.2 登録の操作方法.....	3-13
3.9.3 登録したソースの登録解除.....	3-14

<b>4. クロック、周辺I/Oモジュール設定方法</b> .....	<b>4-1</b>
4.1 クロックの設定 .....	4-1
4.1.1 M16C/62P、M16C/28、M16C/28B、M16C/29のクロック設定 .....	4-1
4.1.2 R8C/13のクロック設定 .....	4-4
4.1.3 R8C/22-23のクロック設定 .....	4-6
4.1.4 R8C/24-25のクロック設定 .....	4-8
4.1.5 R8C/26-29、R8C/2A-2Dのクロック設定 .....	4-11
4.1.6 H8/3687、H8/36049のクロック設定 .....	4-14
4.1.7 H8/36077、H8/36109のクロック設定 .....	4-16
4.1.8 SH7125のクロック設定 .....	4-18
4.2 シリアルインタフェースの設定 .....	4-20
4.2.1 同期式シリアインタフェース .....	4-21
4.2.2 非同期式シリアインタフェース .....	4-26
4.3 I/Oポートの設定 .....	4-28
4.4 タイマの設定 .....	4-30
4.4.1 タイマモード .....	4-31
4.4.2 イベントカウンタモード .....	4-36
4.4.3 パルス幅変調モード .....	4-42
4.4.4 パルス周期測定モード .....	4-51
4.4.5 パルス幅測定モード .....	4-56
4.4.6 インพุットキャプチャモード .....	4-58
4.4.7 アウトプットコンペアモード .....	4-65
4.5 割り込みの設定 .....	4-74
4.6 A/D変換の設定 .....	4-77
<b>5. 生成関数リファレンス</b> .....	<b>5-1</b>
5.1 M16C/60、M16C/TINY、R8C/TINY、H8/300H TINYの生成関数 .....	5-1
5.2 SH/TINYの生成関数 .....	5-12
<b>6. プロジェクトコンバート</b> .....	<b>6-1</b>
6.1 コンバート機能 .....	6-1
6.2 コンバートによる設定内容の変更と表示 .....	6-1
6.3 プロジェクトコンバートの操作方法 .....	6-2

## 1. 概要

### 1.1 PDGの機能

PDGは、シリアル/タイマ/IOなどのマイコン周辺 I/O モジュールの設定を GUI 上で入力し、設定した内容で周辺 I/O モジュール用 API ライブラリを呼び出すための関数を出力します。

各周辺 IO の設定を GUI 上で支援します。

設定した内容を関数で出力します。

自動生成したソースを High-performance Embedded Workshop (以降 HEW と略語で記述します)のプロジェクトに一括登録します。

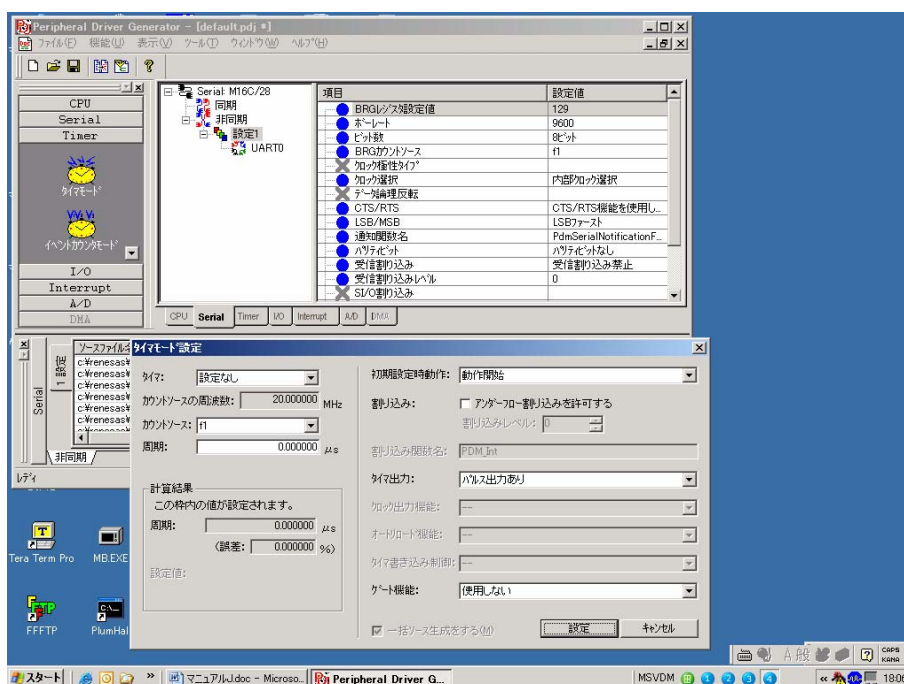


図 1.1 PDGの表示例

本 GUI で設定した内容を、マイコン間で流用することを目的としてコンバートをサポートします。

注意：マイコンによってはコンバートできません。

### 1.2 PDGのプロジェクト

PDGはプロジェクトという概念で情報を管理します。プロジェクトとして管理しているのは以下のものです。

周辺 IO 毎の設定情報

設定内容の関数管理情報

### 1.3 PDGの役割

PDGが生成した API ライブラリを呼び出す関数を、ユーザのプログラムに組み込むことによりアプリケーションを形成します。



以下に、PDG と、API ライブラリ、アプリケーションの関係を示します。

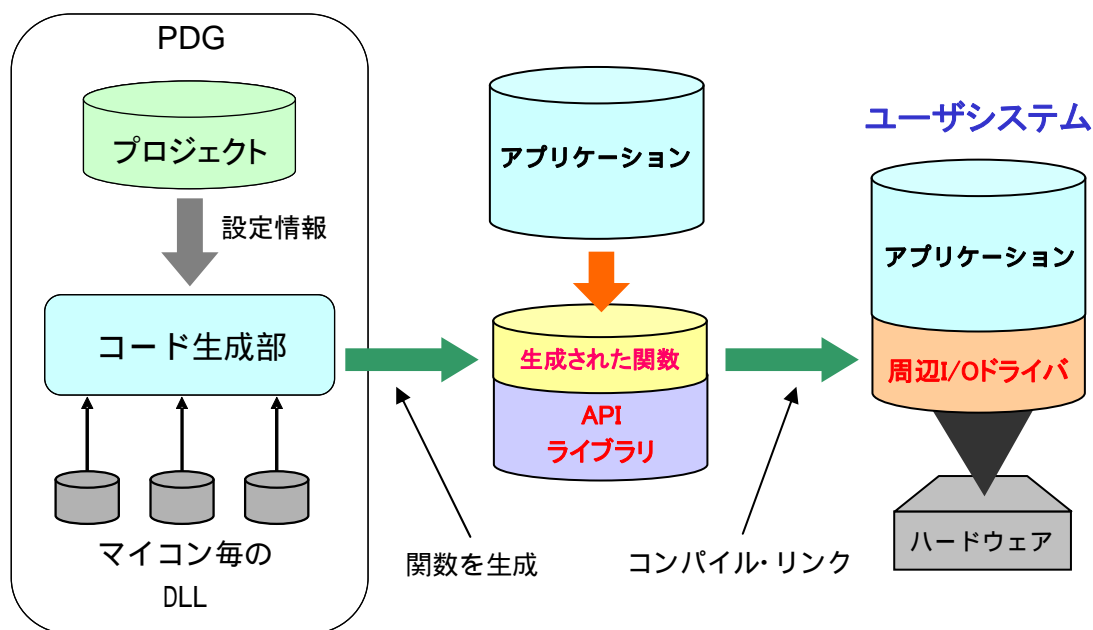


図 1.2 PDGの役割

## 1.4 動作環境

PDG の動作を確認しているホストマシン、および OS のバージョンについて以下に示します。

表 1.1 ホストマシン

ホスト名	OS のバージョン
IBMPC/AT 互換機	Microsoft Windows2000
	Microsoft WindowsXP

上記以外のホストおよび OS 上での動作については、ホストマシンおよび OS の供給メーカーに依存しますので、上記条件で動作するソフトウェアがお客様のホストマシンおよび OS で動作するかどうかを供給メーカーにお問い合わせ願います。

推奨するハードウェアは以下の通りです。

表 1.2 推奨するホストマシンのハードウェア仕様

メインメモリ	OS が正常に動作する状態を推奨(256M バイト以上)
空きディスク容量	70M バイト以上
ディスプレイ	1024×768 以上を推奨

## 1.5 コンパイラとの組み合わせ

PDG は以下に示すコンパイラとの組み合わせにおいて評価しています。

表 1.3 コンパイラパッケージ

マイコンシリーズ	コンパイラ製品
M16C/Tiny, M16C/60, R8C/Tiny	M16Cシリーズ用Cコンパイラパッケージ M3T-NC30WA V.5.40 Release 00
H8/300H Tiny	H8SX,H8S,H8ファミリ用C/C++コンパイラパッケージ V.6.01 Release 02
SH/Tiny	SuperHファミリ用C/C++コンパイラパッケージ V.9.02 Release00

## 1.6 同梱しているAPIライブラリ

同梱している API ライブラリの種類を以下に示します。

表 1.4 APIライブラリー一覧

シリーズ	格納ディレクトリ	ライブラリファイル名
H8/300H Tiny	lib¥h8_tiny	rapi_h8_3687.lib rapi_h8_36049.lib rapi_h8_36077.lib rapi_h8_36109.lib
R8C/Tiny	lib¥r8c_tiny	rapi_r8c_13.lib rapi_r8c_22_23.lib rapi_r8c_24_25.lib rapi_r8c_26_27.lib rapi_r8c_28_29.lib rapi_r8c_2A_2B.lib rapi_r8c_2C_2D.lib
M16C/Tiny	lib¥m16c_tiny	rapi_m16c_28.lib rapi_m16c_29.lib
M16C/60	lib¥m16c	rapi_m16c_62p.lib
SH/Tiny	Lib¥sh_tiny	rapi_sh7125.lib

これらのライブラリは表 1.3 に示すコンパイラで作成されています。ライブラリを使用する場合は、表 1.3 と同一のコンパイラを使用してください。ライブラリのソースは“source”フォルダに同梱されています。

## 1.7 メイン画面

### 1.7.1 設定内容表示ウィンドウ

現在開いているプロジェクトファイルの設定内容を表示します。

下辺のタブは機能を、左側画面のツリーは作成した設定パターンを、右側画面のリストは左画面で選択中の設定の詳細を、表示します。左側画面の「設定」、又は、右側画面の設定項目名をダブルクリックすると、その機能の設定ダイアログボックスが開きます。

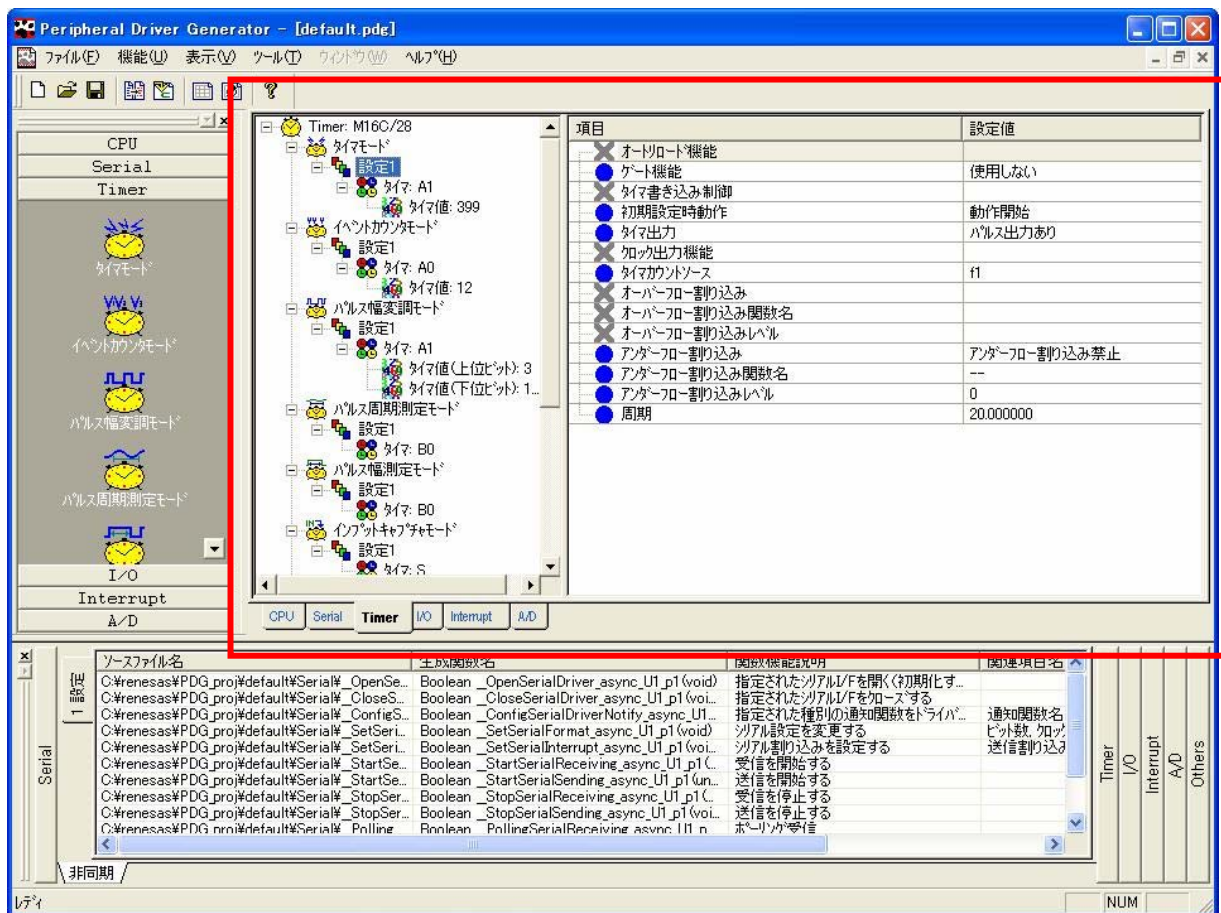


図 1.3 設定内容表示ウィンドウ

### 1.7.2 設定パターン新規作成ウィンドウ

プロジェクトファイルを開いている状態でボタンが有効になります。

機能を選択し、モードのボタンをクリックすると、各機能の設定ダイアログボックスが開き、新しく設定パターンを作成することができます。

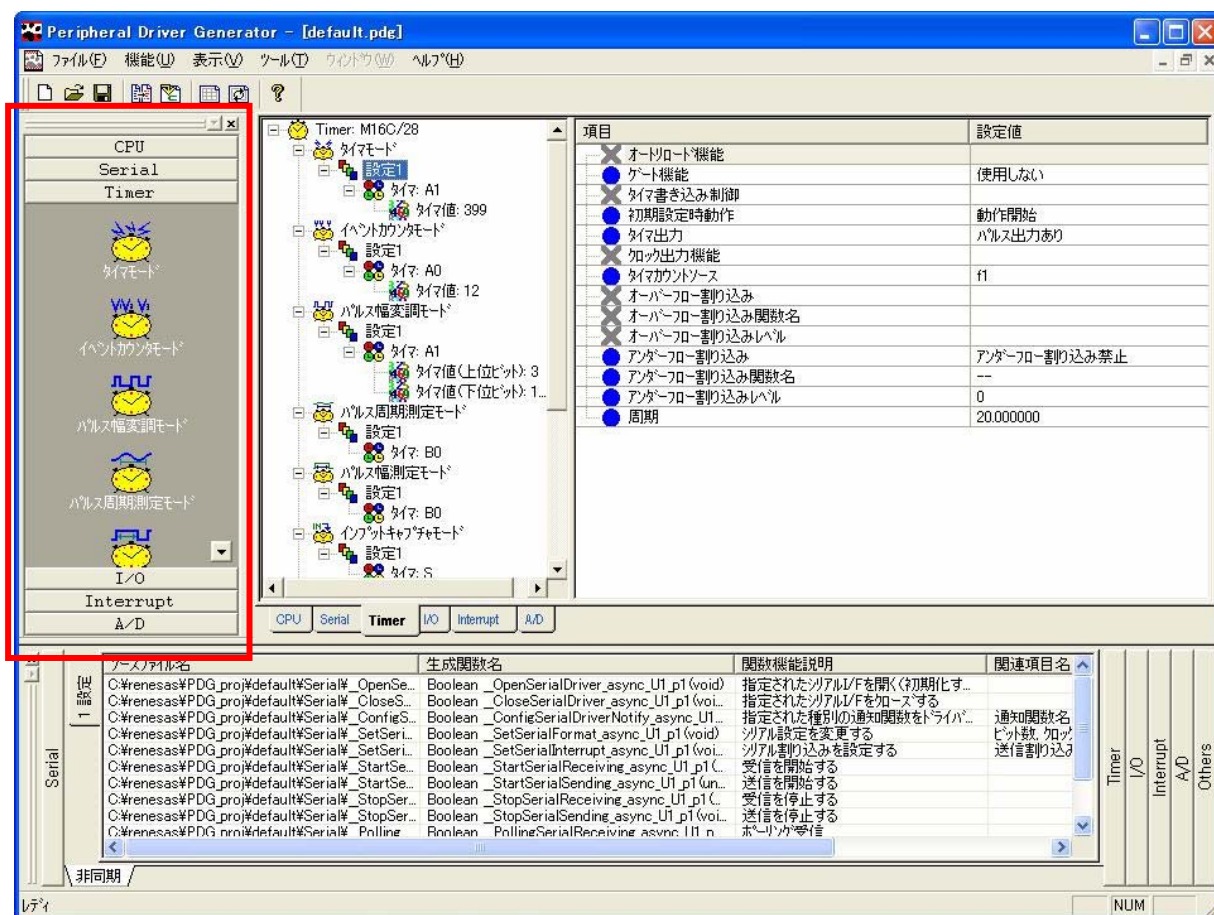


図 1.4 設定パターン新規作成ウィンドウ

### 1.7.3 生成ファイル情報ウィンドウ

#### (1) 表示内容

現在開いているプロジェクトファイルの生成ファイル情報を、機能ごと、モードごとに表示します。生成ソース情報としては、次の内容をリストで表示します。

- 生成ファイル名
- 生成関数名
- 関数の機能説明
- 関連項目名

生成ファイル名をダブルクリックすると、(指定のエディタで)ファイルを開きます。

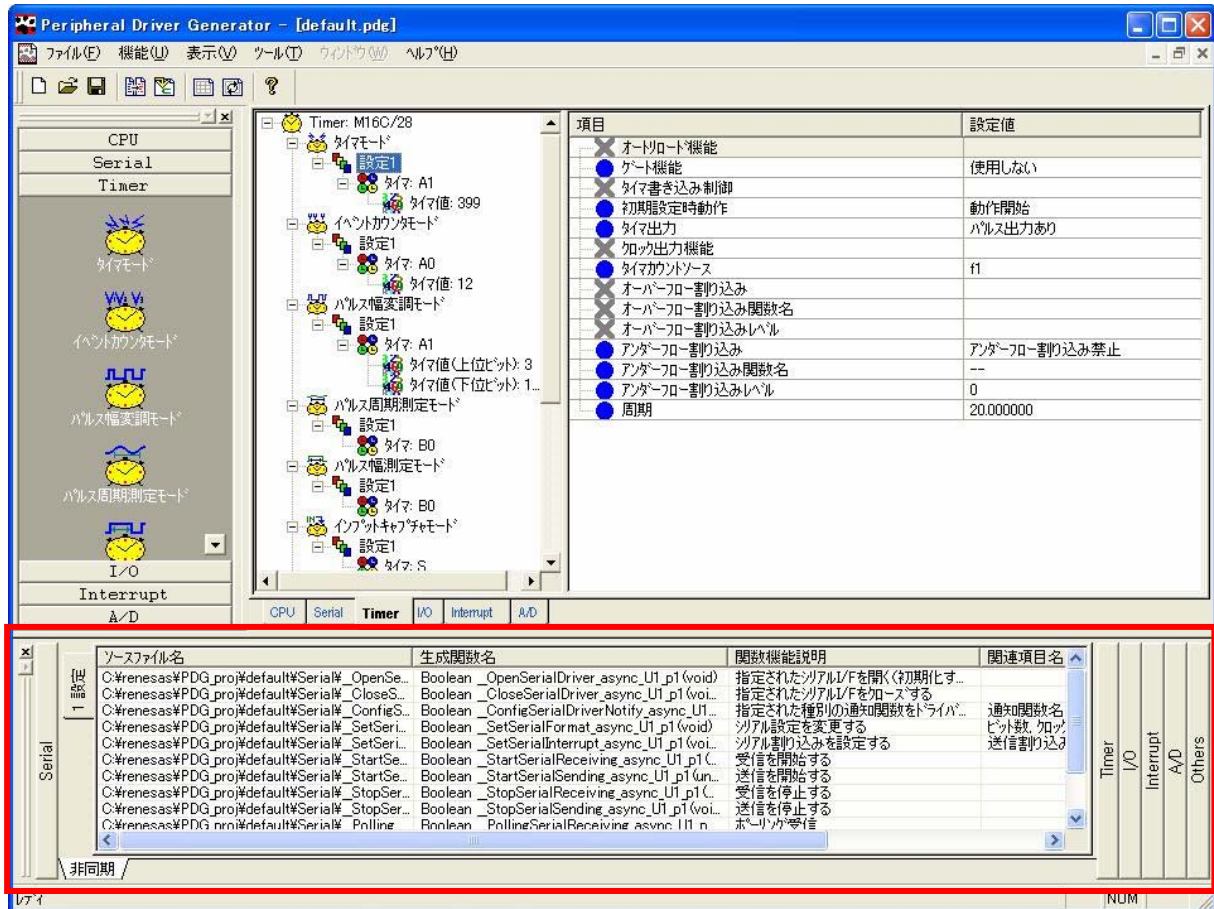


図 1.5 生成ファイル情報ウィンドウ

## (2) 文字サイズの変更

生成ファイル情報ウィンドウにマウスポインタを合わせて右クリックをするか、「表示」 - 「生成ファイル情報ウィンドウの文字の大きさ」メニューを選択します。

「大」「中」「小」からサイズを選択します。

リストの表示文字サイズが変わります。



## 1.8 メニュー

表 1.5 にメニューの一覧を示します。

表 1.5 メニュー一覧

トップメニュー	階層メニュー		説明	
ファイル (F)	プロジェクト新規作成(N)		プロジェクトを新規に作成します。 随時使用可能	
	プロジェクトを開く(O)		既存プロジェクトを開きます。 随時使用可能	
	プロジェクト保存(S)		現在開いているプロジェクトを保存します。 随時使用可能	
	名前を付けてプロジェクトを保存(A)		現在開いているプロジェクトを別の名前を付けて保存します。 随時使用可能	
	プロジェクトコンバート(C)		既存プロジェクトのCPUを変更した新しいプロジェクトを作成します。 プロジェクトを開いているときのみ使用可能 マイコンによってはコンバートを使用できません	
	一括ソース生成(S)		ソースファイルを作成します。 周辺機能I/O設定完了時に使用可能	
	一括ソース削除(D)		生成したソースファイルをすべて削除します。 ソース生成実行後に使用可能	
	履歴		開いたプロジェクトを一覧できます。 随時使用可能	
	アプリケーションの終了(X)		Peripheral Driver Generator を終了します。 随時使用可能	
機能(U)	CPU(C)	設定の修正(M)	CPUに関する設定の修正を行います。 プロジェクトを開いているときのみ使用可能	
	シリアル(S)	設定の新規作成(N)	同期(S)	シリアル同期の設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能
			非同期(A)	シリアル非同期の設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能
		設定の複製(C)	シリアルの設定パターン( 1)を複製します。 シリアルの設定を選択中のみ使用可能	
		設定の削除(D)	シリアルの設定パターン( 1)を削除します。 シリアルの設定を選択中のみ使用可能	
		設定の修正(M)	シリアルに関する設定の修正を行います。 シリアルの設定を選択中のみ使用可能	
		UART番号の設定(S)	シリアルの設定パターン( 1)にUARTを割り当てます。 シリアルの設定を選択中のみ使用可能	
		UART番号の削除(L)	シリアルの設定パターン( 1)からUARTの割り当てを解除します。 UART選択中のみ使用可能	

トップメニュー	階層メニュー	説明	
	A/D(A)	設定の新規作成(N)	単発モード(S) A/D単発モードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能
		繰り返しモード(R)	A/D繰り返しモードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		単掃引モード(G)	A/D単掃引モードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		繰り返し掃引モード0(W)	A/D繰り返し掃引0モードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		繰り返し掃引モード1(E)	A/D繰り返し掃引1モードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		同時サンプル掃引モード(P)	A/D同時サンプル掃引モードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		遅延トリガモード0(D)	A/D遅延トリガ0モードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		遅延トリガモード1(L)	A/D遅延トリガ1モードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		2チャンネルスキャンモード(H)	2チャンネルスキャンモードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		2チャンネル連続スキャンモード(T)	2チャンネル連続スキャンモードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		4チャンネルスキャンモード(F)	4チャンネルスキャンモードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		4チャンネル連続スキャンモード(O)	4チャンネル連続スキャンモードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		設定の複製(C)	A/Dの設定パターン( 1)を複製します。 A/Dの設定を選択中のみ使用可能
		設定の削除(D)	A/Dの設定パターン( 1)を削除します。 A/Dの設定を選択中のみ使用可能
		設定の修正(M)	A/Dに関する設定の修正を行います。 A/Dの設定を選択中のみ使用可能
入力グループ・入力端子の設定(I)	A/Dの設定パターン( 1)に入力グループ・入力端子を割り当てます。 A/Dの設定を選択中のみ使用可能		

トップメニュー	階層メニュー		説明	
		入力グループ・入力端子の削除(L)	A/Dの設定パターン( 1)から入力グループ・入力端子の割り当てを解除します。 入力グループ・入力端子選択中のみ使用可能	
	I/O(I)	設定の新規作成(N)	I/Oの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能	
		設定の複製(C)	I/Oの設定パターン( 1)を複製します。 I/Oの設定を選択中のみ使用可能	
		設定の削除(D)	I/Oの設定パターン( 1)を削除します。 I/Oの設定を選択中のみ使用可能	
		設定の修正(M)	I/Oに関する設定の修正を行います。 I/Oの設定を選択中のみ使用可能	
		ポートの設定(P)	I/Oの設定パターン( 1)にポートを割り当てます。 I/Oの設定を選択中のみ使用可能	
		ポートの削除(L)	I/Oの設定パターン( 1)からポートの割り当てを解除します。 ポート選択中のみ使用可能	
	タイマ (T)	設定の新規作成(N)	タイマモード(T)	タイマモードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能
			イベントカウンタモード(E)	イベントカウンタモードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能
			パルス幅変調モード(M)	パルス幅変調モードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能
			パルス周期測定モード(P)	パルス周期測定モードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能
			パルス幅測定モード(W)	パルス幅測定モードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能
			インプットキャプチャモード(I)	インプットキャプチャモードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
			アウトプットコンペアモード(O)	アウトプットコンペアモードの設定パターン( 1)を新規作成します。 プロジェクトを開いているときのみ使用可能( 2)
		設定の複製(C)	タイマの設定パターン( 1)を複製します。 タイマの設定を選択中のみ使用可能	
		設定の削除(D)	タイマの設定パターン( 1)を削除します。 タイマの設定を選択中のみ使用可能	
		設定の修正(M)	タイマに関する設定の修正を行います。 タイマの設定を選択中のみ使用可能	
		タイマの設定(T)	タイマの設定パターン( 1)にタイマの種類を設定します。 タイマの設定を選択中のみ使用可能	
	タイマの削除(L)	タイマの設定パターン( 1)からタイマの設定を解除します。 タイマ選択中のみ使用可能		





トップメニュー	階層メニュー	説明	
	INT(N)	設定の新規作成(N)	外部割り込みの設定パターン(1)を新規作成します。 プロジェクトを開いているときのみ使用可能
		設定の複製(C)	外部割り込みの設定パターン(1)を複製します。 外部割り込みの設定を選択中のみ使用可能
		設定の削除(D)	外部割り込みの設定パターン(1)を削除します。 外部割り込みの設定を選択中のみ使用可能
		設定の修正(M)	外部割り込みに関する設定の修正を行います。 外部割り込みの設定を選択中のみ使用可能
		割り込みの設定(I)	外部割り込みの設定パターン(1)に割り込み種別を割り当てます。 外部割り込みの設定を選択中のみ使用可能
		割り込みの削除(L)	外部割り込みの設定パターン(1)から割り込み種別の割り当てを解除します。 割り込み種別選択中のみ使用可能
表示(V)	ツールバー(T)	ツールバーの表示 / 非表示を切り替えます。	
	新規作成ツールバー(B)	新規作成ツールバーの表示 / 非表示を切り替えます。	
	ステータスバー(S)	ステータスバーの表示 / 非表示を切り替えます。	
	新規設定ウィンドウ(N)	新規設定ウィンドウの表示 / 非表示を切り替えます。	
	生成ファイル情報ウィンドウ(F)	生成ファイル情報ウィンドウの表示 / 非表示を切り替えます。	
	生成ファイル情報ウィンドウの文字の大きさ(C)	生成ファイル情報ウィンドウの文字の大きさを切り替えます。 大、中、小から選択できます。 プロジェクトを開いているときのみ使用可能	
ツール(T)	設定(S)	生成ファイルを開くエディタの設定をします。	
	オプション(O)	未対応。	
	ファイルをHEWプロジェクトに登録(R)	生成ファイルをHEWプロジェクトに登録します。	
	出力関数一覧を表示する(D)	出力関数の一覧をCSV形式のファイルで表示します。	
	出力関数一覧を最新の状態にする(P)	出力関数の一覧を最新の状態に更新します。	
ウィンドウ(W)	-	未対応。	
ヘルプ(H)	PDGのバージョン情報(A)	Peripheral Driver Generatorのバージョン情報を表示します。	








- 1 「設定パターン」とは：周辺I/Oの設定内容のことを示します。
- 2 マイコンによっては使用できません。

## 1.9 ツールバー

表 2-2 にツールバーの一覧を示します。

表 1.6 ツールバー一覧

ボタン名称	アイコン	動作	使用可能なとき
新規プロジェクト		プロジェクトを新規に作成します。	随時
開く		既存のプロジェクトを開きます。	随時

ボタン名称	アイコン	動作	使用可能なとき
保存		プロジェクトを上書き保存します。	プロジェクトを開いているとき
プロジェクトコンバート		既存プロジェクトのCPUを変更した新しいプロジェクトを作成します。	プロジェクトを開いているとき マイコンによってはコンバートを使用できません
一括ソース生成		資源が設定されているすべての設定に対してソースファイルを生成します。	周辺機能I/O設定完了時
出力関数一覧を表示		出力した関数の一覧を表示します。	一括ソース生成実行後
出力関数一覧を更新		出力した関数の一覧を最新の状態にします。	一括ソース生成実行後
ヘルプ		Peripheral Driver Generator のソフトバージョンを表示します。	随時
CPU設定の修正		CPUに関する設定の修正を行います。	プロジェクトを開いているとき
シリアル同期モード設定の新規作成		シリアル同期の設定パターンを新規作成します。	プロジェクトを開いているとき
シリアル非同期モード設定の新規作成		シリアル非同期の設定パターンを新規作成します。	プロジェクトを開いているとき
A/D単発モード設定の新規作成		A/D単発モードの設定パターンを新規作成します。	プロジェクトを開いているとき
A/D繰り返しモード設定の新規作成		A/D繰り返しモードの設定パターンを新規作成します。	プロジェクトを開いているとき
A/D単掃引モード設定の新規作成		A/D単掃引モードの設定パターンを新規作成します。	プロジェクトを開いているとき
A/D繰り返し掃引0モード設定の新規作成		A/D繰り返し掃引0モードの設定パターンを新規作成します。	プロジェクトを開いているとき
A/D繰り返し掃引1モード設定の新規作成		A/D繰り返し掃引1モードの設定パターンを新規作成します。	プロジェクトを開いているとき
A/D同時サンプル掃引モード設定の新規作成		A/D同時サンプル掃引モードの設定パターンを新規作成します。	プロジェクトを開いているとき
A/D遅延トリガ0モード設定の新規作成		A/D遅延トリガ0モードの設定パターンを新規作成します。	プロジェクトを開いているとき
A/D遅延トリガ1モード設定の新規作成		A/D遅延トリガ1モードの設定パターンを新規作成します。	プロジェクトを開いているとき
2チャンネルスキャンモード設定の新規作成		2チャンネルスキャンモードの設定パターンを新規作成します。	プロジェクトを開いているとき
2チャンネル連続スキャンモード設定の新規作成		2チャンネル連続スキャンモードの設定パターンを新規作成します。	プロジェクトを開いているとき
4チャンネルスキャンモード設定の新規作成		4チャンネルスキャンモードの設定パターンを新規作成します。	プロジェクトを開いているとき
4チャンネル連続スキャンモード設定の新規作成		4チャンネル連続スキャンモードの設定パターンを新規作成します。	プロジェクトを開いているとき
I/O設定の新規作成		I/Oの設定パターンを新規作成します。	プロジェクトを開いているとき
タイマ タイマモード設定		タイマ タイマモードの設定パターンを新規	プロジェクトを開いているとき

ボタン名称	アイコン	動作	使用可能なとき
の新規作成		作成します。	
タイマ イベントカウンタモード設定の新規作成		タイマ イベントカウンタモードの設定パターンを新規作成します。	プロジェクトを開いているとき
タイマ パルス幅変調モード設定の新規作成		タイマ パルス幅変調モードの設定パターンを新規作成します。	プロジェクトを開いているとき
タイマ パルス周期測定モード設定の新規作成		タイマ パルス周期測定モードの設定パターンを新規作成します。	プロジェクトを開いているとき
タイマ パルス幅測定モード設定の新規作成		タイマ パルス幅測定モードの設定パターンを新規作成します。	プロジェクトを開いているとき
タイマ インพุットキャプチャモード設定の新規作成		タイマ インพุットキャプチャモードの設定パターンを新規作成します。	プロジェクトを開いているとき
タイマ アウトプットコンペアモード設定の新規作成		タイマ アウトプットコンペアモードの設定パターンを新規作成します。	プロジェクトを開いているとき
外部割り込み設定の新規作成		外部割り込みの設定パターンを新規作成します。	プロジェクトを開いているとき

アイコンによっては使用できません。

## 2. 使用前の準備

インストール、PDG でのエディタの設定および HEW との連動のための各種設定を行います。なお、HEW の画面イメージはご使用のバージョンによって異なることがあります。

### 2.1 インストール

インストーラ起動後、インストーラの手順に従ってインストールしてください。インストールは、Administrator 権限で行ってください。



図 2.1 インストーラの起動画面

### 2.2 エディタの設定

生成ファイル情報ウィンドウから、プロジェクトで生成したソースファイルを開くときに使用するエディタを任意に設定することができます。

「ツール」 - 「設定」メニューを選択して、「設定」ダイアログボックスを開きます。

ソースファイルを開くときに使用したいエディタのプログラム名を指定します。

プログラムの起動時のパラメータを、そのプログラムの規定に従って指定します。パラメータで、ファイル名にあたる部分は「%F」、行番号にあたる部分は「%L」に置き換えて指定します。必ずしも指定する必要はありません。

「OK」でダイアログボックスを閉じて、設定は完了です。

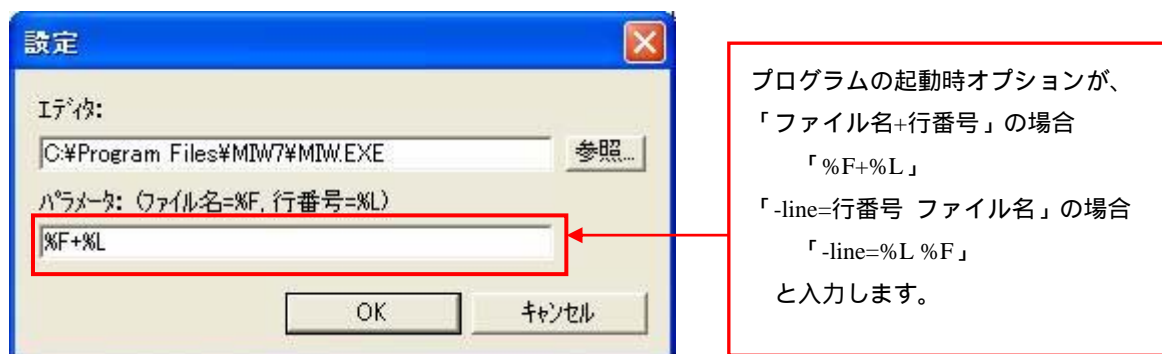


図 2.2 エディタの設定ダイアログボックス

### 2.3 HEWへのPDGの登録

PDG を HEW のメニューに登録し、HEW のメニューから PDG を起動できるようにします。

HEW を立ち上げます。既に立ち上がっている場合は、全てのワークスペースを終了してください。

ようこそダイアログボックスの「アドミニストレーション(A)...」を選択します

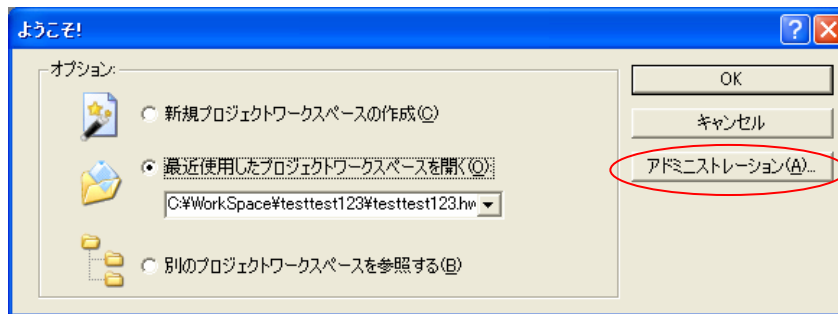


図 2.3 HEWのようこそダイアログボックス

既に HEW を起動している場合は、ツールメニューの「アドミニストレーション(A)...」を選択します。

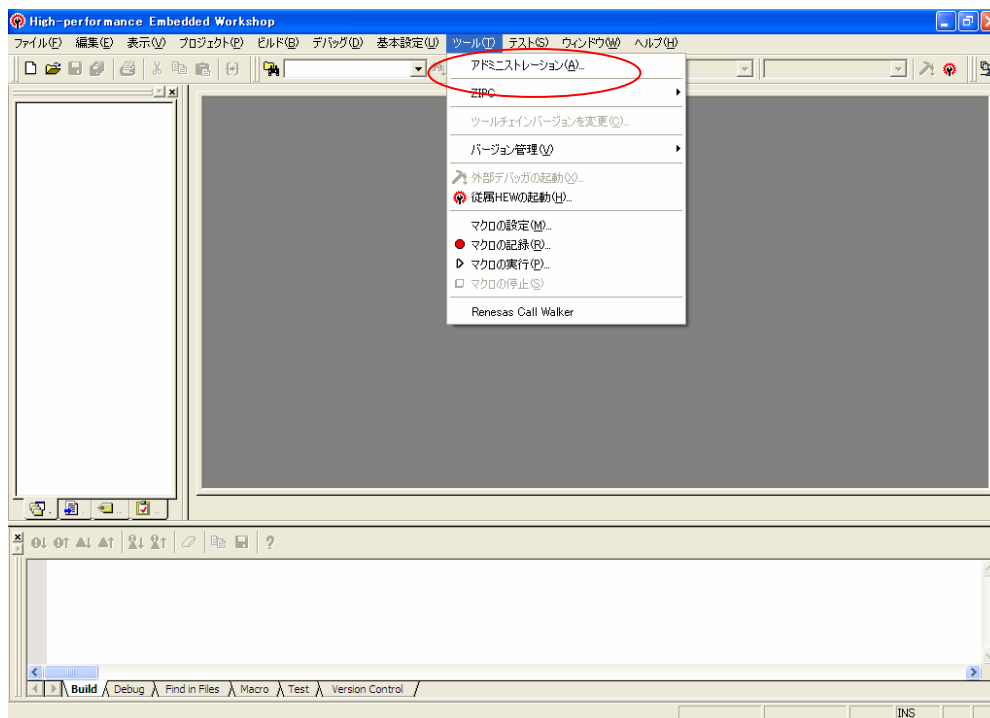


図 2.4 HEWのツールメニュー

登録ボタンを押します。

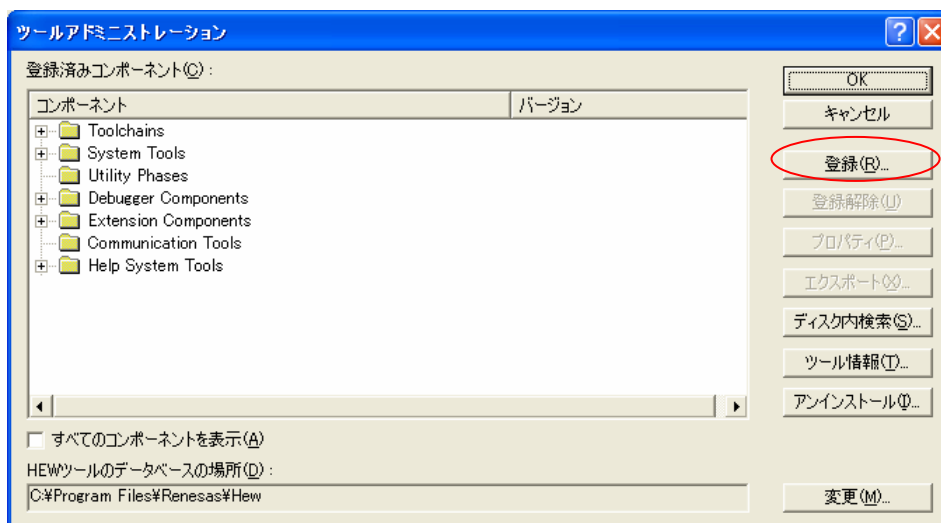


図 2.5 HEWのアドミニストレーションダイアログボックス

PDG をインストールしたディレクトリにある “PDG.hrf”ファイルを選択します。PDG のデフォルトインストールディレクトリは、”C:\Renesas\PDG”です

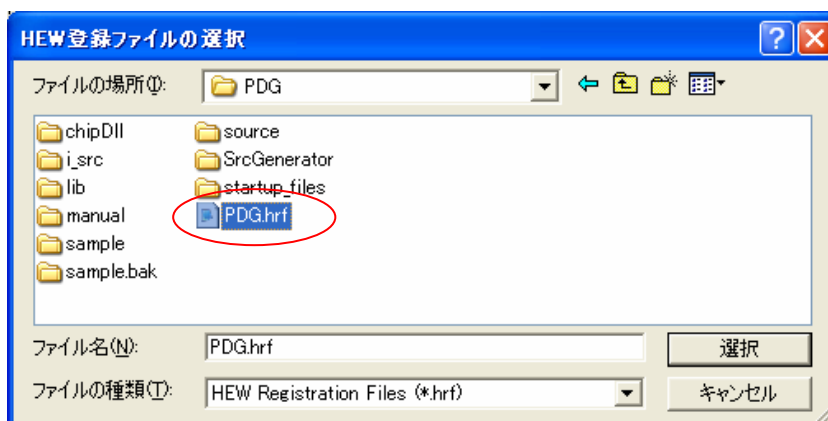


図 2.6 HEW登録ファイルの選択ダイアログボックス

アドミニストレーションダイアログボックスで、System Tools に PDG が登録されていることを確認します。

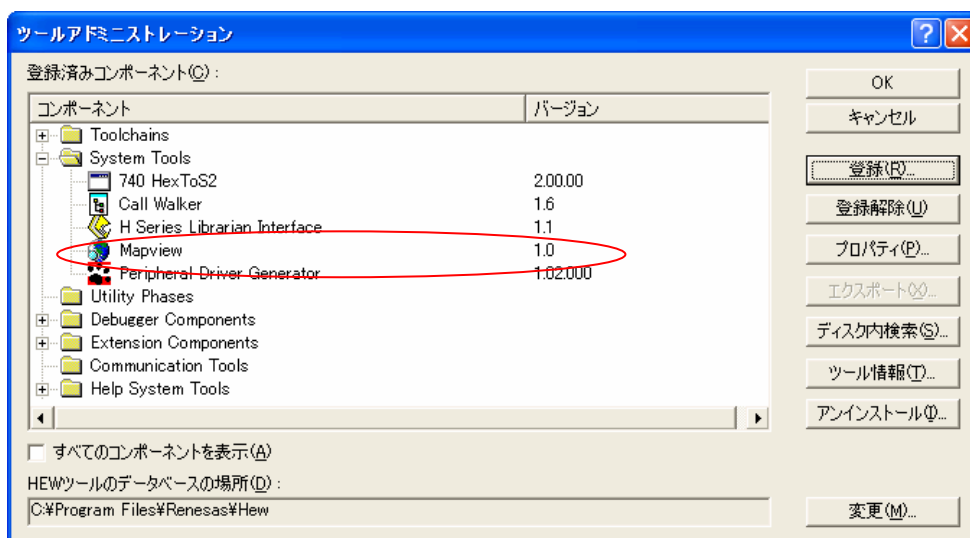


図 2.7 HEW登録ファイルの選択ダイアログボックス

アドミニストレーションダイアログボックスを、OK ボタンで閉じてください。

## 2.4 HEWのHewTargetServer設定

PDG で生成したソースを HEW に登録するためには、HEW の HewTargetServer が正しく設定されている必要があります。以下のように設定してください。

ツールメニューの「アドミニストレーション(A)...」を選択します。

Extension ComponentsにHewTargetServerのバージョンが1.05.00かどうかを確認します。

1.05.00より古いバージョンが表示されている場合は、HewTargetServerを選択した状態で登録解除ボタンを押して登録を解除してください。

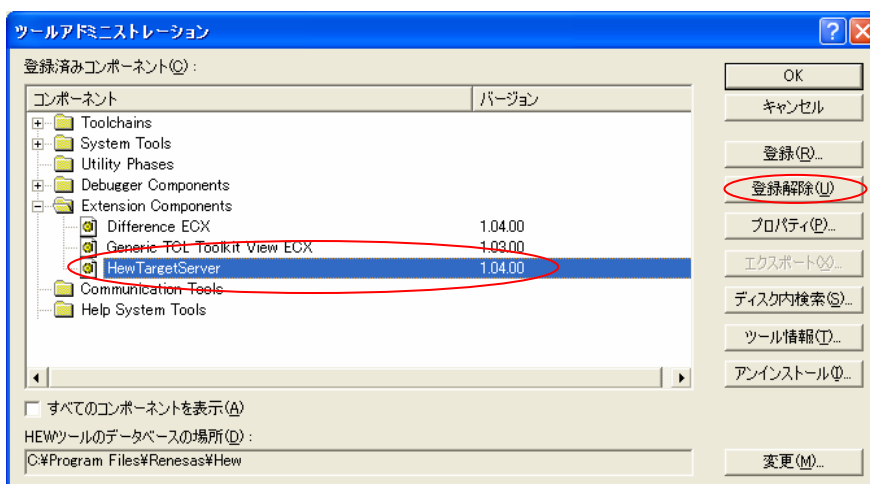


図 2.8 HEWのアドミニストレーションダイアログボックス

ツールアドミニストレーションダイアログボックスの「ディスク内検索(S)...」ボタンを押します。

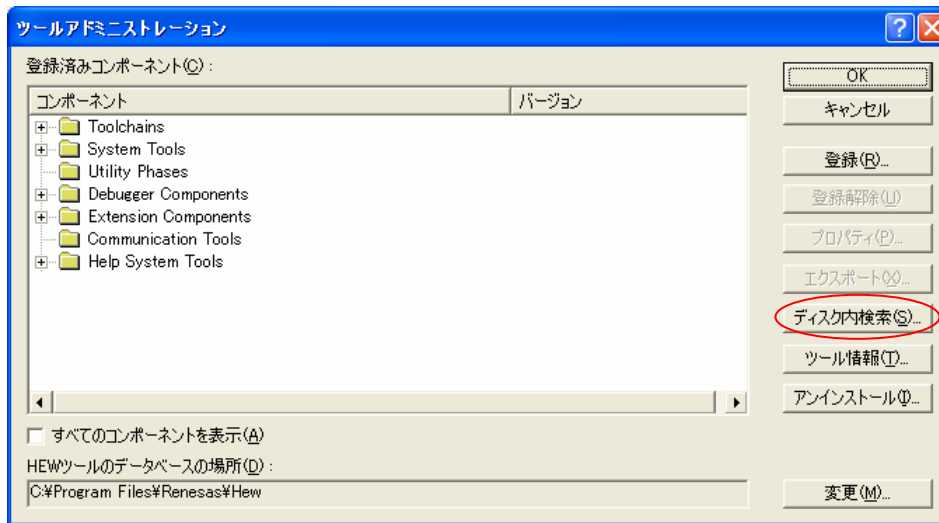


図 2.9 HEWのアドミニストレーションダイアログボックス

コンポーネントのディスク内検索ダイアログボックスで、HEWをインストールしているディレクトリ下で、HewTargetServer を検索します。開始ボタンを押してください。

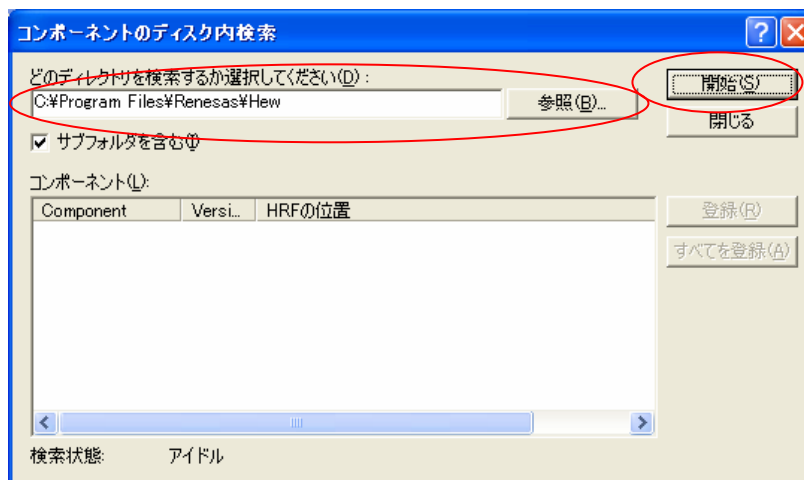


図 2.10 HEWのコンポーネントのディスク内検索ダイアログボックス



見つかったコンポーネント一覧で、HewTargetServer 1.05.00を選択して、登録ボタンを押してください。

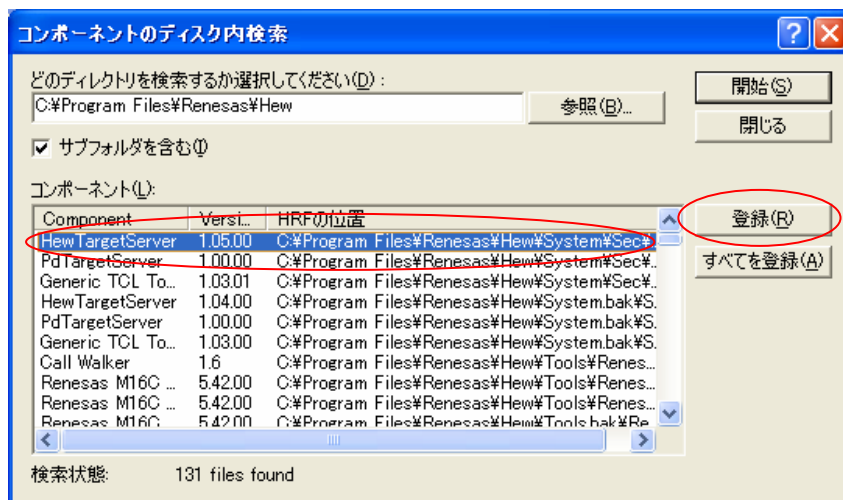


図 2.11 HEWのコンポーネントのディスク内検索ダイアログボックス

コンポーネントのディスク内検索ダイアログボックスを、閉じるボタンで閉じてください。  
 アドミニストレーションダイアログボックスを、OKボタンで閉じてください。  
 HEW インストールフォルダにある REGISTERSERVER.bat を実行してください。  
 デフォルトインストールディレクトリの場合は、次の位置になります。

C:\Program Files\Renesas\Hew\REGISTERSERVER.bat

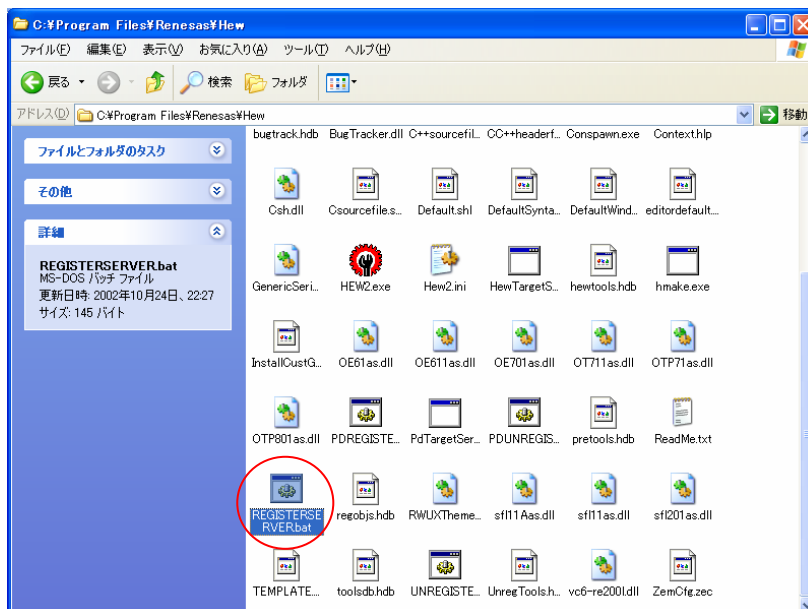


図 2.12 HEWをインストールしたディレクトリの表示例

## 3. PDGの操作方法

### 3.1 PDGを使用したアプリケーションの開発

PDG は、周辺 I/O モジュールへの設定された内容を反映した関数群を C ソースファイルで生成します。

PDG が生成した関数を呼び出して行くことで、周辺 I/O モジュールを操作するアプリケーションを開発することができます。

PDG を使用したアプリケーションの開発の概要を次に示します。

**アプリケーション開発に HEW でワークスペースを作成**  
HEW上で、新規ワークスペース作成などで開発するアプリケーション用のワークスペースを作成します。

**ドライバ開発に PDG でプロジェクトを作成**  
PDG上で、使用するマイコンを選択して、プロジェクトを作成します。

**周辺 I/O モジュールの設定**  
PDGで、作成したプロジェクトで、使用する周辺 I/Oモジュールを設定します。  
CPUの設定から行ってください。

**ソースの生成とワークスペースへソースの登録**  
PDGで、使用する周辺モジュールの設定が終わったら、ソースを一括生成します。  
生成が終わったら、PDGからHEWの作成したワークスペースへ、ソースを登録します。

**アプリケーションの作成**  
PDGが生成したソースに記述された周辺 I/Oモジュールを操作する関数を、アプリケーションから適所で呼び出します。なお、操作関数を呼び出す場合は、あらかじめPDGが生成したヘッダファイルをインクルードしておく必要があります。

**ビルド**  
HEWで、アプリケーションをビルドします。なお、ビルドを行う前に、次のオプションを設定する必要があります。なお、ライブラリファイル指定はHEW V.4.02以降では自動で設定されます。

- ・ PDGが生成したヘッダファイルを参照するためのディレクトリパス指定(-Iオプション)
- ・ APIライブラリをリンクするためのライブラリファイル指定(-Lオプション)

PDGが生成した操作関数において、ビルドでエラーが発生した場合は、操作関数の呼び出しを確認してください。

**デバッグ**  
HEWで、ビルドしたアプリケーションをデバッグします。

**評価**  
アプリケーションの動作が期待されたものであるか、評価します。

### 3.2 PDGの操作の流れ

PDG の操作方法を説明します。

周辺 I/O モジュールの機能の使用方法を設定してから、ソースを生成し、そのソースを使用してドライバの開発を行うまでの流れは以下の通りです。

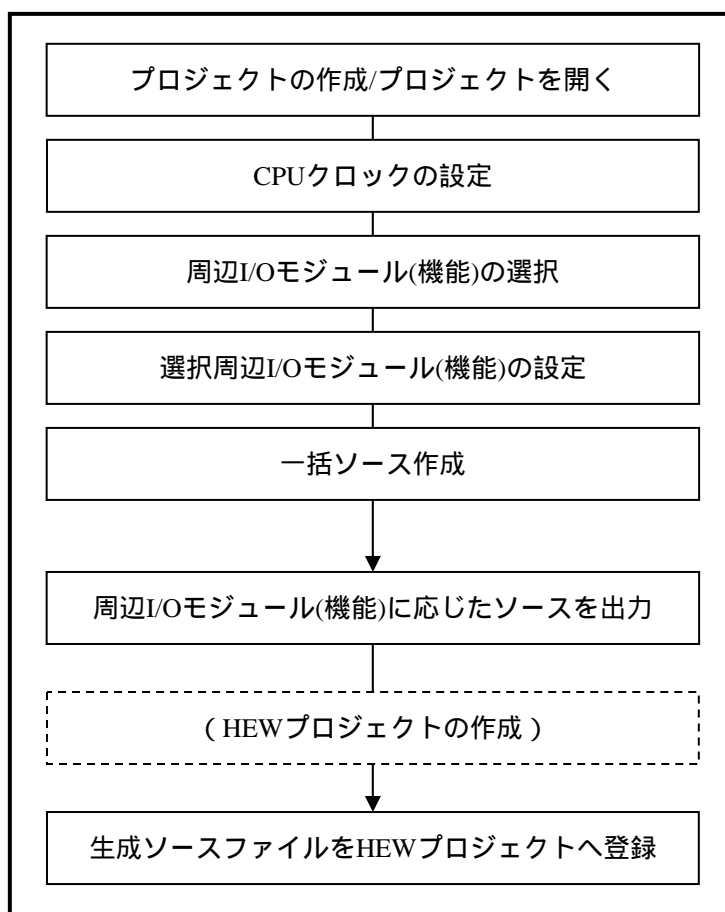


図 3.1 PDGの操作の流れ

### 3.3 プロジェクトの作成/プロジェクトを開く

#### 3.3.1 プロジェクトの新規作成

以下の手順で、新規プロジェクトを作成します。

「ファイル」 - 「プロジェクト新規作成」メニューを選択して、「新規作成」ダイアログボックスを開きます(図 3-2 参照)。



図 3.2 プロジェクトの新規作成ダイアログボックス

作成するプロジェクト名、そのプロジェクトを保存するディレクトリを入力します。使用する CPU 機種の種類をそれぞれ選択します(表 3-1 参照)。

表 3.1 サポートしているマイコンの一覧

シリーズ	グループ	型名
M16C/Tiny	M16C/28	M30280F6, M30280F8, M30280FA M30280FC, M30281F6, M30281F8 M30281FA, M30281FC
	M16C/28B	M30280FCB, M30281FCB
	M16C/29	M30290FA, M30290FC, M30291FA, M30291FC
H8/300H Tiny	H8/3687	HD64F3687, HD64F3684
	H8/36077	HD64F36077, HD64F36074
	H8/36049	HD64F36049
	H8/36109	HD64F36109
R8C/Tiny	R8C/13	R5F21132, R5F21133, R5F21134

シリーズ	グループ	型名
	R8C/22	R5F21226, R5F21227, R5F21228 R5F2122A, R5F2122C
	R8C/23	R5F21236, R5F21237, R5F21238 R5F2123A, R5F2123C
	R8C/24	R5F21244, R5F21245, R5F21246, R5F21247, R5F21248
	R8C/25	R5F21254, R5F21255, R5F21256, R5F21257, R5F21258
	R8C/26	R5F21262, R5F21264, R5F21265 R5F21266
	R8C/27	R5F21272, R5F21274, R5F21275 R5F21276,
	R8C/28	R5F21282, R5F21284
	R8C/29	R5F21292, R5F21294
	R8C/2A	R5F212A7, R5F212A8, R5F212AA R5F212AC
	R8C/2B	R5F212B7, R5F212B8, R5F212BA R5F212BC
	R8C/2C	R5F212C7, R5F212C8, R5F212CA R5F212CC
	R8C/2D	R5F212D7, R5F212D8, R5F212DA R5F212DC
M16C/60	M16C/62P	M30622F8PFP, M30622F8PGP M30623F8PGP, M30620FCPFP M30620FCPGP, M30621FCPGP M3062LFGPFP, M3062LFGPGP M30625FGPGP, M30626FHPFP M30626FHPGP, M30627FHPGP M30626FJPPFP, M30626FHPGP M30627FJPPGP
SH/Tiny	SH7125	R5F71253N50FP, R5F71253D50FP, R5F71253N50FA, R5F71253D50FA, R5F71252N50FP, R5F71252D50FP, R5F71252N50FA, R5F71252D50FA

「OK」ボタンを選択して、新規プロジェクトを作成します。

新規作成直後は、自動的に CPU クロックの設定画面が開きます。続いて CPU クロックの設定を行ってください。

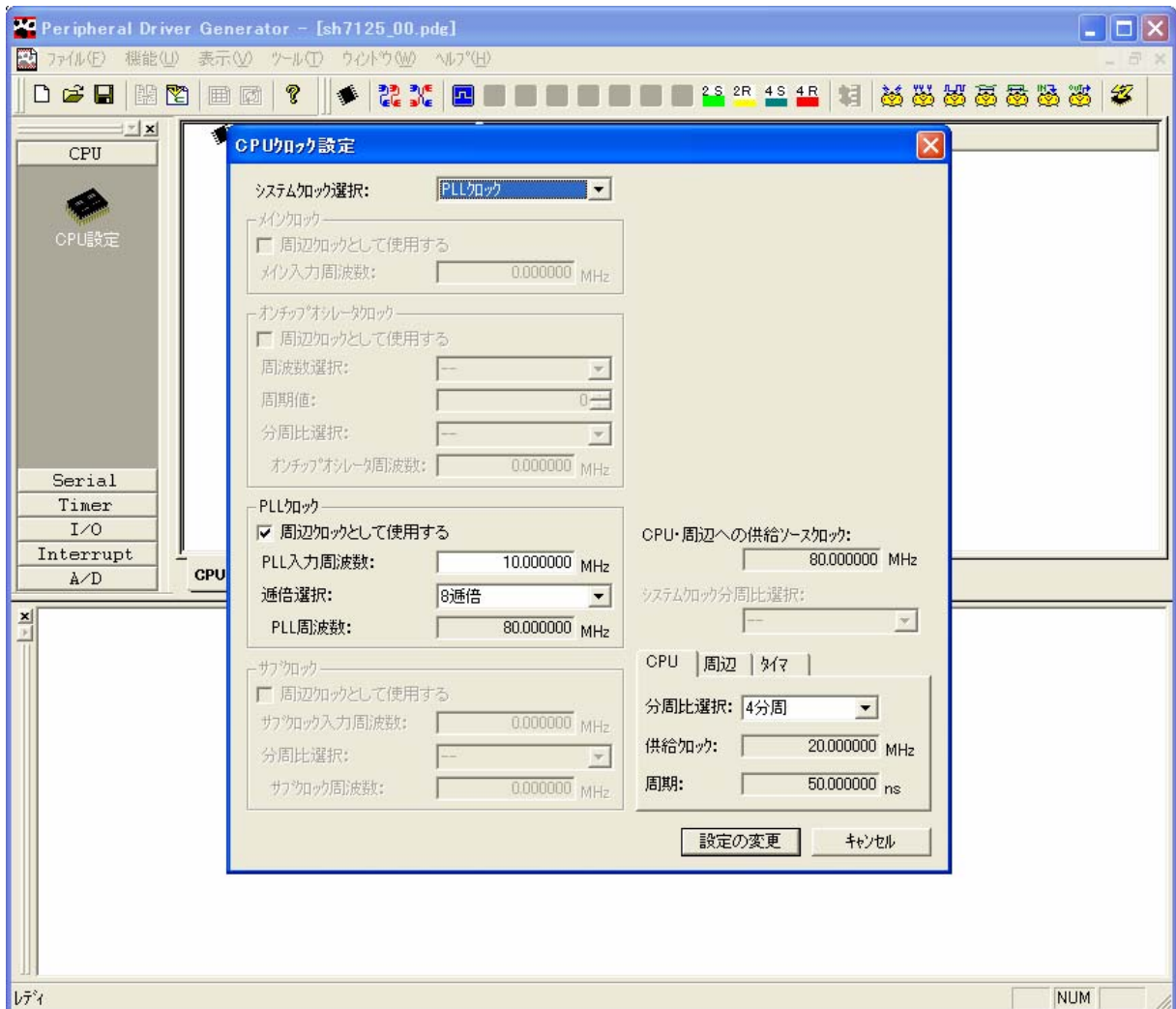


図 3.3 新規作成プロジェクトの画面

### 3.3.2 既存プロジェクトを開く

以下の手順で、既存プロジェクトを開きます。

- 「ファイル」 - 「開く」メニューを選択して、「ファイルを開く」ダイアログボックスを開きます。
- 開きたいプロジェクトを選択し、「開く」ボタンをクリックするか、ファイル名をダブルクリックします。
- 選択したプロジェクトが開きます。

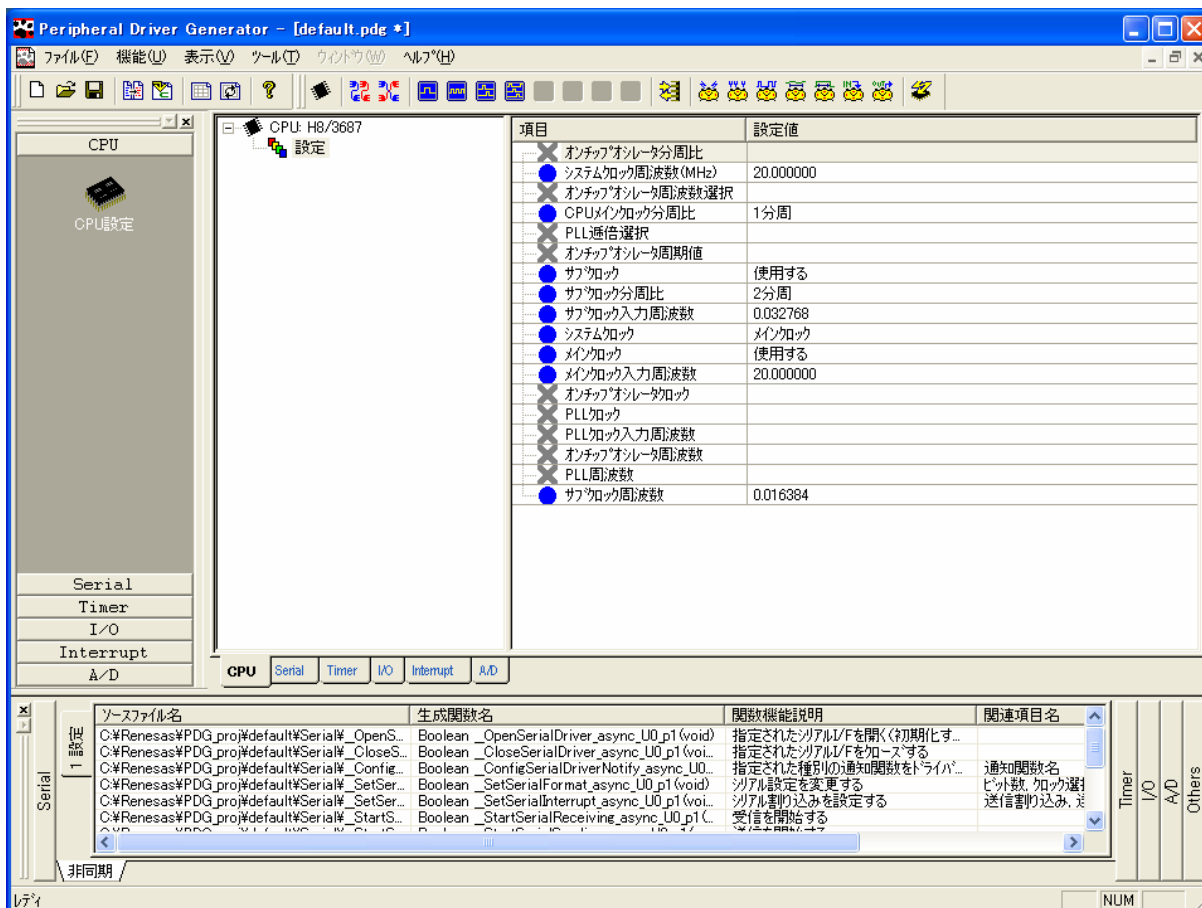


図 3.4 既存プロジェクトの画面

### 3.3.3 CPUクロックの設定

プロジェクトを新規作成した場合、「CPUクロック設定」ダイアログボックスが自動的に開きますので、CPUクロックに関する項目の設定を行います。

The screenshot shows the 'CPUクロック設定' dialog box with the following settings:

- システムクロック選択: PLLクロック
- メインクロック
  - 周辺クロックとして使用する
  - メイン入力周波数: 0.000000 MHz
- オンチップオシレータクロック
  - 周辺クロックとして使用する
  - 周波数選択: --
  - 周期値: 0
  - 分周比選択: --
  - オンチップオシレータ周波数: 0.000000 MHz
- PLLクロック
  - 周辺クロックとして使用する
  - PLL入力周波数: 10.000000 MHz
  - 逡倍選択: 8逡倍
  - PLL周波数: 80.000000 MHz
- サブクロック
  - 周辺クロックとして使用する
  - サブクロック入力周波数: 0.000000 MHz
  - 分周比選択: --
  - サブクロック周波数: 0.000000 MHz
- CPU・周辺への供給ソースクロック: 80.000000 MHz
- システムクロック分周比選択: --
- CPU | 周辺 | タイマ
  - 分周比選択: 4分周
  - 供給クロック: 20.000000 MHz
  - 周期: 50.000000 ns
- 設定の変更 | キャンセル

図 3.5 CPUクロック設定ダイアログボックス



## 3.4 周辺I/Oモジュールの選択・設定

### 3.4.1 周辺I/Oモジュールの設定パターンの新規作成

以下の手順で、周辺 I/O モジュールの設定パターンを新規作成します。

設定パターン新規作成ウィンドウから、制御対象となる周辺 I/O モジュールのそれぞれのモードボタン(図 3-6 参照)を選択するか、「機能」 - 「シリアル、A/D、I/O、タイマ、INT」 - 「設定の新規作成」メニューからそれぞれのモードを選択します。

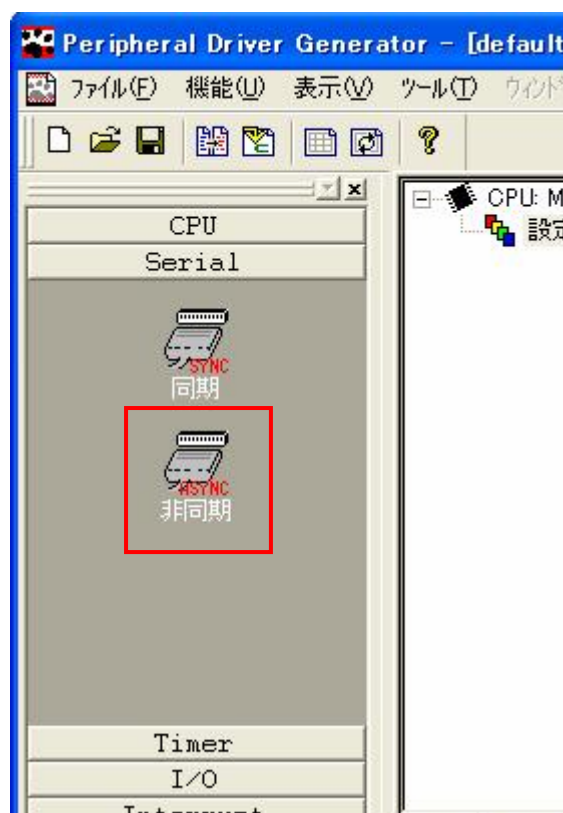


図 3.6 設定パターン新規作成ウィンドウ

各周辺 I/O モジュールの機能を設定後(図 3-7 参照)、「設定」ボタンを選択すると、設定した項目の内容(設定パターン)をメインウィンドウ右側のリストに表示します(図 3-8 参照)。

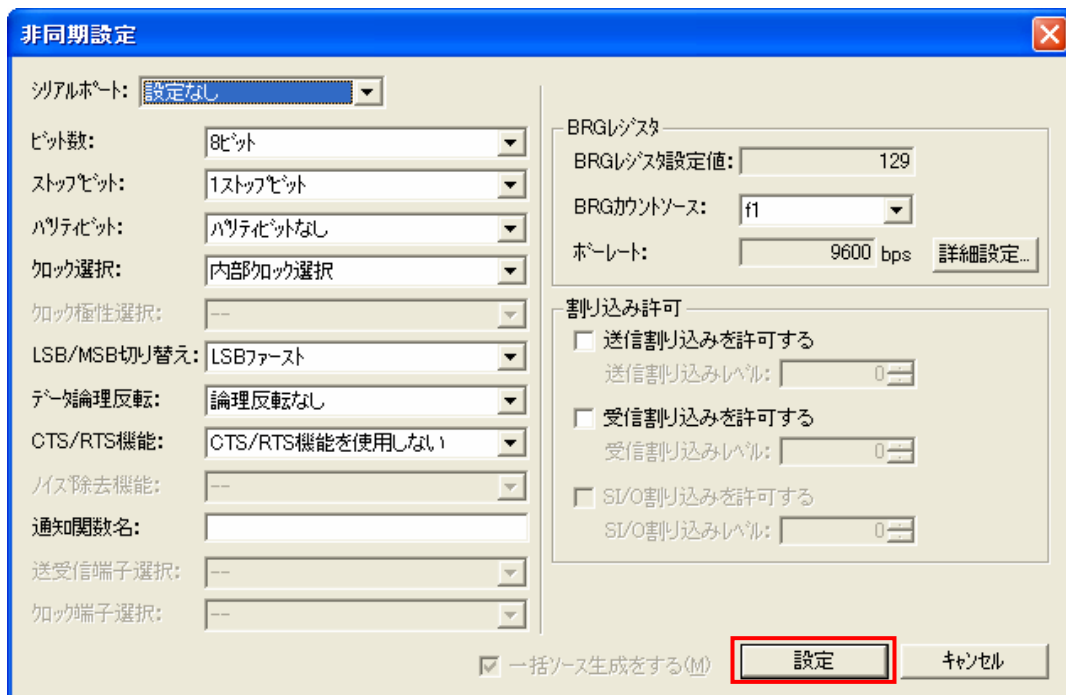


図 3.7 設定パターン新規作成ウィンドウ

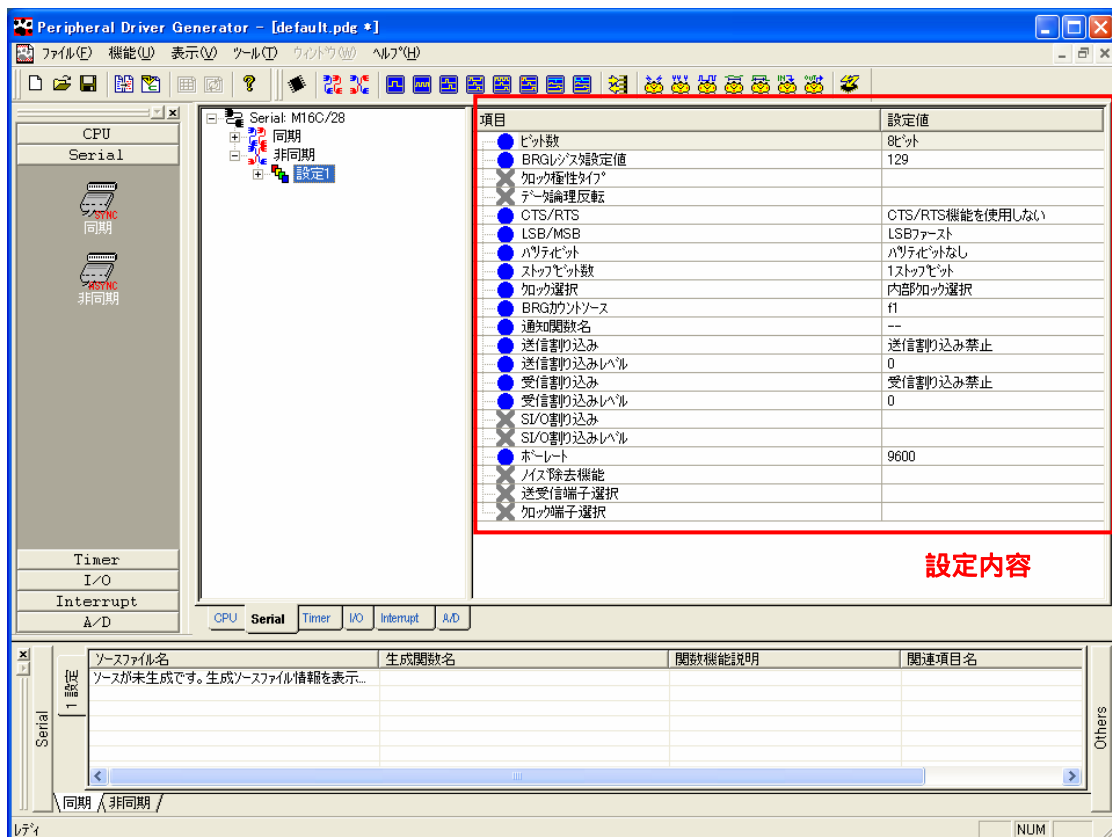


図 3.8 設定パターン表示ウィンドウ

### 3.4.2 周辺I/Oモジュールの設定パターンの変更

既存の設定パターンの変更は、以下の手順で行います。

メイン画面左側ツリー上で、「設定[n]」をダブルクリックするか、メイン画面右側リスト上で設定項目名をダブルクリックします。又は、「機能」 - 「CPU、シリアル、A/D、I/O、タイマ、INT」 - 「設定の修正」メニューを選択します。

対象周辺 I/O モジュールの機能ごとに対応した、設定ダイアログボックスが開くので、設定を変更します。

「設定の変更」ボタンを選択し、ダイアログボックスを閉じると、メイン画面右側のリストに変更を反映します。

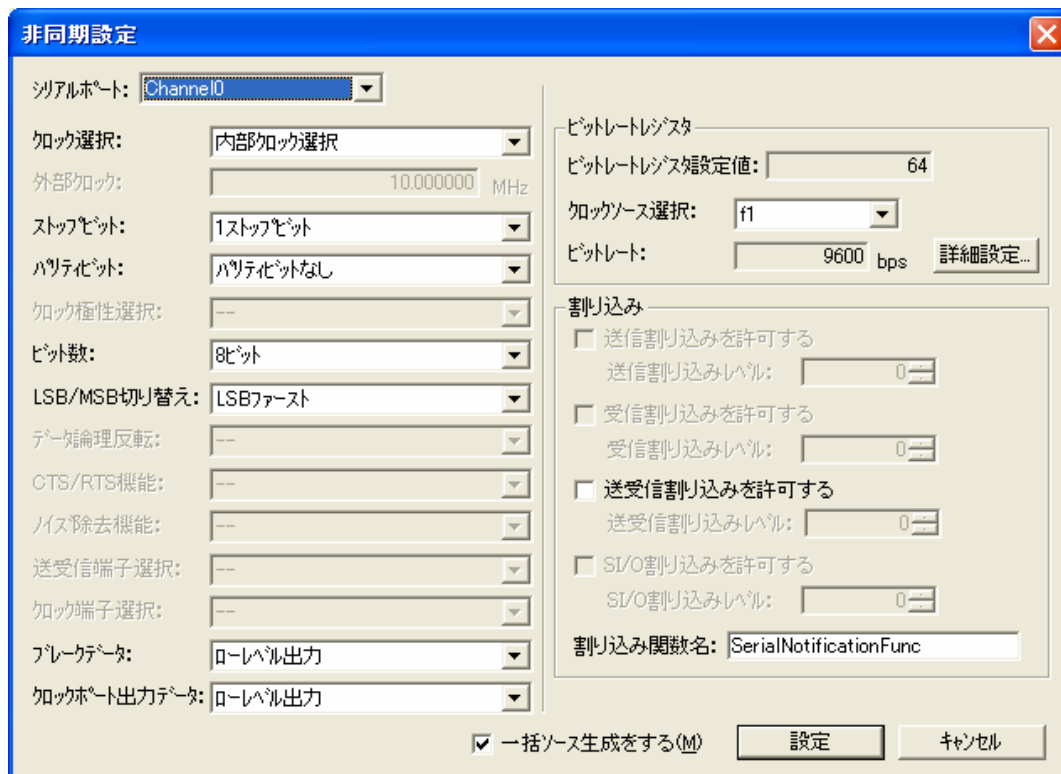


図 3.9 指定した機能に対応する設定ダイアログボックス

### 3.4.3 周辺I/Oモジュールの設定パターンの複製

既存の設定パターンを複製します。設定に資源が割り当てられているときは、資源の設定も複製します。

メイン画面左側のツリー上で、「設定」を選択しているときのみ使用できます。

メイン画面左側のツリー上で「設定[n]」を選択後、「機能」 - 「シリアル、A/D、I/O、タイマ、INT」 - 「設定の複製」メニューを選択するか、「設定[n]」を右クリックし「設定の複製」メニューを選択します。

モード表示の最後尾に、選択中の設定パターンの複製ができます。

### 3.4.4 周辺I/Oモジュールの設定パターンの解除

既存の設定パターンを削除します。設定に資源が割り当てられているときは、資源の設定もすべて削除します。

メイン画面左側のツリー上で、「設定」を選択しているときのみ使用できます。

メイン画面左側のツリー上で「設定[n]」を選択後、「機能」 - 「シリアル、A/D、I/O、タイマ、INT」 - 「設定の削除」メニューを選択するか、「設定[n]」を右クリックし「設定の削除」メニューを選択します。

選択していた設定パターンを削除します。

## 3.5 資源の割り当てと解除


### 3.5.1 資源の割り当て

資源(周辺I/Oモジュール)の割り当てがない設定パターンに対して、それぞれの周辺機能に応じて資源を割り当てます。

各設定パターンに割り当てられる資源は一つです。資源が割り当てられるのは、メイン画面左側のツリー上で「設定」を選択しているときのみです。

メイン画面左側のツリー上で「設定[n]」（CPU クロック以外）を選択後、「機能」 - 「シリアル、A/D、I/O、タイマ、INT」 - 「UART 番号の設定、入力グループ・入力端子の設定、ポートの設定、タイマの設定、割り込みの設定」メニューを選択するか、メイン画面左側ツリー上で「設定[n]」を右クリックし「（各資源の）設定」メニューを選択します。

「（各資源の）設定」ダイアログボックスで、選択中の設定パターンに割り当てたい資源を選択します。

「OK」ボタンでダイアログボックスを閉じると、選択中の設定パターンに資源を割り当てます。このとき、資源を割り当てることにより、無効になる項目があればメッセージを表示します。また、設定値を変える必要のあるものは、資源割り当て後、設定のリスト表示で  アイコンになっていますので、確認してください。

### 3.5.2 割り当てた資源の解除

「（各資源の）設定」で割り当てた資源を解除します。

メイン画面左側のツリー上で、割り当てた資源を選択しているときのみ使用できます。

メイン画面左側のツリー上で「資源名[x]」を選択後、「機能」 - 「シリアル、A/D、I/O、タイマ、INT」 - 「UART 番号の削除、入力グループ・入力端子の削除、ポートの削除、タイマの削除、割り込みに削除」メニューを選択するか、メイン画面左側のツリー上で「資源名[x]」を右クリックし「（各資源の）削除」メニューを選択します。

選択していた資源を削除します。

### 3.6 一括ソース生成

現在開いているプロジェクトの機能設定の内容に合わせて、ソースコードを生成します。

作成した何れかの設定パターンのうち、ひとつでも資源が設定されている場合に選択できます。

「ファイル」 - 「一括ソース生成」メニューを選択します。

ソースファイルの生成が実行され、現在開いているプロジェクトファイルと同じディレクトリにソースファイルを生成します。同時に、「生成ファイル情報ウィンドウ」に生成したソースファイルの情報を表示します。

周辺 I/O 機能設定ダイアログの「一括ソース生成を行う」チェックボックスを利用して、設定パターンの作成と連続して行う場合は、設定ダイアログを閉じると同時に自動的にソースファイルを生成します。

生成したすべてのソースファイルを一括で削除する場合は、「ファイル」 - 「一括ソース削除」メニューを選択します。

### 3.7 出力関数情報をCSV形式で見る

PDGで一括生成した関数情報の一覧をCSV形式のファイルで見ることができます。

一括ソース生成を実行後、使用できます。

「ツール」 - 「出力関数一覧を表示する」メニューを選択します。

\*.csv ファイルと関連付けのされているプログラムで、出力関数の一覧を開いて表示します。

### 3.8 出力関数情報を更新する

PDGで一括生成した関数情報一覧のCSVファイルを最新の状態にします。

一括ソース生成を実行後、使用できます ( )。

「ツール」 - 「出力関数一覧を最新の状態にする」を選択します。

出力関数一覧の CSV ファイルを更新します。

関数情報一覧の CSV ファイルは、一括ソース生成実行時に更新します。但し、そのときに CSV ファイルを開いているなどの事情により更新できないことがあります。その場合は、CSV ファイルを一旦閉じた上でこのメニューを使用してください。

### 3.9 生成ファイルのHEWプロジェクトへの登録

#### 3.9.1 登録機能

PDGで一括生成したすべてのソースファイルを、既存のHEWプロジェクトへ自動的に登録します。同時に、PDG生成ソースで使用しているAPIライブラリをライブラリオプションに登録し、割り込み関数のバッティングを防ぐため、登録先のHEWプロジェクトに「intprg.c」ファイルが登録されている場合は、「intprg.c」ファイルをビルドから除外します。

すでに登録されていた「intprg.c」ファイルに独自処理を追加していた場合は、新たに登録した「intprg.c」ファイルへ、独自処理をマニュアルでコピーする必要があります。

### 3.9.2 登録の操作方法

すでにソースを生成している場合に、使用できます。

「ツール」 - 「ファイルを HEW プロジェクトに登録」メニューを選択します。

HEW を起動していない場合は、HEW の起動を確認するメッセージが表示されるので、「はい」を選択し、HEW を起動します。

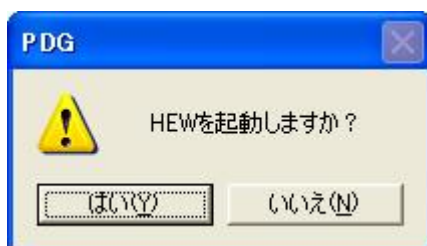


図 3.10 起動確認メッセージ (PDG)

次に、ファイル登録を確認するメッセージが表示されます。

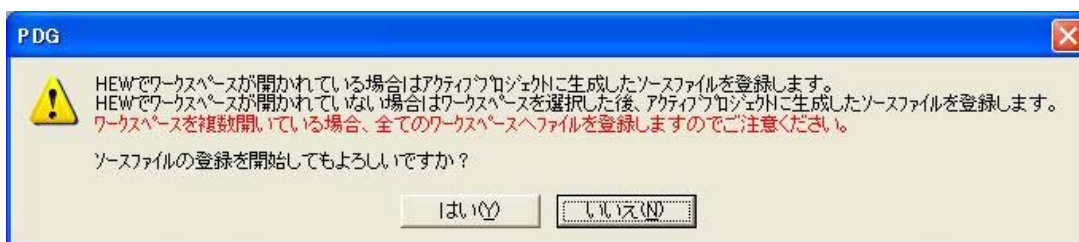


図 3.11 ファイル登録の確認メッセージ (PDG)

- 登録先の HEW ワークスペースが既に開いている場合  
「はい」を選択します。
- 登録先の HEW ワークスペースが既存の場合  
「はい」を選択して、次のステップ「ファイルを開く」に進みます。登録先の HEW ワークスペースを指定します。「開く」を選択して、ワークスペースを開きます。
- 登録先の HEW ワークスペースが未作成の場合  
確認メッセージをそのままに、一旦 HEW の操作に移り、新しく登録先の HEW ワークスペースを作成します。ワークスペースを開いた状態で PDG 操作に戻り、「はい」を選択します。

「ライブラリリンク順序設定ダイアログ」が開くので、リンクの優先順位が高いものを上、低いものを下へ移動します。「OK」を選択すると、HEW プロジェクトへのファイル登録を開始します( )。

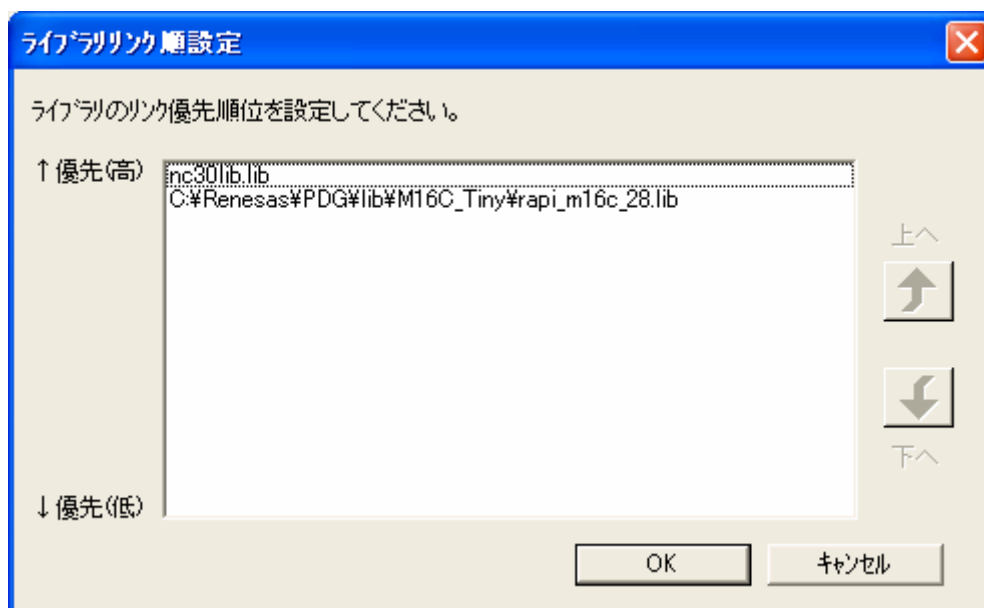


図 3.12 ライブラリリンク順序設定ダイアログ

ファイル登録の確認メッセージにもあるように、複数の HEW ワークスペースが開いている状態では、すべてのアクティブプロジェクトにファイルを登録します。ファイルを登録したくないワークスペースは、ファイル登録実行の前に閉じてください。

ファイルの登録完了メッセージが表示されれば、登録は完了です。

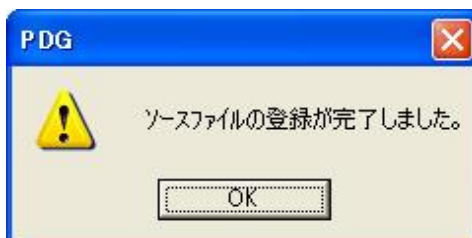


図 3.13 登録完了メッセージ(PDG)

注意：SH7125でHEW V.4.04以前のバージョンを使用している場合、PDGが登録するintprg\_sh7125.cおよびintprg\_sh7125\_pdg.cに対し、HEWのスタートアッププログラムのファイルvect.hへのパスを設定する必要があります。HEW V.4.05以降のバージョンを使用している場合、インクルードパスは自動的に登録されます。

### 3.9.3 登録したソースの登録解除

HEW ヘソースを一度登録すると、PDG からは解除することはできません。

HEW のワークスペースウィンドウのプロジェクトタブで、解除したいソースを選択し、右クリックでポップアップメニューを表示します。ポップアップメニューから、ファイルの削除またはビルドの除外を選択してください。



## 4. クロック、周辺I/Oモジュール設定方法

### 4.1 クロックの設定

プロジェクトを新規作成した場合、最初に[CPU クロック設定]ダイアログボックスが開きます。周辺 IO モジュールの設定を行う前に CPU と周辺モジュールのクロックの設定を行ってください。[CPU クロック設定]ダイアログボックスの設定内容は、周辺 IO モジュールの設定に使用されます。

「CPU クロック設定」ダイアログボックスは、周辺 IO モジュールの設定に必要なクロック情報を入力するためのものです。「CPU クロック設定」ダイアログボックスの設定に対応したソースコードの生成は行いません。クロックに関する初期設定はユーザプログラムで行ってください。

以下に各 CPU でのクロックの設定方法を示します。

#### 4.1.1 M16C/62P、M16C/28、M16C/28B、M16C/29 のクロック設定

M16C/62P、M16C/28、M16C/28B、およびM16C/29 の[CPUクロック設定]ダイアログボックスを 図 4.1.1 に示します。

The screenshot shows the 'CPUクロック設定' dialog box with the following settings:

- システムクロック選択:** メインクロック
- メインクロック**
  - 周辺クロックとして使用する
  - メイン入力周波数: 20.000000 MHz
- オンチップオシレータクロック**
  - 周辺クロックとして使用する
  - 周波数選択: --
  - 周期値: 0
  - 分周比選択: --
  - オンチップオシレータ周波数: 0.000000 MHz
- PLLクロック**
  - 周辺クロックとして使用する
  - PLL入力周波数: 10.000000 MHz
  - 通倍選択: --
  - PLL周波数: 0.000000 MHz
- サブクロック**
  - 周辺クロックとして使用する
  - サブクロック入力周波数: 0.032768 MHz
  - 分周比選択: 1分周
  - サブクロック周波数: 0.032768 MHz
- CPU・周辺への供給ソースクロック:** 20.000000 MHz
- システムクロック分周比選択:** --
- CPU | 周辺**
  - 分周比選択: 1分周
  - 供給クロック: 20.000000 MHz
  - 周期: 50.000000 ns
- Buttons: 設定の変更, キャンセル

図4.1.1 CPUクロック設定ダイアログボックス (M16C/62P,28,28B,29)



## (1) システムクロック選択

CPUクロックとして使用するクロックを選択します。メインクロック、オンチップオシレータクロック、PLLクロックおよびサブクロックから選択可能です。[システムクロック選択]の各項目は、表4.1.1に示すクロック源に対応します。

表4.1.1 M16C/62P、M16C/28、M16C/28B、M16C/29 のクロック源

項目	クロック源
メインクロック	メインクロック発信回路
オンチップオシレータクロック	オンチップオシレータ
PLLクロック	PLL周波数シンセサイザ
サブクロック	サブクロック発信回路

## (2) 各クロックの設定

[メインクロック]、[オンチップオシレータクロック]、[PLLクロック]および[サブクロック]の各項目より[システムクロック選択]で選択したクロックの設定を行います。[システムクロック選択]で指定したCPU供給クロックは周辺クロックとしても使用されます。ただしタイマA、タイマBのカウントソースとして選択可能なfC32はサブクロックのみが供給可能です。CPUクロックとしてサブクロック以外を選択し、fC32を使用する場合はサブクロックの設定も行ってください。サブクロックが設定されていない場合、タイマ設定においてカウントソースとしてfC32を選択できません。

CPUクロックおよび周辺機能クロックとして使用できるクロックの組み合わせを表4.1.2に示します。

表4.1.2 CPUおよび周辺クロック源の組み合わせ(M16C/62P、M16C/28、M16C/28B、M16C/29)

CPUクロック	周辺機能クロック	
	f1～f32	fC32
メインクロック	メインクロック	サブクロック
オンチップオシレータクロック	オンチップオシレータクロック	サブクロック
PLLクロック	PLLクロック	サブクロック
サブクロック	メインクロック、オンチップオシレータクロック、PLLクロックのいずれか	サブクロック

## [メインクロック]

CPUクロックとしてメインクロックを指定した場合と、CPUクロックとしてサブクロックを指定し、fC32を除く周辺機能クロックとしてメインクロックを使用する場合に設定します。

- [周辺クロックとして使用する]：CPU クロックとしてメインクロックを使用する場合は常にチェックされます。CPU クロックとしてサブクロックを指定し、周辺機能クロックとしてメインクロックを使用する場合はチェックしてください。
- [Main 入力周波数]：メインクロックの周波数を設定してください。

## [オンチップオシレータクロック]

CPUクロックとしてオンチップオシレータクロックを指定した場合と、CPUクロックとしてサブクロックを指定し、fC32を除く周辺機能クロックとしてオンチップオシレータクロックを使用する場合に設定します。

- [周辺クロックとして使用する]：CPU クロックとしてオンチップオシレータクロックを使用する場合は常にチェックされます。CPU クロックとしてサブクロックを指定し、fC32を除く周辺機能クロックとしてオンチップオシレータクロックを使用する場合はチェックしてください。
- [周波数選択]：オンチップオシレータ発振周波数を選択してください。

- [周期値]：設定しません。
- [分周比選択]：オンチップオシレータの分周比を選択してください。
- [オンチップオシレータ周波数]：オンチップオシレータ発振周波数と分周比から算出したオンチップオシレータクロックの周波数が表示されます。

#### [PLL クロック]

CPU クロックとして PLL クロックを指定した場合と、CPU クロックとしてサブクロックを指定し、fC32 を除く周辺機能クロックとして PLL クロックを使用する場合に設定します。

- [周辺クロックとして使用する]：システムクロックとして使用する場合は常にチェックされます。CPU クロックとしてサブクロックを指定し、fC32 を除く周辺機能クロックとして PLL クロックを使用する場合はチェックしてください。
- [PLL 入力周波数]：PLL 周波数シンセサイザへの入力(XIN)周波数を設定してください。
- [逡倍選択]：PLL 周波数シンセサイザの逡倍率を選択してください。
- [PLL 周波数]：PLL 入力周波数と逡倍率から算出した PLL クロックの周波数が表示されます。

#### [サブクロック]

CPU クロックとしてサブクロックを指定した場合と、CPU クロックとしてサブクロック以外を指定し、周辺機能クロック fC32 を使用する場合に設定します。

- [周辺クロックとして使用する]：システムクロックとして使用する場合は常にチェックされます。システムクロックとしてサブクロック以外を選択した場合は、タイマ A、タイマ B のカウントソースとして fC32 を使用するためにチェックします。
- [Sub 入力周波数]：サブクロックの周波数は 0.03276MHz 固定です。
- [分周比選択]：1 分周のみ指定可能です。
- [サブクロック周波数]：Sub 入力周波数と分周比から算出したサブクロックの周波数が表示されます。0.03276MHz 固定です。

### (3) システムクロック設定

(1)[システムクロックの選択]で指定したクロックを CPU に供給するための分周率の設定を行います。

- [システムクロック周波数]：(1)[システムクロック選択]で指定したクロックの周波数が表示されます。
- [システムクロック分周比]：設定しません。
- [CPU]、[周辺]タブ：CPU クロックおよび周辺機能クロックの設定を行います。表 4.1.3 に各項目の設定内容を示します。

表4.1.3 システムクロック設定内容 (M16C/62P、M16C/28、M16C/28B、M16C/29)

項目	設定内容	
CPU	分周比選択	CPU に供給するクロックの分周比を選択してください。
	供給クロック	[システムクロック周波数]と[分周比選択]の設定値から算出した、CPU 供給クロックの周波数が表示されます。
	内部周期	[システムクロック周波数]と[分周比選択]の設定値から算出した、CPU 供給クロックの周期が表示されます。
周辺	分周比選択	設定しません。 [システムクロックの選択]で指定したクロックを各分周率で分周したクロックが周辺 IO モジュールに供給されます。各周辺 IO モジュールで使用するクロックは、周辺 IO モジュールの設定で指定してください。 fC32 の周波数は[サブクロック]の設定に従います。
	供給クロック	[システムクロック選択]で指定したクロックの周波数が表示されます。
	内部周期	[システムクロック選択]で指定したクロックの周期が表示されます。

## 4.1.2 R8C/13 のクロック設定

R8C/13 の[CPUクロック設定]ダイアログボックスを 図 4.14.1.2 に示します。

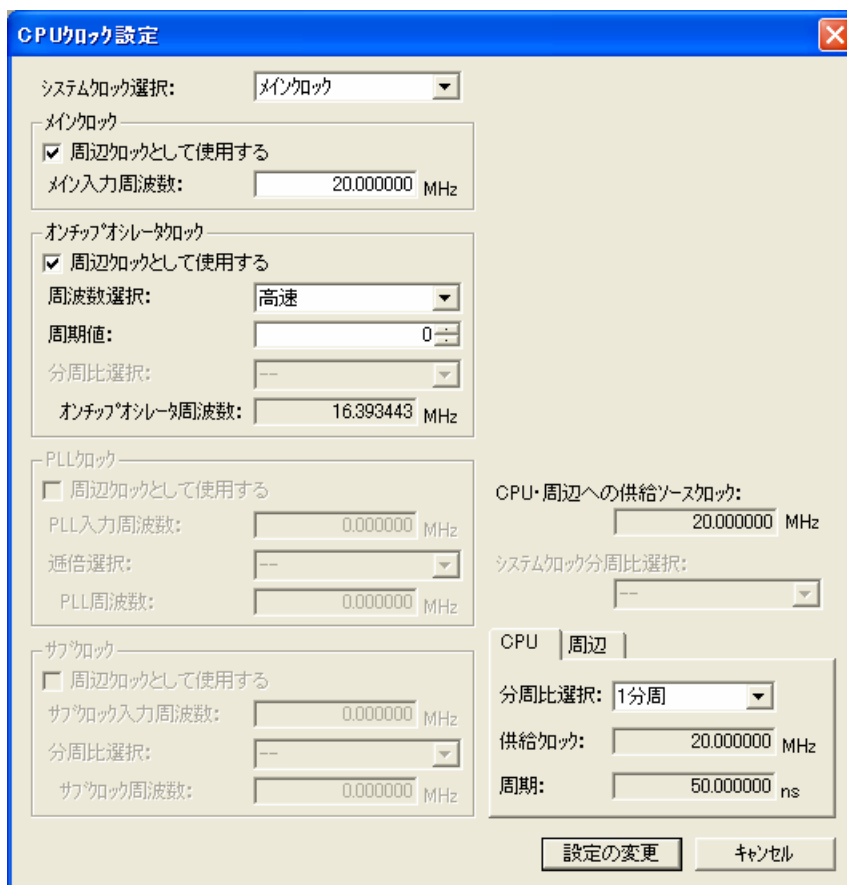


図4.1.2 CPUクロック設定ダイアログボックス (R8C/13)

## (1) システムクロック選択

CPUクロックとして使用するクロックを選択します。メインクロックまたはオンチップオシレータクロックから選択可能です。[システムクロック選択]の各項目は、表4.1.4に示すクロック源に対応します。

表4.1.4 R8C/13のクロック源

項目	クロック源	
メインクロック	メインクロック発信回路	
オンチップオシレータクロック	オンチップオシレータ	高速オンチップオシレータ
		低速オンチップオシレータ

## (2) 各クロックの設定

[メインクロック]、[オンチップオシレータクロック]の各項目より[システムクロック選択]で選択したクロックの設定を行います。[システムクロック選択]で指定したCPU供給クロックは周辺クロックとしても使用されます。ただしタイマカウントソースとして選択可能なfRINGはオンチップオシレータのみが供給可能です。CPUクロックとしてメインクロックを選択し、fRINGを使用する場合はオンチップオシレータクロックの設定も行ってください。オンチップオシレータクロックを設定していない場合、タイマ設定においてカウントソースとしてfRINGを選択できません。

CPUクロックおよび周辺機能クロックとして使用できるクロックの組み合わせを 表表4.1.5に示します。

表4.1.5 CPUおよび周辺クロック源の組み合わせ (R8C/13)

CPUクロック	周辺機能クロック	
	f1~f32,fAD	fRING
メインクロック	メインクロック	オンチップオシレータクロック
オンチップオシレータクロック	オンチップオシレータクロック	オンチップオシレータクロック

#### [メインクロック]

CPUクロックとしてメインクロックを指定した場合に設定します。

- [周辺クロックとして使用する]: CPU クロックとしてメインクロックを使用する場合は常にチェックされます。
- [Main 入力周波数]: メインクロックの周波数を設定してください。

#### [オンチップオシレータクロック]

CPUクロックとしてオンチップオシレータクロックを指定した場合と、CPUクロックとしてメインクロックを指定し、タイマのカウントソースとしてfRINGを使用する場合に設定します。

- [周辺クロックとして使用する]: CPU クロックとしてオンチップオシレータクロックを使用する場合は常にチェックされます。CPU クロックとしてメインクロックを指定し、fRING を使用する場合はチェックしてください。
- [周波数選択]: 高速、低速から使用するものを選択してください。それぞれ高速オンチップオシレータ、低速オンチップオシレータに対応します。
- [周期値]: 高速オンチップオシレータを選択した場合に設定します。高速オンチップオシレータの周期を決定する高速オンチップオシレータ制御レジスタ 1(HR1)の設定値を入力します。
- [分周比選択]: 低速オンチップオシレータを選択した場合に有効になります。1分周のみ選択可能です。
- [オンチップオシレータ周波数]: オンチップオシレータ発振周波数と周期値、分周比から算出したオンチップオシレータクロックの周波数が表示されます。

### (3) システムクロック設定

(1)[システムクロックの選択]で指定したクロックをCPUに供給するための分周率の設定を行います。

- [システムクロック周波数]: (1)[システムクロック選択]で指定したクロックの周波数が表示されます。
- [システムクロック分周比]: 設定しません。
- [CPU]、[周辺]タブ: CPUクロックおよび周辺機能クロックの設定を行います。表4.1.6 に各項目の設定内容を示します。

表4.1.6 システムクロック設定内容 (R8C/13)

項目	設定内容	
CPU	分周比選択	CPU に供給するクロックの分周比を選択してください。
	供給クロック	[システムクロック周波数]と[分周比選択]の設定値から算出した、CPU 供給クロックの周波数が表示されます。
	内部周期	[システムクロック周波数]と[分周比選択]の設定値から算出した、CPU 供給クロックの周期が表示されます。
周辺	分周比選択	設定しません [システムクロックの選択]で指定したクロックを各分周率で分周したクロックが周辺 IO モジュールに供給されます。各周辺 IO モジュールで使用するクロックは、周辺 IO モジュールの設定で指定してください。 fRING の周波数は[オンチップオシレータクロック]の設定に従います。
	供給クロック	[システムクロック選択]で指定したクロックの周波数が表示されます。
	内部周期	[システムクロック選択]で指定したクロックの周期が表示されます。

## 4.1.3 R8C/22-23 のクロック設定

R8C/22-23 の[CPUクロック設定]ダイアログボックスを 図4.1.3 に示します。

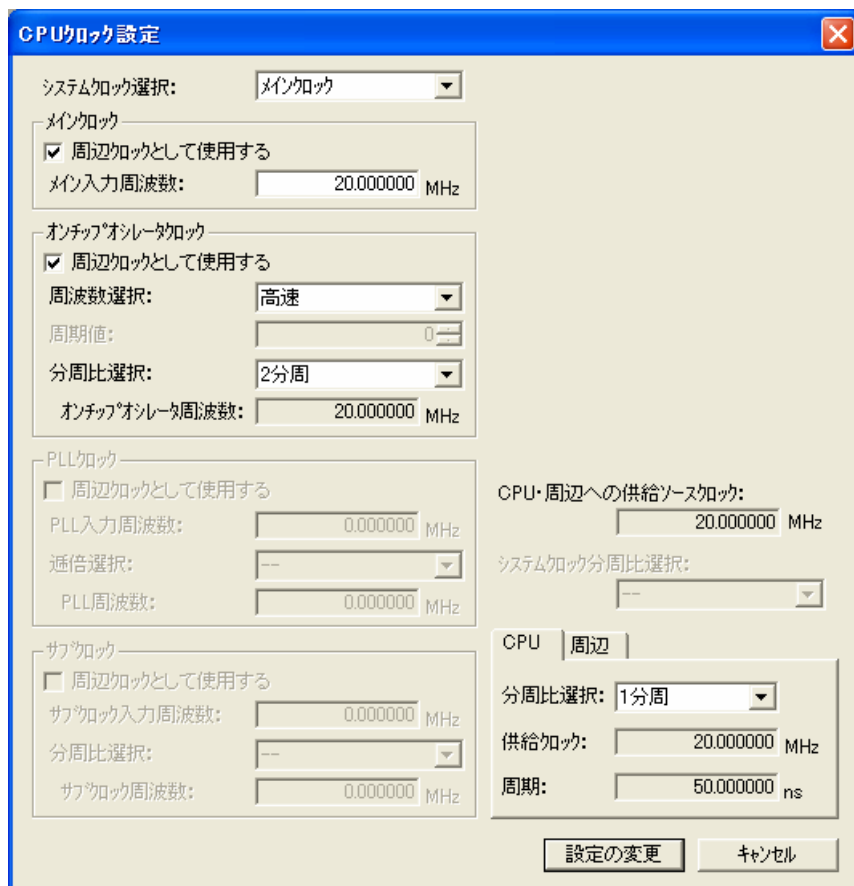


図4.1.3 CPUクロック設定ダイアログボックス (R8C/22-23)

## (1) システムクロック選択

CPUクロックとして使用するクロックを選択します。メインクロックまたはオンチップオシレータクロックから選択可能です。[システムクロック選択]の各項目は、表4.1.7に示すクロック源に対応します。

表4.1.7 R8C/22-23のクロック源

項目	クロック源	
メインクロック	XINクロック発信回路	
オンチップオシレータクロック	オンチップオシレータ	高速オンチップオシレータ
		低速オンチップオシレータ

## (2) 各クロックの設定

[メインクロック]、[オンチップオシレータクロック]の各項目より[システムクロック選択]で選択したクロックの設定を行います。[システムクロック選択]で指定したCPU供給クロックは周辺クロックとしても使用されます。ただしタイマカウントソースまたはA/Dコンバータの動作クロックとして選択可能なfOCO40M、fOCOおよびfOCO-Fはオンチップオシレータのみが供給可能です。CPUクロックとしてメインクロックを選択し、fOCO40M、fOCOまたはfOCO-Fを使用する場合はオンチップオシレータクロックの設定も行ってください。オンチップオシレータクロックを設定していない場合、タイマ設定またはA/D変換の設定時、動作クロックとしてfOCO40M、fOCOおよびfOCO-Fを選択できません。

CPUクロックおよび周辺機能クロックとして使用できるクロックの組み合わせを表4.1.8に示します。

表4.1.8 CPUおよび周辺クロック源の組み合わせ (R8C/22-23)

CPUクロック	周辺機能クロック	
	f1～f32	fOCO40M、fOCO、fOCO-F
メインクロック	メインクロック	オンチップオシレータクロック
オンチップオシレータクロック	オンチップオシレータクロック	オンチップオシレータクロック

#### [メインクロック]

CPUクロックとしてメインクロックを指定した場合に設定します。

- [周辺クロックとして使用する]：CPU クロックとしてメインクロックを使用する場合は常にチェックされます。
- [Main 入力周波数]：メインクロックの周波数を設定してください。設定可能な値は 0MHz～20MHz です。

#### [オンチップオシレータクロック]

CPUクロックとしてオンチップオシレータクロックを指定した場合と、CPUクロックとしてメインクロックを指定し、周辺機能でfOCO40M、fOCOまたはfOCO-Fを使用する場合に設定します。

- [周辺クロックとして使用する]：CPU クロックとしてオンチップオシレータクロックを使用する場合は常にチェックされます。CPU クロックとしてメインクロックを指定し、fOCO40M、fOCO または fOCO-F を使用する場合はチェックしてください。
- [周波数選択]：高速、低速から使用するものを選択してください。それぞれ高速オンチップオシレータ、低速オンチップオシレータに対応します。
- [周期値]：設定しません。
- [分周比選択]：オンチップオシレータクロックの分周比を選択してください。低速オンチップオシレータの場合、分周率は1分周のみ選択可能です。
- [オンチップオシレータ周波数]：オンチップオシレータ発振周波数と分周比から算出したオンチップオシレータクロックの周波数が表示されます。

### (3) システムクロック設定

(1)[システムクロックの選択]で指定したクロックをCPUに供給するための分周率の設定を行います。

- [システムクロック周波数]：(1)[システムクロック選択]で指定したクロックの周波数が表示されます。
- [システムクロック分周比]：設定しません。
- [CPU]、[周辺]タブ：CPUクロックおよび周辺機能クロックの設定を行います。表4.1.9に各項目の設定内容を示します。

表4.1.9 システムクロック設定内容 (R8C/22-23)

項目	設定内容	
CPU	分周比選択	CPU に供給するクロックの分周比を選択してください。
	供給クロック	[システムクロック周波数]と[分周比選択]の設定値から算出した、CPU 供給クロックの周波数が表示されます。
	内部周期	[システムクロック周波数]と[分周比選択]の設定値から算出した、CPU 供給クロックの周期が表示されます。
周辺	分周比選択	設定しません [システムクロックの選択]で指定したクロックを各分周率で分周したクロックが周辺 IO モジュールに供給されます。各周辺 IO モジュールで使用するクロックは、周辺 IO モジュールの設定で指定してください。 fOCO40M、fOCO および fOCO-F の周波数は[オンチップオシレータクロック]の設定に従います。
	供給クロック	[システムクロック選択]で指定したクロックの周波数が表示されます。
	内部周期	[システムクロック選択]で指定したクロックの周期が表示されます。



## 4.1.4 R8C/24-25 のクロック設定

R8C/24-25 の[CPUクロック設定]ダイアログボックスを 図4.1.4 に示します。

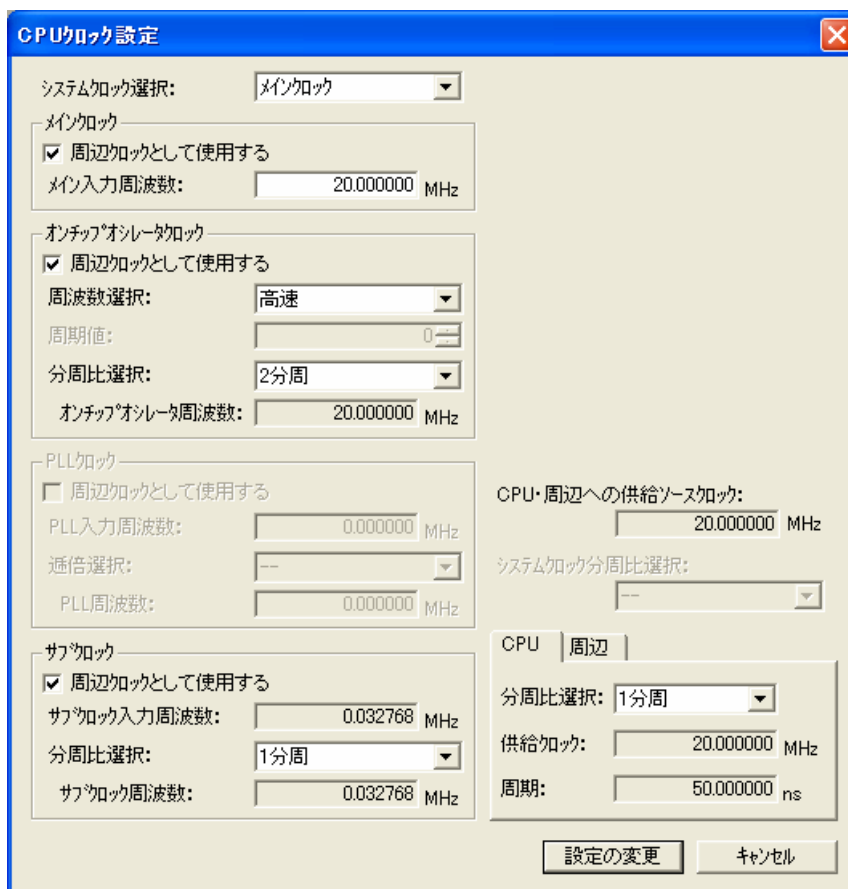


図4.1.4 CPUクロック設定ダイアログボックス (R8C/24-25)

## (1) システムクロック選択

CPUクロックとして使用するクロックを選択します。メインクロック、オンチップオシレータクロックおよびサブクロックから選択可能です。[システムクロック選択]の各項目は、表4.1.10に示すクロック源に対応します。

表4.1.10 R8C/24-25のクロック源

項目	クロック源	
メインクロック	XINクロック発信回路	
サブクロック	XCINクロック発振回路	
オンチップオシレータクロック	オンチップオシレータ	高速オンチップオシレータ
		低速オンチップオシレータ

## (2) 各クロックの設定

[メインクロック]、[オンチップオシレータクロック]および[サブクロック]の各項目より、CPUクロックとして選択したクロックの設定を行います。[システムクロック選択]で指定したCPU供給クロックは周辺クロックとしても使用されます。各クロック源が供給可能なクロックを表4.1.11に示します。

表4.1.11 各クロック源の供給可能クロック(R8C/24-25)

クロック源		供給可能クロック
メインクロック		CPUクロック、周辺クロック(f1～f32)
サブクロック		CPUクロック、周辺クロック(fC4、fC32)
オンチップ オシレータクロック	高速	CPUクロック、周辺クロック(f1～f32、fOCO、fOCO-F、fOCO40M)
	低速	CPUクロック、周辺クロック(f1～f32、fOCO、fOCO-S)

[システムクロック選択]で指定したクロックが供給可能なクロック以外に必要なクロックがある場合は、必要なクロックの[周辺クロックとして使用する]チェックボックスをチェックして設定してください。

CPUクロックおよび周辺機能クロックとして使用できるクロックの組み合わせを 表4.1.12に示します。

表4.1.12 CPUおよび周辺クロック源の組み合わせ(R8C/24-25)

CPUクロック	周辺機能クロック		
	f1～f32	fC4、fC32	fOCO40M、fOCO、fOCO-F
メインクロック	メインクロック	サブクロック	オンチップオシレータクロック
オンチップオシレータクロック	オンチップオシレータクロック	サブクロック	オンチップオシレータクロック
サブクロック	メインクロック	サブクロック	オンチップオシレータクロック
	オンチップオシレータクロック	サブクロック	オンチップオシレータクロック

#### [メインクロック]

CPUクロックとしてメインクロックを指定した場合と、CPUクロックとしてサブクロックを指定し、周辺機能クロックf1～f32のクロック源としてメインクロックを使用する場合に設定します。

- [周辺クロックとして使用する]: CPU クロックとしてメインクロックを使用する場合は常にチェックされます。CPU クロックとしてサブクロックを指定し、周辺機能クロック f1～f32 のクロック源としてメインクロックを使用する場合はチェックしてください。
- [Main 入力周波数]: メインクロックの周波数を設定してください。

#### [オンチップオシレータクロック]

CPUクロックとしてオンチップオシレータクロックを指定した場合と周辺機能クロックとしてオンチップオシレータクロックを使用する場合に設定します。

- [周辺クロックとして使用する]: CPU クロックとしてオンチップオシレータクロックを使用する場合は常にチェックされます。CPU クロックとしてオンチップオシレータクロック以外を選択し、周辺機能クロックとしてオンチップオシレータクロックを使用する場合はチェックしてください。
- [周波数選択]: 高速、低速から使用するものを選択してください。それぞれ高速オンチップオシレータ、低速オンチップオシレータに対応します。
- [周期値]: 設定しません。
- [分周比選択]: オンチップオシレータクロックの分周比を選択してください。低速オンチップオシレータの場合、分周率は1分周のみ選択可能です。
- [オンチップオシレータ周波数]: オンチップオシレータ発振周波数と分周比から算出したオンチップオシレータクロックの周波数が表示されます。



## [サブクロック]

CPUクロックとしてサブクロックを指定した場合と、周辺機能クロックの供給源としてサブクロックを使用する場合に設定します。

- [周辺クロックとして使用する]: システムクロックとして使用する場合は常にチェックされます。システムクロックとしてサブクロック以外を選択した場合は、タイマ RA またはタイマ RE のカウントソースとして fC4 または fC32 を使用するためにチェックします。
- [Sub 入力周波数]: サブクロックの周波数は 0.03276MHz 固定です。
- [分周比選択]: 1 分周のみ指定可能です。
- [サブクロック周波数]: Sub 入力周波数と分周比から算出したサブクロックの周波数が表示されます。0.03276MHz 固定です。

## (3) システムクロック設定

(1)[システムクロックの選択]で指定したクロックをCPUに供給するための分周率の設定を行います。

- [システムクロック周波数]: (1)[システムクロック選択]で指定したクロックの周波数が表示されます。
- [システムクロック分周比]: 設定しません。
- [CPU]、[周辺]タブ: CPU クロックおよび周辺機能クロックの設定を行います。表 4.1.13 に各項目の設定内容を示します。

表4.1.13 システムクロック設定内容 (R8C/24-25)

項目	設定内容	
CPU	分周比選択	CPU に供給するクロックの分周比を選択してください。 サブクロックを CPU クロックとして使用する場合は分周期を介さないため、分周比は選択できません。
	供給クロック	[システムクロック周波数]と[分周比選択]の設定値から算出した、CPU 供給クロックの周波数が表示されます。 サブクロックを CPU クロックとして使用する場合はサブクロックの周波数が表示されます。
	内部周期	[システムクロック周波数]と[分周比選択]の設定値から算出した、CPU 供給クロックの周期が表示されます。
周辺	分周比選択	設定しません [システムクロックの選択]で指定したクロックを各分周率で分周したクロックが周辺 IO モジュールに供給されます。各周辺 IO モジュールで使用するクロックは、周辺 IO モジュールの設定で指定してください。
	供給クロック	[システムクロック選択]で指定したクロックの周波数が表示されます。
	内部周期	[システムクロック選択]で指定したクロックの周期が表示されます。

## 4.1.5 R8C/26-29、R8C/2A-2Dのクロック設定

R8C/26-29、R8C/2A-2Dの[CPUクロック設定]ダイアログボックスを図4.1.5に示します。

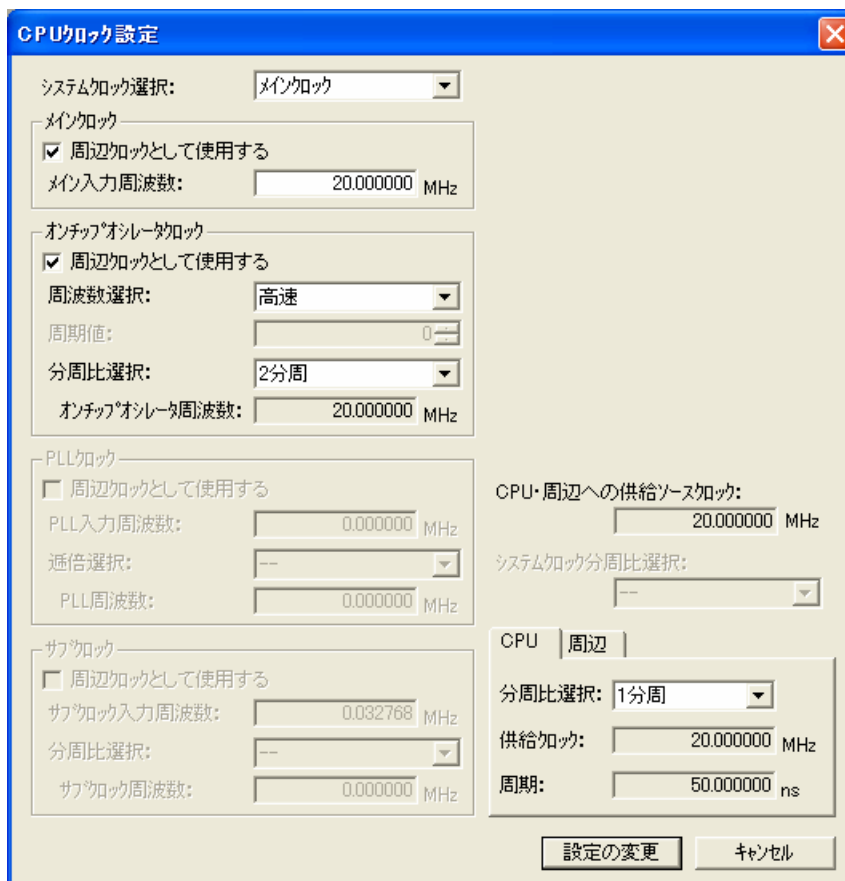


図4.1.5 CPUクロック設定ダイアログボックス (R8C/26-29、R8C/2A-2D)

## (1) システムクロック選択

CPUクロックとして使用するクロックを選択します。メインクロック、オンチップオシレータクロックおよびサブクロックから選択可能です。[システムクロック選択]の各項目は、表4.1.14に示すクロック源に対応します。

表4.1.14 R8C/26-29、R8C/2A-2Dのクロック源

項目	クロック源	
メインクロック	XINクロック発信回路	
サブクロック	XCINクロック発振回路	
オンチップオシレータクロック	オンチップオシレータ	高速オンチップオシレータ
		低速オンチップオシレータ

## (2) 各クロックの設定

[メインクロック]、[オンチップオシレータクロック]および[サブクロック]の各項目より、CPUクロックとして選択したクロックの設定を行います。[システムクロック選択]で指定したCPU供給クロックは周辺クロックとしても使用されます。各クロック源が供給可能なクロックを表4.1.15に示します。

表4.1.15 各クロック源の供給可能クロック (R8C/26-29、R8C/2A-2D)

クロック		供給可能クロック
メインクロック		CPUクロック、周辺クロック(f1~f32)
サブクロック		CPUクロック、周辺クロック(f1~f32、fC4、fC32)
オンチップ	高速	CPUクロック、周辺クロック(f1~f32、fOCO、fOCO-F、fOCO40M)
オシレータクロック	低速	CPUクロック、周辺クロック(f1~f32、fOCO、fOCO-S)

[システムクロック選択]で指定したクロックが供給可能なクロック以外に必要なクロックがある場合は、必要なクロックの[周辺クロックとして使用する]チェックボックスをチェックし、設定してください。ただしメインクロックとサブクロックを同時に使用することはできません。

CPUクロックおよび周辺機能クロックとして使用できるクロックの組み合わせを表4.1.16に示します。

表4.1.16 CPUおよび周辺クロック源の組み合わせ (R8C/26-29、R8C/2A-2D)

CPUクロック	周辺機能クロック		
	f1~f32	fC4、fC32	fOCO40M、fOCO、fOCO-F
メインクロック	メインクロック	—	オンチップ オシレータクロック
オンチップ オシレータクロック	オンチップ オシレータクロック	サブクロック	オンチップ オシレータクロック
サブクロック	オンチップ オシレータクロック	サブクロック	オンチップ オシレータクロック

#### [メインクロック]

CPUクロックとしてメインクロックを指定した場合に設定します。

- [周辺クロックとして使用する]: CPU クロックとしてメインクロックを使用する場合は常にチェックされます。
- [Main 入力周波数]: メインクロックの周波数を設定してください。

#### [オンチップオシレータクロック]

CPUクロックとしてオンチップオシレータクロックを指定した場合と周辺機能クロックの供給源としてオンチップオシレータクロックを使用する場合に設定します。

- [周辺クロックとして使用する]: CPU クロックとしてオンチップオシレータクロックを使用する場合は常にチェックされます。CPU クロックとしてオンチップオシレータクロック以外を選択し、周辺機能クロックとしてオンチップオシレータクロックを使用する場合はチェックしてください。
- [周波数選択]: 高速、低速から使用するものを選択してください。それぞれ高速オンチップオシレータ、低速オンチップオシレータに対応します。
- [周期値]: 設定しません。
- [分周比選択]: オンチップオシレータクロックの分周比を選択してください。低速オンチップオシレータの場合、分周率は1分周のみ選択可能です。
- [オンチップオシレータ周波数]: オンチップオシレータ発振周波数と分周比から算出したオンチップオシレータクロックの周波数が表示されます。

#### [サブクロック]

CPUクロックとしてサブクロックを指定した場合と、周辺機能クロックの供給源としてサブクロックを使用する場合に設定します。

- [周辺クロックとして使用する]: システムクロックとしてサブクロックを使用する場合は常にチェックされます。システムクロックとしてオンチップオシレータクロックを選択した場合は、タイマ RA またはタイマ RE のカウントソースとして fC4 または fC32 を使用するためにチェックします。
- [Sub 入力周波数]: サブクロックの周波数は 0.03276MHz 固定です。
- [分周比選択]: 1 分周のみ指定可能です。
- [サブクロック周波数]: Sub 入力周波数と分周比から算出したサブクロックの周波数が表示されます。0.03276MHz 固定です。

### (3) システムクロック設定

(1)[システムクロックの選択]で指定したクロックをCPUに供給するための分周率の設定を行います。

- [システムクロック周波数]: (1)[システムクロック選択]で指定したクロックの周波数が表示されます。
- [システムクロック分周比]: 設定しません。
- [CPU]、[周辺]タブ: CPU クロックおよび周辺機能クロックの設定を行います。表 4.1.17 に各項目の設定内容を示します。

表4.1.17 システムクロック設定内容 (R8C/26-29、R8C/2A-2D)

項目		設定内容
CPU	分周比選択	CPU に供給するクロックの分周比を選択してください。
	供給クロック	[システムクロック周波数]と[分周比選択]の設定値から算出した、CPU 供給クロックの周波数が表示されます。
	内部周期	[システムクロック周波数]と[分周比選択]の設定値から算出した、CPU 供給クロックの周期が表示されます。
周辺	分周比選択	設定しません [システムクロックの選択]で指定したクロックを各分周率で分周したクロックが周辺 IO モジュールに供給されます。各周辺 IO モジュールで使用するクロックは、周辺 IO モジュールの設定で指定してください。
	供給クロック	[システムクロック選択]で指定したクロックの周波数が表示されます。
	内部周期	[システムクロック選択]で指定したクロックの周期が表示されます。

## 4.1.6 H8/3687、H8/36049 のクロック設定

H8/3687、H8/36049 の[CPUクロック設定]ダイアログボックスを 図4.1.6 に示します。

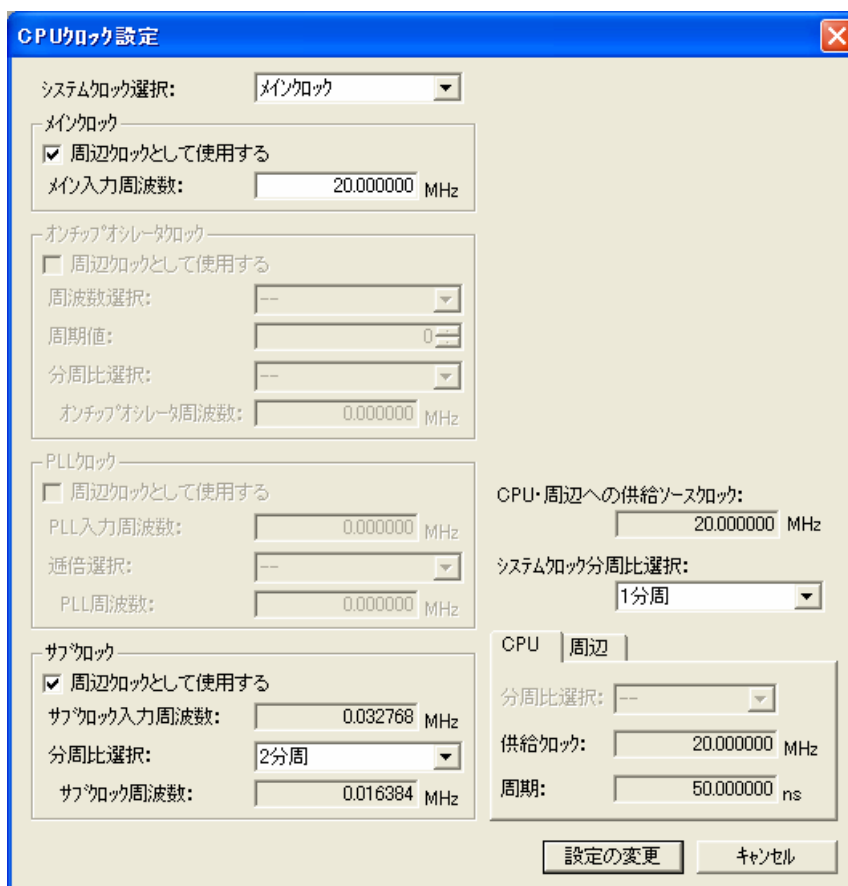


図4.1.6 CPUクロック設定ダイアログボックス (H8/3687,H8/36049)

## (1) システムクロック選択

CPUクロックとして使用するクロックを選択します。メインクロックのみ選択可能です。

## (2) 各クロックの設定

[メインクロック]および[サブクロック]の各項目より、各クロックの設定を行います。[メインクロック]および[サブクロック]は表4.1.18に示す各クロック源に対応します。

表4.1.18 H8/3687、H8/36049のクロック源

項目	クロック源
メインクロック	システムクロック発信器
サブクロック	サブクロック発信器

## [メインクロック]

メインクロックの周波数 ( OSC ) を設定します。

- [周辺クロックとして使用する]: メインクロックは CPU および周辺機能を動作させる基準クロックとなります。常にチェックされます。
- [Main 入力周波数]: メインクロックの周波数 ( OSC ) を設定してください。システムクロック ( ) の周波数を決定するシステムクロック分周比の選択は後述のシステムクロック設定で行います。

#### [サブクロック]

サブクロック周波数( SUB)を設定します。

- [周辺クロックとして使用する]: サブクロックは周辺機能を動作させる基準クロックとなります。常にチェックされます。
- [サブクロック入力周波数]: サブクロックの入力周波数は 0.03276MHz 固定です。
- [分周比選択]: サブクロック分周器の分周率を選択してください。
- [サブクロック周波数]: Sub 入力周波数と分周比から算出したサブクロック( SUB)の周波数が表示されます。サブクロック( SUB)をプリスケアラ W により分周したクロックが周辺機能に供給されます。

### (3) システムクロック設定

CPU および周辺機能に供給する基準クロック( ) の設定を行います。

- [システムクロック周波数]: メインクロックの周波数が表示されます。
- [システムクロック分周比]: システムクロック分周器の分周率を選択してください。
- [CPU]、[周辺]タブ: CPU クロックおよび周辺機能クロックの設定を行います。表 4.1.19 に各項目の設定内容を示します。

表 4.1.19 システムクロック設定内容 (H8/3687、H8/36049)

項目		設定内容
CPU	分周比選択	設定しません
	供給クロック	[システムクロック周波数]と[システムクロック分周比]の設定値から算出した、システムクロック( $\phi$ )の周波数が表示されます。
	内部周期	[システムクロック周波数]と[分周比選択]の設定値から算出した、システムクロック( $\phi$ )の周期が表示されます。
周辺	分周比選択	設定しません 各周辺機能にはシステムクロック( $\phi$ )をプリスケアラ S により分周したクロックが供給されます。
	供給クロック	システムクロック( $\phi$ )の周波数が表示されます。
	内部周期	システムクロック( $\phi$ )の周期が表示されます。

## 4.1.7 H8/36077、H8/36109 のクロック設定

H8/36077、H8/36109 の[CPUクロック設定]ダイアログボックスを 図4.1.7 に示します。

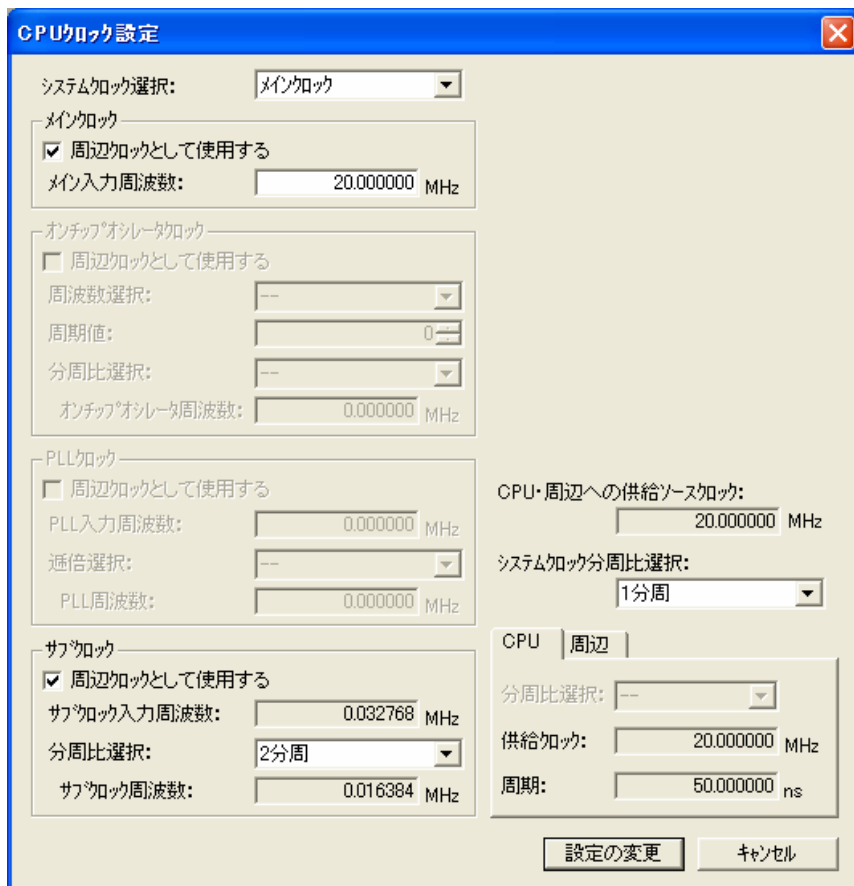


図4.1.7 CPUクロック設定ダイアログボックス (H8/36077,H8/36109)

## (1) システムクロック選択

システムクロック として使用するクロックを選択します。[メインクロック]および[オンチップオシレータクロック]から選択してください。それぞれ表4.1.20に示すクロック源に対応します。

表4.1.20 H8/36077、H8/36109のクロック源

項目	クロック源
メインクロック	外部クロック発信器
オンチップオシレータクロック	オンチップオシレータ

## (2) 各クロックの設定

[システムクロック選択]で選択したクロックとサブクロックの設定を行います。

## [メインクロック]

[システムクロック選択]でメインクロックを選択した場合に OSCの周波数を設定します。

- [周辺クロックとして使用する]：システムクロック はCPU および周辺機能を動作させる基準クロック



クとなります。[システムクロック選択]でシステムクロック の供給源としてメインクロックを指定した場合は常にチェックされます。

- [Main 入力周波数]: メインクロックの周波数 ( OSC ) を設定してください。システムクロック ( ) の周波数を決定するシステムクロック分周比の選択は後述のシステムクロック設定で行います。

#### [オンチップオシレータクロック]

[システムクロック選択]でオンチップオシレータクロックを選択した場合に RCの周波数を設定します。

- [周辺クロックとして使用する]: システムクロック は CPU および周辺機能を動作させる基準クロックとなります。[システムクロック選択]でシステムクロック の供給源としてオンチップオシレータクロックを指定した場合は常にチェックされます。
- [周波数選択]: オンチップオシレータクロック ( ROOSC ) の発振周波数を選択してください。
- [周期値]: 設定しません。
- [分周比選択]: RC クロック分周器の分周比を選択してください。
- [オンチップオシレータ周波数]: オンチップオシレータ発振周波数と RC クロック分周比から算出した周波数が表示されます。システムクロック ( ) の周波数を決定するシステムクロック分周比の選択は後述のシステムクロック設定で行います。

#### [サブクロック]

サブクロック周波数( SUB)を設定します。

- [周辺クロックとして使用する]: サブクロックは周辺機能を動作させる基準クロックとなります。常にチェックされます。
- [Sub 入力周波数]: サブクロックの周波数は 0.03276MHz固定です。
- [分周比選択]: サブクロック分周器の分周率を選択してください。
- [サブクロック周波数]: Sub 入力周波数と分周比から算出したサブクロック( SUB)の周波数が表示されます。サブクロック( SUB)をプリスケラ W により分周したクロックが周辺機能に供給されます。

### (3) システムクロック設定

CPUおよび周辺機能に供給する基準クロック( )の設定を行います。

- [システムクロック周波数]: [メインクロック]または[オンチップオシレータクロック]で設定したシステムクロック( )供給源のクロックの周波数が表示されます。
- [システムクロック分周比]: システムクロック分周器の分周率を選択してください。
- [CPU]、[周辺]タブ: CPU クロックおよび周辺機能クロックの設定を行います。表 4.1.21 に各項目の設定内容を示します。

表4.1.21 システムクロック設定内容 (H8/36077、H8/36109)

項目	設定内容	
CPU	分周比選択	設定しません
	供給クロック	[システムクロック周波数]と[システムクロック分周比]の設定値から算出した、システムクロック( $\phi$ )の周波数が表示されます。
	内部周期	[システムクロック周波数]と[分周比選択]の設定値から算出した、システムクロック( $\phi$ )の周期が表示されます。
周辺	分周比選択	設定しません 各周辺機能にはシステムクロック( $\phi$ )をプリスケラ S により分周したクロックが供給されます。
	供給クロック	システムクロック( $\phi$ )の周波数が表示されます。
	内部周期	システムクロック( $\phi$ )の周期が表示されます。



## 4.1.8 SH7125 のクロック設定

SH7125 の[CPUクロック設定]ダイアログボックスを 図4.1.8 に示します。

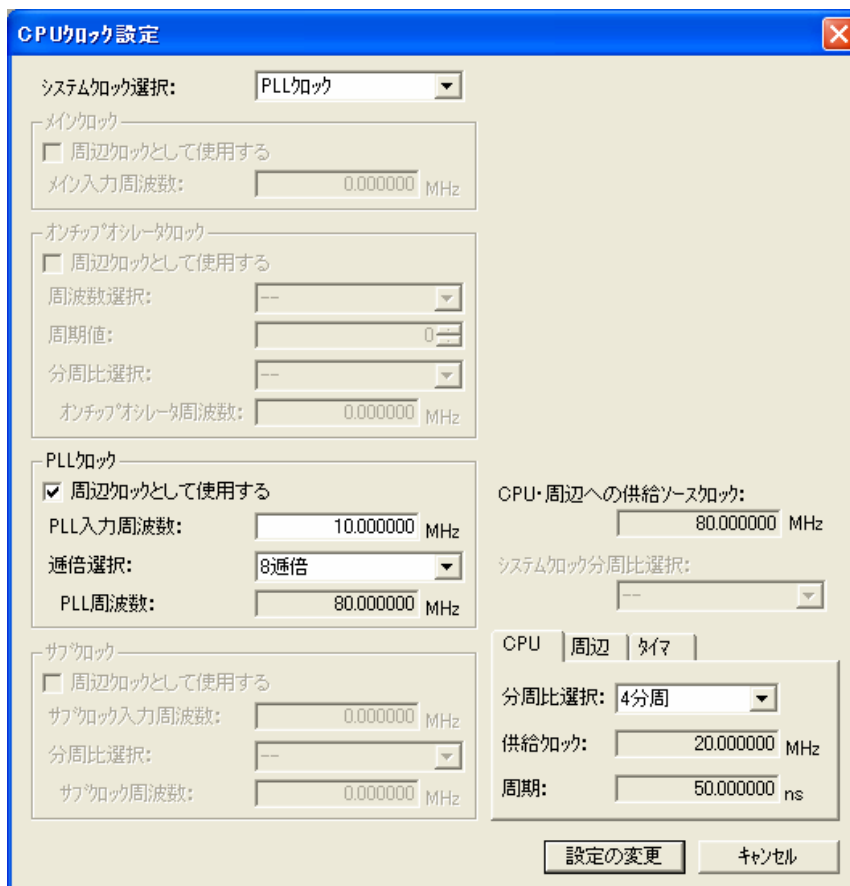


図4.1.8 CPUクロック設定ダイアログボックス (SH7125)

## (1) システムクロック選択

CPUクロック、周辺機能クロックの供給源を設定します。SH7125のクロック供給源はPLL回路のみです。常にPLLクロックが指定されます。

## (2) PLL クロックの設定

[PLLクロック]でPLL回路の入力周波数を設定してください。

## [PLL クロック]

PLL回路の出力周波数を設定します。内部クロック (I )、周辺クロック (P )、およびMTU2 クロック (MP )の周波数の設定は後述のシステムクロック設定で行います。

- [周辺クロックとして使用する]: PLL クロックは CPU および周辺機能に供給されます。常にチェックされます。
- [PLL 入力周波数]: PLL 回路への入力周波数を設定してください。
- [逓倍選択]: PLL 回路は入力クロックを 8 逓倍して出力します。8 逓倍のみ選択可能です。
- [PLL 周波数]: PLL 回路の入力周波数と逓倍率から算出した PLL 回路の出力の周波数が表示されます。

## (3) システムクロック設定

各クロックの分周率を設定し、内部クロック (I $\phi$ )、周辺クロック (P $\phi$ )、および MTU2 クロック (MP $\phi$ ) の周波数を設定します。

- [システムクロック周波数]: PLL 回路の出力周波数が表示されます。
- [システムクロック分周比]: 設定しません。
- [CPU]、[周辺]および[タイマ]タブ: 内部クロック (I $\phi$ )、周辺クロック (P $\phi$ )、および MTU2 クロック (MP $\phi$ ) の設定を行います。表 4.1.22 に各項目の設定内容を示します。

表4.1.22 システムクロック設定内容 (SH7125)

項目	設定内容	
CPU	分周比選択	内部クロック(I $\phi$ )周波数の分周率を選択してください
	供給クロック	PLL 回路の出力周波数と[分周比選択]で選択した分周率から算出した、内部クロック(I $\phi$ )の周波数が表示されます。
	内部周期	内部クロック(I $\phi$ )の周期が表示されます。
周辺	分周比選択	周辺クロック(P $\phi$ )周波数の分周率を選択してください
	供給クロック	PLL 回路の出力周波数と[分周比選択]で選択した分周率から算出した、周辺クロック(P $\phi$ )の周波数が表示されます。
	内部周期	周辺クロック(P $\phi$ )の周期が表示されます。
タイマ	分周比選択	MTU2 クロック(MP $\phi$ )周波数の分周率を選択してください
	供給クロック	PLL 回路の出力周波数と[分周比選択]で選択した分周率から算出した、MTU2 クロック(MP $\phi$ )の周波数が表示されます。
	内部周期	MTU2 クロック(MP $\phi$ )の周期が表示されます。

## 4.2 シリアルインタフェースの設定

同期式、非同期式の2方式の設定が可能です。これらの方式に対応する各マイコンの通信モードを表 4.2.1 に示します。

表4.2.1 同期式/非同期式と各CPUにおける通信モードの対応

マイコン	同期式	非同期式
M16C/62p	クロック同期形	クロック非同期形
M16C/28,28B,29		
R8C/13,22-29,2A-2D		
H8/3687,6077, 36049, 36109	クロック同期式	調歩同期式
SH7125		

各マイコンのシリアル通信リソースで設定可能な通信モードを表 4.2.2 に示します。

表4.2.2 各CPUのシリアル通信リソース

マイコン	シリアル通信リソース	転送方式
M16C/62p	UART <sub>i</sub> (i=0~2)	同期式、非同期式
	SI/O3、SI/O4	同期式
M16C/28,28B,29	UART <sub>i</sub> (i=0~2)	同期式、非同期式
	SI/O3、SI/O4	同期式
R8C/13,22,23,28,29	UART0	同期式、非同期式
	UART1	非同期式
R8C/24-27	UART0	同期式、非同期式
	UART1	同期式、非同期式
R8C/2A-2D	UART <sub>i</sub> (i=0~2)	同期式、非同期式
H8/3687,36077	SCI3チャンネル1	同期式、非同期式
	SCI3チャンネル2	同期式、非同期式
H8/36049,36109	SCI3チャンネル1	同期式、非同期式
	SCI3チャンネル2	同期式、非同期式
	SCI3チャンネル3	同期式、非同期式
SH7125	チャンネル0-2	同期式、非同期式

各方式の設定方法を以下に示します。

## 4.2.1 同期式シリアルインタフェース

同期式シリアルインタフェースの設定ダイアログを図 4.2.1 に示します。

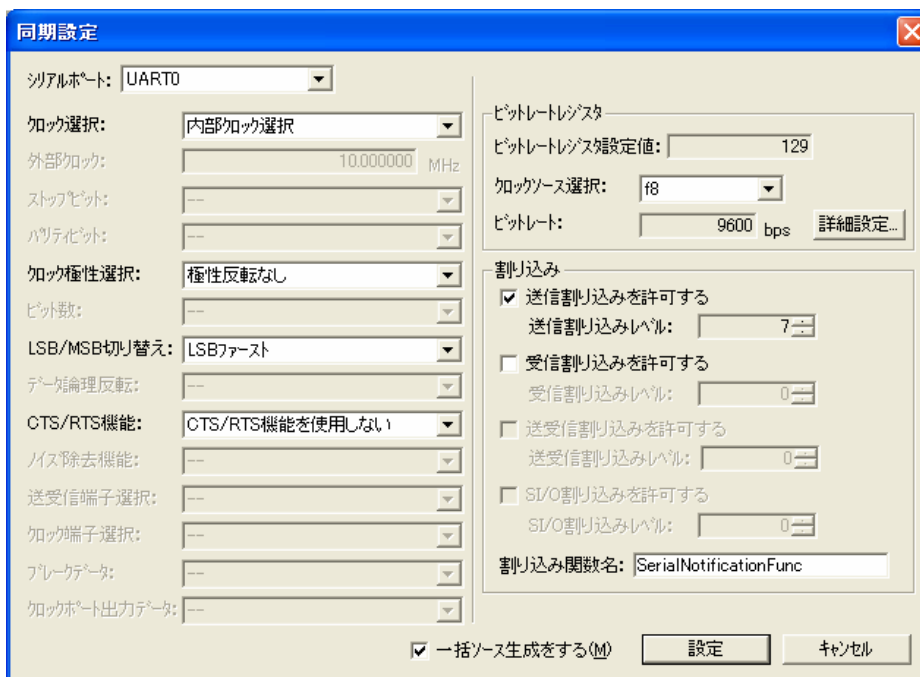


図4.2.1 同期式シリアルインタフェース設定ダイアログ

各項目の設定方法を以下に示します。

## [シリアルポート]

設定するシリアル通信インタフェースリソースを指定してください。[設定なし]を選択すると、リソースを指定しない設定を作成し、後からリソースを割り当てることができます。ただし SH7125 では[設定なし]を選択できません。

## [クロック選択]

転送クロックを内部クロックと外部クロックから選択してください。

## [ストップビット]

設定しません。

## [パリティビット]

設定しません。

## [クロック極性選択]

転送クロックの極性を選択します。設定の有無はマイコンにより異なります。[転送クロック極性選択]の設定内容は表 4.2.3 に示す通りです。

表4.2.3 [転送クロック]極性設定内容

マイコン	設定	内容
M16C/62p M16C/28,28B,29 R8C/13,22-29,2A-2D	[極性反転なし]	転送クロックの立ち下がりで送信データ出力、立ち上がりで受信データ入力
	[極性反転あり]	転送クロックの立ち上がりで送信データ出力、立ち下がりで受信データ入力

## [ビット数]

設定しません。

## [LSB/MSB 切り替え]

転送フォーマットを選択します。設定の有無はマイコンにより異なります。[LSB/MSB 切り替え] の設定内容を表 4.2.4 に示します。

表4.2.4 [LSB/MSB切り替え]設定内容

マイコン	設定	内容
M16C/62p	LSB ファースト	転送フォーマットを LSB ファーストにします
M16C/28,28B,29 R8C/13,22-29,2A-2D	MSB ファースト	転送フォーマットを MSB ファーストにします

## [データ論理反転]

データ論理反転を設定します。設定の有無はマイコン、シリアル通信リソースにより異なります。[データ論理反転] の設定内容を表 4.2.5 に示します。

表4.2.5 [データ論理反転] 設定内容

マイコン	設定	内容
M16C/62p ※1	データ論理選択反転あり	送受信データを論理反転します
M16C/28,28B,29 ※2	データ論理選択反転なし	送受信データを論理反転しません

※1 UART0~2 選択時のみ有効です。

※2 UART2 選択時のみ有効です。

## [CTS/RTS 機能]

CTS/RTS 機能を設定します。設定の有無はマイコン、シリアル通信リソースにより異なります。[CTS/RTS 機能] の設定内容を表 4.2.6 に示します。

表4.2.6 [CTS/RTS機能] 設定内容

マイコン	設定	内容
M16C/62p ※1	CTS/RTS 機能を使用しない	CTS/RTS 機能を使用しません
M16C/28,28B,29 ※1	CTS 選択	CTS 機能を使用します
	RTS 選択 ※2	RTS 機能を使用します

※1 UART0~2 選択時のみ有効です。

※2 クロック同期シリアル通信モードの内部クロック選択時は選択できません。

## [ノイズ除去機能]

設定しません。

## [クロック端子選択]

転送クロックの入出力に使用する端子を選択します。設定の有無はマイコン、選択したシリアル通信リソースにより異なります。[クロック端子選択] の設定内容を表 4.2.7 に示します。

表4.2.7 [クロック端子選択]設定内容

マイコン	設定	内容
R8C/2A,2B,2C,2D ※	P0_5	転送クロックの入出力に P0_5 端子を使用します
	P6_5	転送クロックの入出力に P6_5 端子を使用します

UART1 選択時のみ有効です。

## [送受信端子選択]

送受信に使用する端子を選択します。設定の有無はマイコン、選択したシリアル通信リソースにより異なります。[送受信端子選択] の設定内容を表 4.2.8 に示します。

表4.2.8 [送受信端子選択]設定内容

マイコン	設定	内容
M16C/26,27 ※	Tx1_P0_0, RX1_P3_7	送受信に P3_7(TXD1/RXD1)端子を使用します
	Tx1_P3_7, RX1_P4_5	送受信に P3_7(TXD1)、P4_5(RXD1)端子を使用します
	Tx1_P0_0, Rx1_P3_6	送受信に P3_6(TXD1/RXD1)端子を使用します

UART1 選択時のみ有効です。

## [ブ레이크データ]

設定しません。

## [クロックポート出力データ]

設定しません。

## [ビットレートレジスタ]

以下の各項目で転送速度の設定を行います。

## [ビットレートレジスタ設定値]

ビットレートレジスタの値が表示されます。値は後述の[カウントソース]、[詳細設定]で指定したボーレートから自動的に算出されます。

## [クロックソース選択]

内部クロック選択時のビットレートレジスタのカウントソースを選択してください。外部クロック選択時は[f1]が表示されますが、カウントソースは外部クロックです。

## [ボーレート]

ビットレートレジスタ値と選択したカウントソースの周波数または外部クロック周波数から算出したボーレート値が表示されます。

## [詳細設定]

ビットレートレジスタ値とエラー率の算出、ボーレート値の変更またはカウントソースの選定を行う場合に使用します。クリックすると、図4.2.2 に示す[ビットレートレジスタ設定値計算]ダイアログが表示されます。

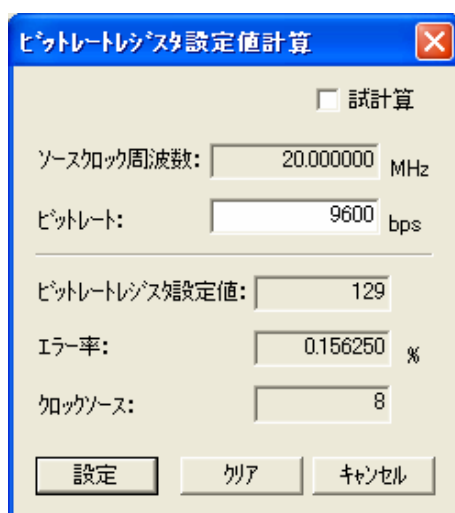


図4.2.2 ビットレートレジスタ設定値計算ダイアログ

[ビットレートレジスタ設定値計算]ダイアログ設定内容を以下に示します。

[試計算]

内部クロック使用時のみ有効です。チェックするとメインクロックの周波数が変更可能になります。設定したボーレート値でエラー率が高くなる場合にチェックすると、入力クロック周波数の変更を仮定したエラー率の試計算が可能です。チェックすると[設定]ボタンは無効になり、設定値を適用することはできません。またチェックを外すとメインクロック周波数は初期値に戻ります。内部クロック選択時に実際に入力クロックの周波数を変更する場合は、CPU クロック設定を変更してください。

[入力クロック]

内部クロック使用時は、CPU クロック設定で設定したシリアル通信モジュールの入力クロック周波数が表示されます。[試計算]をチェックした場合は編集可能となります。

外部クロック使用時は、外部クロックの周波数を入力してください。

[ビットレート]

シリアル通信のボーレートを設定します。変更すると、指定したボーレートに適切なカウントソースとビットレートレジスタ値を選定し、エラー率を算出します。

[ビットレートレジスタ設定値]

入力クロック周波数と指定したボーレート値から算出した最適なビットレートレジスタの値を表示します。

[エラー率]

入力クロック周波数と指定したボーレート値から算出したエラー率を表示します。

[ビットレートレジスタカウントソース]

内部クロック使用時は、入力クロック周波数と指定したボーレート値から選定した最適なカウントソースの分周率が表示されます。

外部クロック使用時は、常に 1 が表示されます。

[割り込み]

送信、受信および SI/O 割り込みを許可し、割り込み発生の検出を行うために設定します。検出する割り込みをチェックし、割り込みレベルを設定すると、割り込み発生時にユーザ作成の割り込み関数が呼び出されます。各項目の設定方法を以下に示します。

[<割り込み種別> 割り込みを許可する]

各割り込み種別に対応した項目をチェックすると、該当の割り込みを検出することができます。各マイコンで設定可能な割り込み種別を表 4.3.1 に示します。

[<割り込み種別> 割り込みレベル]

チェックした割り込みの優先度を設定します。マイコンまたは割り込み種別によっては優先度を設定できません。

各マイコンの割り込み許可設定と通知関数呼び出し条件を表 4.2.9 に示します。

表4.2.9 各マイコンの割り込み許可設定と割り込み種別、通知関数呼び出し条件

マイコン	通信リソース	割り込み許可設定	割り込み関数呼び出し条件
M16C/62p M16C/28,28B,29	UART0~2	送信割り込み	送信完了時
		受信割り込み	受信完了時、受信エラー発生時
R8C/13,22~29 R8C/2A~2D	UARTi ※1	送信割り込み	送信完了時
		受信割り込み	受信完了時、受信エラー発生時
H8/3687, 36077 H8/36049, 36109	SCI3チャンネルi ※2	送信割り込み	送信完了時
		受信割り込み	受信完了時、受信エラー発生時
SH7125	チャンネルi(i=0~2)	送受信割り込み	受信エラー(ERI)時 受信データフル(RXI)時 送信データエンプティ(TXI)時 送信終了(TEI)時

※1 R8C/13,22,23,28,29:i=0、R8C/24,25:i=0~1、R8C/2A~2D:i=0~2

※2 H8/3687,36077:i=1~2、H8/36049,36109:i=1~3

#### [通知関数名]

該当の割り込み発生時に呼び出される通知関数名を指定します。割り込み通知関数を使用する場合は、ユーザプログラムに指定した名前関数を追加してください。通知関数は各シリアル通信リソースで1つ設定可能です。複数の割り込みを指定した場合、全ての割り込み発生時に同じ関数が呼び出されますので、割り込みやエラーの種別に応じて処理を変える場合は、関数内で割り込みやエラーの種別を確認してください。関数の宣言は表 4.2.10 のようになります。

表4.2.10 割り込み関数宣言

マイコン	関数宣言
M16C/62p M16C/28,28B,29 R8C/13,22~29,2A~2D H8/3687,36077,36049,36109	void [設定した通知関数名]( unsigned char notify);
SH7125	void [設定した通知関数名](void);

#### [一括ソース生成をする]

チェックすると、[設定]ボタンクリック時に全周辺 I/O 設定に対応するドライバソースを生成します。

#### [設定]

設定した内容を保存し、ダイアログを閉じます。[一括ソース生成をする]にチェックされている場合はドライバソースを生成します。

#### [キャンセル]

設定した内容を保存せずにダイアログを閉じます。



## 4.2.2 非同期式シリアルインタフェース

非同期式シリアルインタフェースの設定ダイアログを図4.2.3 に示します。

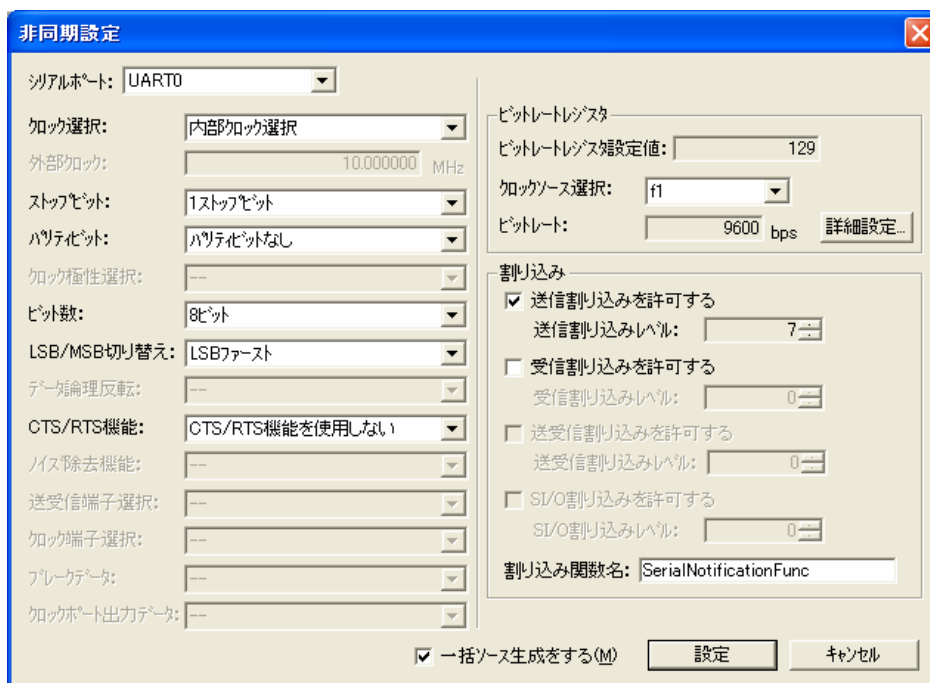


図4.2.3 非同期式シリアルインタフェース設定ダイアログ

同期式と異なる設定は以下の通りです。これらの項目以外の設定については「4.2.1 同期式」を参照してください。

### [ビット数]

転送ビット長を選択します。設定の有無はマイコンにより異なります。[ビット数]の設定内容を表4.2.11に示します。

表4.2.11 [ビット数]設定内容

マイコン	設定	内容
M16C/62p	7ビット	データ長を7ビットに設定します
M16C/28,28B,29	8ビット	データ長を8ビットに設定します
R8C/13,22-29,2A-2D	9ビット	データ長を9ビットに設定します
H8/3687,36077,36049,36109	7ビット	データ長を7ビットに設定します
SH7125	8ビット	データ長を8ビットに設定します

### [ストップビット]

ストップビット長を選択してください。1ストップビットおよび2ストップビットから選択可能です。

### [パリティビット]

パリティの付加とチェックを、パリティビットなし、偶数パリティ、または奇数パリティから選択してください。

### [クロック極性選択]

設定しません。

## [LSB/MSB 切り替え]

転送フォーマットを選択します。設定の有無はマイコンにより異なります。[LSB/MSB切り替え] の設定内容を表4.2.12 に示します。

表4.2.12 [LSB/MSB切り替え]設定内容

マイコン	設定	内容
M16C/62p ※ M16C/28,28B,29 ※	LSB ファースト	転送フォーマットを LSB ファーストにします
R8C/13,22-29,2A-2D ※	MSB ファースト	転送フォーマットを MSB ファーストにします

転送ビット長を 8 ビットに設定した場合に選択可能です

## [ノイズ除去機能]

ノイズ除去機能を設定します。設定の有無はマイコンにより異なります。[ノイズ除去機能] の設定内容を表4.2.13 に示します。

表4.2.13 [ノイズ除去機能]設定内容

マイコン	設定	内容
H8/36049,36109 ※	ノイズ除去機能あり	RXD_3 端子入力のノイズ除去を行います
	ノイズ除去機能なし	ノイズ除去機能を使用しません

SCI3 チャンネル 3 選択時のみ有効です。

## [ブ레이크データ]

シリアル送信終了時のブ레이크出力を設定します。設定の有無はマイコンにより異なります。[ブ레이크データ] の設定内容を表 4.2.14 に示します。

表4.2.14 [ブ레이크データ]設定内容

マイコン	設定	内容
SH7125	ローレベル出力	TXD 端子にローレベルを出力します。
	ハイレベル出力	TXD 端子にハイレベルを出力します。

## [クロックポート出力データ]

クロックポートからの出力を設定します。設定の有無はマイコンにより異なります。[クロックポート出力データ] の設定内容を表 4.2.15.4.2.15 に示します。

表4.2.15. [クロックポート出力データ]設定内容

マイコン	設定	内容
SH7125	ローレベル出力 ※	SCK 端子にローレベルを出力します。
	ハイレベル出力 ※	SCK 端子にハイレベルを出力します
	クロック出力 ※	SCK 端子からビットレートの 16 倍の周波数のクロックを出力します。

外部クロックを使用する場合は本設定にかかわらず、SCK 端子は常に外部クロック(ビットレートの 16 倍の周波数のクロック)入力となります。

### 4.3 I/Oポートの設定

I/Oポートの設定ダイアログを図4.3.1 に示します。



図4.3.1 IOポート設定ダイアログ

各項目の設定方法を以下に示します。

#### [ポート]

設定するポートのグループを指定してください。

SH7125 では設定番号にポートグループが割り当てられています。選択した設定番号のポートグループが表示されます。

#### [使用する端子]

[ポート]で選択したグループの端子が表示されます。設定する端子をチェックしてください。チェックすると対応する端子の設定タブが開きます。

#### 端子設定タブ

##### ・ [端子機能]

SH7125 でのみ設定可能です。複数の機能を持つ端子の使用する機能を選択します。

##### ・ 入力/出力指定

入力ポートとして使用するか、出力ポートとして使用するか指定してください。設定の有無はポートにより異なります。

#### [機能]

ポートの各機能を設定します。

##### [初期設定を High にする]

入出力設定において出力に設定した端子の初期出力を設定します。設定の有無はマイコンにより異なります。設定内容を表4.3.1 に示します。

表4.3.1 初期出力設定内容

マイコン	内容
SH7125	非チェック時: 対応する出力ポートの初期出力を High にします チェック時: 対応する出力ポートの初期出力を制御しません。

## [駆動能力を High にする]

出力トランジスタの駆動能力の設定です。設定の有無はマイコン、ポートにより異なります。設定内容を表4.3.2 に示します。

表4.3.2 出力トランジスタ駆動能力設定内容

マイコン	ポート	内容
R8C/13,26-29	1	非チェック時: 出力トランジスタの駆動能力が Low になります。
R8C/24-25,2A-2D	2	チェック時: 出力トランジスタの駆動能力が High になります。

## [プルアップあり]

端子をプルアップする場合はチェックしてください。設定の有無はマイコン、ポートにより異なります。

## [入力ポート出力ポートに拘らずポートラッチを読む]

ポートレジスタの読み出し設定です。設定の有無はマイコン、ポートにより異なります。設定内容を表4.3.3 に示します。

表4.3.3 ポートレジスタの読み出し設定

マイコン	ポート	内容
M16C/62P M16C/28,28B,29	1	非チェック時: P1 レジスタの読み出し時、入力モードでは端子の入力レベルを読み、出力モードではポートラッチを読みます。 チェック時: P1 レジスタの読み出し時、入力ポート、出力ポートにかかわらず、ポートラッチを読みます。

## [ピンファンクション一覧表示]

SH7125 でのみ有効です。クリックすると、各ポートで選択されている機能の一覧を表示します。

## 4.4 タイマの設定

タイマで設定可能な各モードの内容は次の通りです。

- タイマモード  
内部で生成されたカウントソースをカウントし、割り込みを発生させるモードです。
- イベントカウンタモード  
外部信号、他のタイマのオーバフロー、または他のタイマのアンダフローをカウントするモードです。
- パルス幅変調モード  
任意の幅のパルスを連続して出力するモードです。
- パルス周期測定モード  
外部信号のパルス周期を測定するモードです。
- パルス幅測定モード  
外部信号のパルス幅を測定するモードです。
- インพุットキャプチャモード  
外部または内部信号のエッジを検出して時間を計測するモードです。
- アウトプットコンペアモード  
任意のタイミングで端子出力レベルを変更または割り込みを発生させるモードです。

各マイコンにおいて設定可能なタイマリソースとモードを表4.4.1～表4.4.4に示します。

表4.4.1 設定可能なタイマのモード (N16C/60、M16C/Tinyシリーズ)

	M16C/62p	M16C/28,28B,29
タイマモード	A0-4、B0-5	A0-4、B0-2
イベントカウンタモード	A0-4、B0-5	A0-4、B0-2
パルス幅変調モード	A0-4	A0-4
パルス周期測定モード	B0-5	B0-2
パルス幅測定モード	B0-5	B0-2
インพุットキャプチャモード	—	S
アウトプットコンペアモード	—	S

表4.4.2 設定可能なタイマのモード (R8C/Tinyシリーズ)

	R8C/13	R8C/22-25	R8C/26-29	R8C/2A-2D
タイマモード	X,Y,Z	RA,RB	RA,RB	RA,RB
イベントカウンタモード	X,Y	RA	RA	RA
パルス幅変調モード	Y,Z	RB	RB	RB
パルス周期測定モード	X	RA	RA	RA
パルス幅測定モード	X	RA	RA	RA
インพุットキャプチャモード	C	RD0-1	RC	RC,RD0-1,RF
アウトプットコンペアモード	C	RD0-1,RE	RC,RE	RC,RD0-1,RE

表4.4.3 設定可能なタイマのモード (H8/300H Tinyシリーズ)

	H8/3687	H8/36077	H8/36049	H8/36109
タイマモード	B1,V	B1,V	B1,V	B1,V
イベントカウンタモード	B1,V	B1,V	B1,V	B1,V
パルス幅変調モード	V	V	V	V
パルス周期測定モード	Z0,Z1	Z0,Z1	W,Z0,Z1	RC,RD0-3
パルス幅測定モード	Z0,Z1	Z0,Z1	W,Z0,Z1	RC,RD0-3
インพุットキャプチャモード	Z0,Z1	Z0,Z1	W,Z0,Z1	RC,RD0-3
アウトプットコンペアモード	Z0,Z1	Z0,Z1	W,Z0,Z1	RC,RD0-3

表4.4.4 設定可能なタイマのモード (SH/Tinyシリーズ)

	SH7125
タイマモード	チャンネル 0-5
イベントカウンタモード	チャンネル 0-4
パルス幅変調モード	チャンネル 0-4
パルス周期測定モード	チャンネル 0-5
パルス幅測定モード	チャンネル 0-5
インプットキャプチャモード	チャンネル 0-5
アウトプットコンペアモード	チャンネル 0-4

各モードの設定方法を以下に示します。

#### 4.4.1 タイマモード

タイマモードの設定が可能な各マイコンのタイマリソースを表4.4.5 に示します。

表4.4.5 タイマモードに対応する各マイコンのタイマリソース

シリーズ	グループ	タイマリソース
N16C/60	M16C/62p	A0-4、B0-5
M16C/Tiny	M16C/28,28B,29	A0-4、B0-2
R8C/Tiny	R8C/13	X,Y,Z
	R8C/22-29, 2A-2D	RA,RB
H8/300H Tiny	H8/3687, 36077, 36049, 36109	B1,V
SH/Tiny	SH7125	チャンネル 0-5

各マイコンでのタイマモード設定の概要は次の通りです。

- M16C/62P,M16C/28,28B,29, R8C/13,22-29,2A-2D  
 カウンタをダウンカウントし、アンダフローするとリロードレジスタの値をリロードしてダウンカウントを再開します。リロードレジスタの値は指定した周期とカウントソースの周波数より算出されます。アンダフロー時のアンダフロー割り込みを検出することができます。
- H8/3687,36049,36077,36109  
 タイマ B1 またはタイマ V のカウンタを内部のクロックによりカウントアップします。  
 タイマ B1 はインターバルタイマまたはオートリロードタイマの動作が選択可能です。インターバルタイマ動作時、カウンタは 0 からカウントアップし、カウンタ値が H'FF になった後にカウントソースクロックが入力されるとオーバーフローします。オーバーフローの周期はカウントソースの周波数で決まります。オートリロードタイマ動作時はタイマロードレジスタの値がカウンタにロードされ、その値からカウントアップします。タイマロードレジスタの値は設定したオーバーフロー周期とのカウントソースの周波数より算出されます。いずれの動作でもオーバーフロー割り込みを検出することができます。  
 タイマ V 使用時は 0 からオーバーフローするまでカウントアップします。オーバーフローの周期はカウントソースの周波数で決まります。オーバーフロー割り込みの検出が可能です。
- SH7125  
 内部クロックでカウンタをアップカウントし、SH7125 の基本動作であるフリーランニング動作または周期カウンタ動作を行います。  
 フリーランニング動作ではカウンタオーバーフローまでカウントアップします。オーバーフローの周期はカウントソースの周波数で決まります。オーバーフロー割り込みを検出することができます。  
 周期動作ではカウンタと指定したジェネラルレジスタのコンペアマッチでカウンタがクリアされ、カウントアップを再開します。コンペアマッチ時、コンペアマッチ割り込みの検出と、端子からの任意の出力が可能です。ジェネラルレジスタ値は設定した周期とカウントソースの周波数より算出されます。

タイマモードの設定ダイアログを 図4.4.1 に示します。

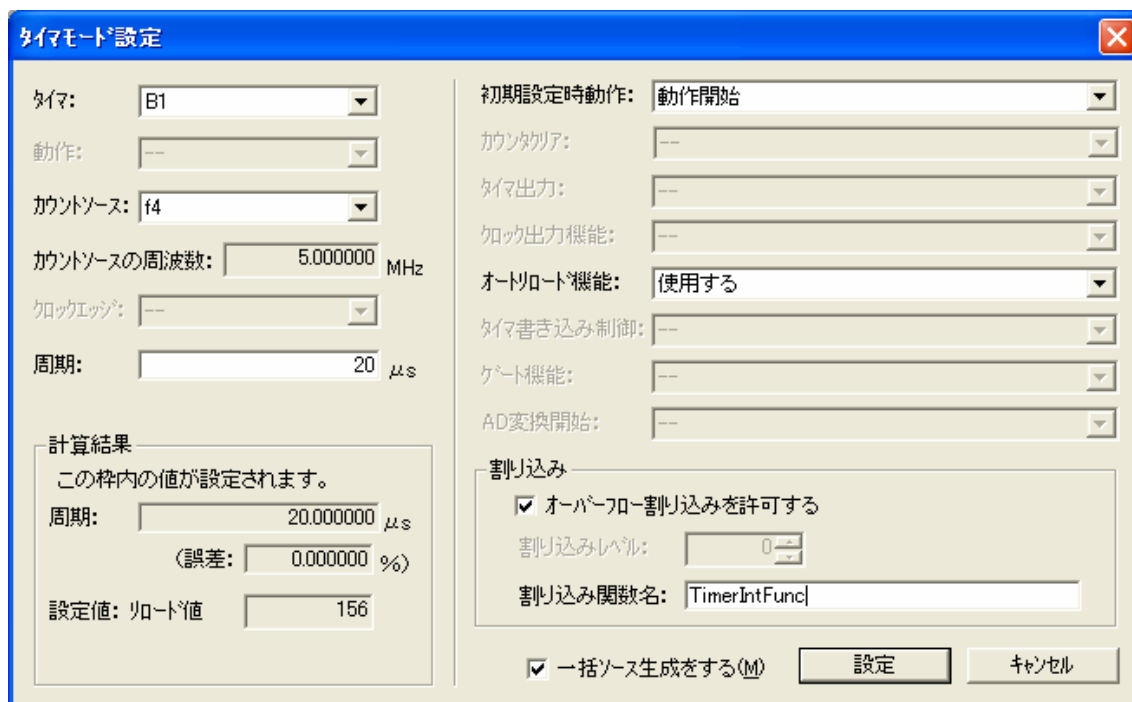


図4.4.1 タイマモード設定ダイアログ

#### [タイマ]

設定するタイマリソースを指定してください。[設定なし]を選択すると、リソースを指定しない設定を作成し、後からリソースを割り当てることができます。ただし SH7125 では[設定なし]を選択できません。

#### [動作]

タイマモードでの動作を選択します。設定の有無はマイコンにより異なります。[動作] の設定内容を表 4.4.6.に示します。

表4.4.6 [動作]設定内容

マイコン	設定	内容
SH7125	フリーラン※	カウントアップを続け、オーバフローを発生させます。周期は指定できません。オーバフローの周期はカウントソースの周波数で決まります。
	周期動作	ジェネラルレジスタとのコンペアマッチによりカウンタをクリアします。指定した周期でコンペアマッチを発生させることができます。

チャンネル 0~4 選択時のみ設定可能です。

#### [カウントソース]

カウンタのカウントソースを選択してください。

#### [カウントソースの周波数]

選択したカウントソースの周波数が表示されます。

#### [クロックエッジ]

カウントに使用するクロックのエッジを選択してください。設定の有無はマイコンにより異なります。

#### [周期]

オーバフロー、アンダフロー、またはコンペアマッチの周期を入力してください。

#### [計算結果]

選択したカウントソースと入力した周期の値に対し、適切なレジスタ値と実際の周期、誤差率の計算結



果を表示します。

・ [周期]

指定した周期に対して算出したレジスタ値を適用した場合の実際の周期が表示されます。

・ [誤差]

指定した周期に対する実際の周期の誤差を表示します。

・ [設定値]

指定した周期に適した各レジスタの値の計算結果を表示します。

[初期設定動作]

タイマを初期設定した直後の動作を[動作開始]、[動作停止]から選択してください。選択できる項目はマイコン、タイマリソースにより異なります。

[カウンタクリア]

カウンタのクリア要因を選択します。設定の有無はマイコンにより異なります。

[タイマ出力]

タイマに関連する端子の設定です。設定の有無、選択項目はマイコンにより異なります。[タイマ出力]の設定内容を表 4.4.7 に示します。

表4.4.7 [タイマ出力]設定内容

マイコン	設定	内容
M16C/62p ※1	パルス出力なし	TAiOUT 端子を入出力ポートとして使用
M16C/28 ※1	パルス出力あり	TAiOUT 端子をパルス出力端子として使用
M16C/28B ※1		
M16C/29 ※1		
SH7125※2	出力保持	TIOCij(※4)端子は出力を保持
	初期出力 0、コンペアマッチ 0 出力※3	TIOCij(※4)端子は初期出力 0、コンペアマッチ時 0 出力
	初期出力 0、コンペアマッチ 1 出力※3	TIOCij(※4)端子は初期出力 0、コンペアマッチ時 1 出力
	初期出力 0、コンペアマッチトグル出力※3	TIOCij(※4)端子は初期出力 0、コンペアマッチ時トグル出力
	初期出力 1、コンペアマッチ 0 出力※3	TIOCij(※4)端子は初期出力 1、コンペアマッチ時 0 出力
	初期出力 1、コンペアマッチ 1 出力※3	TIOCij(※4)端子は初期出力 1、コンペアマッチ時 1 出力
	初期出力 1、コンペアマッチトグル出力※3	TIOCij(※4)端子は初期出力 1、コンペアマッチ時トグル出力

※1 タイマ A 選択時のみ設定可能です

※2 チャンネル 0~4 選択時のみ設定可能です

※3 [動作]で周期動作を選択したときのみ選択可能です

※4 i=0~4 j=A~D

[クロック出力機能]

設定しません。

[オートリロード機能]

オートリロード機能を設定します。設定の有無はマイコンにより異なります。オートリロード機能の設定内容を表 4.4.8 に示します。

表4.4.8 [オートリロード機能]設定内容

マイコン	設定	内容
H8/3687	使用しない	インターバルタイマとして動作します。カウンタは 0 からカウントアップし、オーバフローまでの時間はカウントソースの周波数により決まります。[周期]で指定した値は実際のオーバフロー周期とは異なります。
H8/36077		
H8/36049	使用する	オートリロードタイマとして動作します。指定した周期に基づく値をロードレジスタに設定し、[計算]-[周囲]に表示されている間隔でオーバフローが発生します。
H8/36109		



## [タイマ書き込み制御]

レジスタへの書き込みに関する設定です。設定の有無はマイコン、タイマリソースにより異なります。タイマ書き込みの設定内容を表 4.4.9 に示します。

表4.4.9 [タイマ書き込み制御]設定内容

マイコン	設定	内容
R8C/13 ※1 R8C/22-29 ※2 R8C/2A-2D ※2	リロードレジスタのみ書き込み	プリスケアラレジスタとプライマリレジスタに値を書き込んだ際、リロードレジスタにのみ値が書き込まれます。
	リロードレジスタとカウンタへの書き込み	プリスケアラレジスタとプライマリレジスタに値を書き込んだ際、リロードレジスタとカウンタに値が書き込まれます。

※1 タイマ Y,Z

※2 タイマ RA,RB

## [ゲート機能]

ゲート機能の設定です。設定の有無はマイコン、タイマリソースにより異なります。ゲート機能の設定内容を表 4.4.10 に示します。

表4.4.10 [ゲート機能]設定内容

マイコン	設定	内容
M16C/62P ※ M16C/28-29 ※	使用しない	ゲート機能を使用しません。TailN 端子の状態によらずダウンカウントされます。
	TailN 端子に”L”が入力されている間カウントする	TailN 端子に”L”が入力されている期間ダウンカウントされます。
	TailN 端子に”H”が入力されている間カウントする	TailN 端子に”H”が入力されている期間ダウンカウントされます。

タイマ A 選択時のみ設定可能です。

## [AD 変換開始]

AD 変換開始機能を設定します。設定の有無はマイコンにより異なります。AD 変換開始機能の設定内容を表 4.4.11 に示します。

表4.4.11. [AD変換開始]設定内容

マイコン	設定	内容
SH7125	A/D 変換開始を禁止する	TGRA のコンペアマッチによる A/D 変換器開始要求の発生を禁止します。
	A/D 変換開始を許可する	TGRA のコンペアマッチによる A/D 変換器開始要求の発生を許可します。TGRA のコンペアマッチで A/D 変換を開始するには A/D 設定でトリガを TGRA に設定してください。

## [割り込み]

オーバフロー割り込み、アンダフロー割り込み、またはコンペアマッチ割り込みを許可し、割り込み発生を検出を行うために設定します。検出する割り込みをチェックし、割り込みレベルを設定すると、割り込み発生時にユーザ作成の割り込み関数が呼び出されます。各項目の設定方法を以下に示します。

- ・ [ <割り込み種別> 割り込みを許可する ]

各割り込み種別に対応した項目をチェックすると、該当の割り込みを検出することができます。タ

イマの動作により検出可能な割り込み種別は異なります。

・ [ <割り込み種別> 割り込みレベル]

チェックした割り込みの優先度を設定します。マイコンまたは割り込み種別によっては優先度を設定できません。

・ [割り込み通知関数名]

該当の割り込み発生時に呼び出される通知関数名を指定します。割り込み通知関数を使用する場合は指定した名前に関数を追加してください。割り込み関数の宣言は次のようになります。

関数宣言

```
void [設定した通知関数名](void);
```

[一括ソース生成をする]

チェックすると、[設定]ボタンクリック時にドライバソースを生成します。

[設定]

設定した内容を保存し、ダイアログを閉じます。[一括ソース生成をする]にチェックされている場合はドライバソースを生成します。

[キャンセル]

設定した内容を保存せずにダイアログを閉じます。

#### 4.4.2 イベントカウンタモード

イベントカウンタモードの設定が可能な各マイコンのタイマリソースを表 4.4.12 に示します。

表4.4.12 イベントカウンタモードに対応する各マイコンのタイマリソース

シリーズ	グループ	タイマリソース
N16C/60	M16C/62p	A0-4、B0-5
M16C/Tiny	M16C/28,28B,29	A0-4、B0-2
R8C/Tiny	R8C/13	X,Y
	R8C/22-29, 2A-2D	RA
H8/300H Tiny	H8/3687, 36077, 36049, 36109	B1,V
SH/Tiny	SH7125	チャネル 0-4

各マイコンでのイベントカウンタモード設定の概要は次の通りです。

- M16C/62P,M16C/28,28B,29

タイマ A またはタイマ B のカウンタを、他のタイマのオーバフロー、他のタイマのアンダフロー、または外部信号の指定したエッジでカウントします。

タイマ A はアップカウントとダウンカウントの選択が可能です。カウンタのアンダフローまたはオーバフローで割り込みが発生します。オーバフローまたはアンダフロー時にリロードレジスタの値をカウンタにロードするかしないかを選択できます。リロードレジスタの値は任意に指定可能です。

タイマ B はダウンカウントによるアンダフロー割り込みを検出できます。アンダフロー時にリロードレジスタの値がカウンタにロードされます。リロードレジスタの値は任意に指定可能です。

- R8C/13

カウンタを CNTR0 端子に入力された外部信号の指定したエッジでダウンカウントします。アンダフロー割り込みの検出が可能です。アンダフロー割り込み時、リロードレジスタの値をリロードしてカウントを継続します。リロードレジスタの値は任意に指定可能です。

- R8C/22-29,2A-2D

カウンタを TRAI0 端子に入力された外部信号の指定したエッジでダウンカウントします。アンダフロー割り込みの検出が可能です。アンダフロー割り込み時、リロードレジスタの値をリロードしてカウントを継続します。リロードレジスタの値は任意に指定可能です。

- H8/3687,36049,36077,36109

タイマ B1 またはタイマ V のカウンタを外部信号の指定したエッジでカウントアップします。

タイマ B1 はインターバルタイマまたはオートリロードタイマの動作が選択可能です。インターバルタイマ動作時、カウンタは 0 からカウントアップし、カウンタ値が H'FF になった後にカウントソースが入力されるとオーバフローします。オートリロードタイマ動作時はタイマロードレジスタの値がカウンタにロードされ、カウントアップします。タイマロードレジスタの値は任意に指定可能です。いずれの動作でもオーバフロー割り込みを検出することができます。

タイマ V 使用時は 0 からオーバフローするまでカウントアップします。オーバフロー割り込みの検出が可能です。

- SH7125

外部クロックでカウンタをアップカウントし、SH7125 の基本動作であるフリーランニング動作または周期カウンタ動作を行います。

フリーランニング動作ではカウンタオーバフローまでカウントアップします。オーバフローの周期はカウントソースの周波数で決まります。オーバフロー割り込みを検出することができます。

周期動作ではカウンタと指定したジェネラルレジスタのコンペアマッチでカウンタがクリアされ、カウントアップを再開します。コンペアマッチ時、コンペアマッチ割り込みの検出と、端子からの任意の出力が可能です。ジェネラルレジスタ値は設定した周期とカウントソースの周波数より算出されます。

チャンネル 1 および 2 では位相計数モードによる 2 端子(チャンネル 1 の場合 TCLKA と TCLKB、チャンネル 2 の場合 TCLKC と TCLKD )の位相差によるカウントアップまたはカウントダウン動作を選択可能です。この場合、カウンタのオーバーフロー割り込みおよびアンダフロー割り込みの検出が可能です。

イベントカウンタモードの設定ダイアログを図 4.4.2 に示します。

図4.4.2 イベントカウンタモード設定ダイアログ

各項目の設定内容を以下に示します。

#### [タイマ]

設定するタイマリソースを指定してください。[設定なし]を選択すると、リソースを指定しない設定を作成し、後からリソースを割り当てることができます。ただし SH7125 では[設定なし]を選択できません。

#### [動作]

イベントカウンタモードの動作を選択します。設定の有無はマイコンにより異なります。[動作]の設定内容を表 4.4.13 に示します。

表4.4.13 [動作]設定内容

マイコン	設定	内容
SH7125	フリーラン	カウントアップを続け、オーバーフローを発生させます。
	周期動作	ジェネラルレジスタとのコンペアマッチによりカウンタをクリアします。
	位相計数モード※	チャンネル 1 の場合 TCLKA と TCLKB、チャンネル 2 の場合 TCLKC と TCLKD の位相差を検出し、カウントアップまたはカウントダウンします。

※チャンネル 1,2 選択時のみ設定可能です。

## [位相計数モード]

SH7125 のチャンネル 1 または 2 で[動作]に位相計数モードを指定したときのみ有効です。各位相計数モードの設定内容を表 4.4.14 に示します。

表4.4.14 [動作]設定内容

マイコン	設定	内容
SH7125※	位相計数モード 1	モードを指定してください。各モードのカウント条件を表 4.4.15 に示します。
	位相計数モード 2	
	位相計数モード 3	
	位相計数モード 4	

チャンネル 1 または 2 で位相計数モード選択時のみ設定可能です

表4.4.15 各位相計数モードのカウント条件 (SH7125)

モード	TCLKA(チャンネル 1) TCLKC(チャンネル 2)	TCLKB(チャンネル 1) TCLKD(チャンネル 2)	動作
位相計数モード 1	High レベル	立ち上がりエッジ	アップカウント
	Low レベル	立ち下がりエッジ	
	立ち上がりエッジ	Low レベル	
	立ち下がりエッジ	High レベル	
	High レベル	立ち下がりエッジ	ダウンカウント
	Low レベル	立ち上がりエッジ	
	立ち上がりエッジ	High レベル	
	立ち下がりエッジ	Low レベル	
位相計数モード 2	High レベル	立ち上がりエッジ	カウントしない
	Low レベル	立ち下がりエッジ	カウントしない
	立ち上がりエッジ	Low レベル	カウントしない
	立ち下がりエッジ	High レベル	アップカウント
	High レベル	立ち下がりエッジ	カウントしない
	Low レベル	立ち上がりエッジ	カウントしない
	立ち上がりエッジ	High レベル	カウントしない
	立ち下がりエッジ	Low レベル	アップカウント
位相計数モード 3	High レベル	立ち上がりエッジ	カウントしない
	Low レベル	立ち下がりエッジ	カウントしない
	立ち上がりエッジ	Low レベル	カウントしない
	立ち下がりエッジ	High レベル	アップカウント
	High レベル	立ち下がりエッジ	ダウンカウント
	Low レベル	立ち上がりエッジ	カウントしない
	立ち上がりエッジ	High レベル	カウントしない
	立ち下がりエッジ	Low レベル	カウントしない
位相計数モード 4	High レベル	立ち上がりエッジ	アップカウント
	Low レベル	立ち下がりエッジ	カウントしない
	立ち上がりエッジ	Low レベル	
	立ち下がりエッジ	High レベル	
	High レベル	立ち下がりエッジ	ダウンカウント
	Low レベル	立ち上がりエッジ	
	立ち上がりエッジ	High レベル	カウントしない
	立ち下がりエッジ	Low レベル	

## [カウントソース]

カウント対象のイベントを選択してください。

## [カウントソースの周波数]

設定しません。

## [クロックエッジ]

イベントが外部信号の場合、カウントに使用するクロックのエッジを選択してください。

## [設定値]

各レジスタに設定する値を入力してください。設定内容はマイコンにより異なります。各マイコンの設定内容を表 4.4.16 に示します。

表4.4.16 各マイコンの設定値内容

マイコン	設定値	内容
M16C/62p	タイマ値	カウンタリロード値
M16C/28,28B,29	タイマ値	カウンタリロード値
R8C/13	プリスケアラ値	プリスケアラ(X または Y)レジスタリロード値
	タイマ値	タイマ(X または Y)レジスタリロード値
R8C/22-29, 2A-2D	プリスケアラ値	タイマ RA プリスケアラレジスタリロード値
	タイマ値	タイマ RA レジスタリロード値
H8/3687, 36077, 36049, 36109※1	リロード値	カウンタリロード値
SH7125 ※2	比較値	周期動作時のジェネラルレジスタ値

※1 タイマ B1 選択時のみ設定可能です

※2 周期動作選択時のみ設定可能です

## [割り込み]

オーバフロー割り込み、アンダフロー割り込み、またはコンペアマッチ割り込みを許可し、割り込み発生の検出を行うために設定します。検出する割り込みをチェックし、割り込みレベルを設定すると、割り込み発生時にユーザ作成の割り込み関数が呼び出されます。各項目の設定方法を以下に示します。

- ・ [ <割り込み種別> 割り込みを許可する ]

各割り込み種別に対応した項目をチェックすると、該当の割り込みを検出することができます。タイマの動作により検出可能な割り込み種別は異なります。

- ・ [ <割り込み種別> 割り込みレベル ]

チェックした割り込みの優先度を設定します。マイコンまたは割り込み種別によっては優先度を設定できません。

- ・ [割り込み通知関数名]

該当の割り込み発生時に呼び出される通知関数名を指定します。割り込み通知関数を使用する場合は指定した名前関数を追加してください。割り込み関数の宣言は次のようになります。

## 関数宣言

```
void [設定した通知関数名](void);
```

## [初期設定動作]

タイマをダイアログで指定した状態に初期設定した直後の動作を[動作開始]、[動作停止]から選択してください。選択できる項目はマイコンとタイマリソースにより異なります。

## [カウンタクリア]

カウンタのクリア要因を選択します。設定の有無や選択項目は、マイコン、タイマの動作により異なります。

## [出力機能]

タイマに関連する端子の設定です。設定の有無、選択項目はマイコンにより異なります。[出力機能]の設定内容を表 4.4.17 に示します。

表4.4.17 [出力機能]設定内容

マイコン	設定	内容
M16C/62p ※1	パルス出力なし	TAiOUT 端子を入出力ポートとして使用
M16C/28 ※1 M16C/28B ※1 M16C/29 ※1	パルス出力あり	TAiOUT 端子をパルス出力端子として使用
R8C/22-29, R8C/2A-2D	パルス出力なし	TRAO 端子を入出力ポートとして使用
	パルス出力あり	TRAO 端子をパルス出力端子として使用
SH7125 ※2	出力なし	TIOcij(※4)端子は出力を保持
	初期出力 0、コンペアマッチ 0 出力※3	TIOcij(※4)端子は初期出力 0、コンペアマッチ時 0 出力
	初期出力 0、コンペアマッチ 1 出力※3	TIOcij(※4)端子は初期出力 0、コンペアマッチ時 1 出力
	初期出力 0、コンペアマッチトグル出力※3	TIOcij(※4)端子は初期出力 0、コンペアマッチ時トグル出力
	初期出力 1、コンペアマッチ 0 出力※3	TIOcij(※4)端子は初期出力 1、コンペアマッチ時 0 出力
	初期出力 1、コンペアマッチ 1 出力※3	TIOcij(※4)端子は初期出力 1、コンペアマッチ時 1 出力
	初期出力 1、コンペアマッチトグル出力※3	TIOcij(※4)端子は初期出力 1、コンペアマッチ時トグル出力

※1 タイマ A 選択時のみ設定可能です

※2 チャンネル 0~4 選択時のみ設定可能です

※3 [動作]で周期動作を選択したときのみ選択可能です

※4 i:0~4(チャンネル 0~4 に対応) j:A~D(TGRA~TGRD に対応)

#### [オートリロード機能]

オートリロード機能を設定します。設定の有無はマイコンにより異なります。オートリロード機能の設定内容を表 4.4.18 に示します。

表4.4.18 [オートリロード機能]設定内容

マイコン	設定	内容
M16C/62p ※1 M16C/28,28B,29 ※1	使用しない	オーバフローまたはアンダフローが発生してもリロードレジスタからリロードしません。
H8/3687,36077,36049,36109 ※2	使用する	オーバフローまたはアンダフロー時は、リロードレジスタの内容をリロードしてカウントを継続します。

※1 タイマ A 選択時のみ設定可能です

※2 タイマ B1 選択時のみ設定可能です

#### [アップ/ダウン切り替え要因]

カウント方向の指定方法を選択します。設定の有無はマイコンにより異なります。アップ/ダウン切り替え要因の設定内容を表 4.4.19 に示します。

表4.4.19 [アップ/ダウン切り替え要因]設定内容

マイコン	設定	内容
M16C/62pi※ M16C/28,28B,29※	UDF レジスタ	UDF レジスタ値によりカウント方向を指定します。選択するとプログラムによる切り替えが可能となります。後述の[カウント方向]により、アップカウントまたはダウンカウントの指定ができます。
	TAiOUT 端子への入力信号	外部信 TAiOUT によりカウント方向を指定します。TAiOUT 端子に“L”が入力されているときダウンカウント、“H”が入力されているときアップカウントします。

タイマ A 選択時のみ設定可能です



## [外部イベント極性]

イベントが外部信号の場合、カウントに使用するクロックのエッジを立ち上がりエッジ、立ち下がりエッジ、または両エッジから選択してください。設定の有無と選択項目は CPU、選択したタイマリソースにより異なります。

## [タイマ書き込み制御]

レジスタへの書き込みに関する設定です。設定の有無はマイコン、選択したタイマリソースにより異なります。タイマ書き込みの設定内容を表 4.4.20 に示します。

表4.4.20 [タイマ書き込み制御]設定内容

マイコン	設定	内容
R8C/13 ※	リロードレジスタのみ書き込み	プリスケアラレジスタとタイマレジスタに値を書き込んだ際、リロードレジスタにのみ値が書き込まれます。
	リロードレジスタとカウンタへの書き込み	プリスケアラレジスタとタイマレジスタに値を書き込んだ際、リロードレジスタとカウンタに値が書き込まれます。
R8C/22-29 ※ R8C/2A-2D ※	リロードレジスタとカウンタへの書き込み	プリスケアラレジスタとタイマレジスタに値を書き込んだ際、リロードレジスタとカウンタに値が書き込まれます。

タイマ Y 選択時のみ設定可能

## [フィルタ機能]

フィルタ機能を使用する場合にフィルタのサンプリングを選択してください。設定の有無はマイコンにより異なります。

## [入力端子]

外部信号の入力端子を選択してください。設定の有無はマイコンにより異なります。

## [一括ソース生成をする]

チェックすると、[設定]ボタンクリック時にドライバソースを生成します。

## [設定]

設定した内容を保存し、ダイアログを閉じます。[一括ソース生成をする]にチェックされている場合はドライバソースを生成します。

## [キャンセル]

設定した内容を保存せずにダイアログを閉じます。

### 4.4.3 パルス幅変調モード

パルス幅変調モードの設定が可能な各マイコンのタイマリソースを表 4.4.21 に示します。

表4.4.21 パルス幅変調モードに対応する各マイコンのタイマリソース

シリーズ	グループ	タイマリソース
N16C/60	M16C/62p	A0-4
M16C/Tiny	M16C/28,28B,29	A0-4
R8C/Tiny	R8C/13	Y,Z
	R8C/22-29, 2A-2D	RB
H8/300H Tiny	H8/3687, 36077, 36049, 36109	V
SH/Tiny	SH7125	チャンネル 0-4

各マイコンでのパルス幅変調モード設定の概要は次の通りです。

- M16C/62P,M16C/28,28B,29

タイマ A のパルス幅変調モード(PWM モード)で TAiOUT 端子より PWM 波形を連続出力します。16 ビットパルス幅変調器、8 ビットパルス幅変調器のいずれかのパルス幅変調器として動作させます。指定した周期と“H”幅の波形を生成します。PWM パルスの立ち下がり時(“H” 期間終了時)の割り込みを検出することができます。

- R8C/13

タイマ Y またはタイマ Z のプログラマブル波形発生モードにより PWM 波形を連続出力します。プログラマブル波形発生モードではタイマプライマリレジスタとタイマセカンダリレジスタの値を交互にカウントし、カウンタがアンダフローするごとに CNTR1 端子または TZOUT 端子から出力する信号を反転します。タイマプライマリレジスタとタイマセカンダリレジスタの値は設定した周期とデューティより自動的に算出されます。セカンダリ期間の終了時に発生するタイマ Y 割り込みまたはタイマ Z 割り込みを検出できます。

- R8C/22-29,2A-2D

タイマ RB のプログラマブル波形発生モードにより PWM 波形を連続出力します。プログラマブル波形発生モードではタイマプライマリレジスタとタイマセカンダリレジスタの値を交互にカウントし、カウンタがアンダフローするごとに TRBO 端子から出力する信号を反転します。タイマプライマリレジスタとタイマセカンダリレジスタの値は設定した周期とデューティより自動的に算出されます。セカンダリ期間の終了時に発生するタイマ RB 割り込みを検出できます。

- H8/3687,36049,36077,36109

タイマ V のタイムコンスタントレジスタ A、B とカウンタとのコンペアマッチで出力を制御することにより任意のデューティパルスを生成します。タイムコンスタントレジスタ A、B の値は設定した周期とデューティより自動的に算出されます。本設定ではタイムコンスタントレジスタ B が周期レジスタとなります。タイムコンスタントレジスタ A、B のコンペアマッチ割り込み (CMFB および CMFA) を検出できます。

- SH7125

SH7125 の PWM モードにより出力端子から PWM 波形を出力します。各 TGR のコンペアマッチによる出力レベルを 0 出力 / 1 出力 / トグル出力から選択することにより、デューティ 0 ~ 100% の PWM 波形を最大 8 相生成できます。

パルス幅変調モードの設定ダイアログは SH7125 とその他の機種では設定方法が異なります。それぞれのマイコンでの設定方法を以下に示します。

・ SH7125

SH7125 のパルス幅変調モードの設定ダイアログを図 4.4.3 に示します。

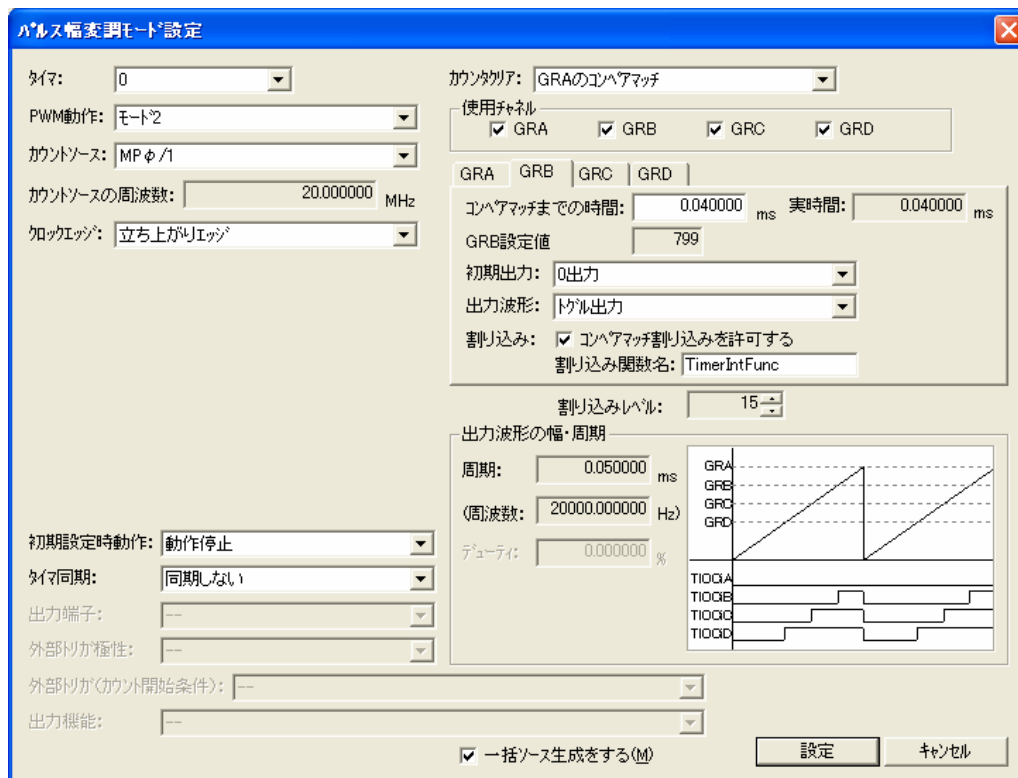


図4.4.3 パルス幅変調モード設定ダイアログ (SH7125)

注意: ダイアログおよび以下の説明で表記するジェネラルレジスタ名 GRA、GRB、GRC、GRD は、SH7125 の TGRA、TGRB、TGRC、TGRD に対応します。

[タイマ]

設定するタイマリソースを指定してください。

[カウントソース]

カウンタのカウントソースを選択してください。

[カウントソースの周波数]

[カウントソース]で選択したクロックの周波数が表示されます。

[クロックエッジ]

カウントに用いるカウントソースのエッジを指定してください。

[PWM 動作]

PWM モードを選択してください。PWM 動作の設定内容を表 4.4.22 に示します。

表4.4.22 [PWM動作]設定内容

設定	内容
モード 1	TGRAとTGRB、TGRCとTGRD をペアで使用して、TIOCA 端子、TIOCC 端子からPWM出力を生成します。TIOCA 端子と TIOCC 端子の初期値は GRA、GRC で設定した値となります。TGRA と TGRB で設定したコンペアマッチ時出力値は TIOCA 端子に、TGRC と TGRD で設定したコンペアマッチ時出力値は TIOCC 端子に出力されます。カウンタクリア要因の TGR 値が周期レジスタとなります。チャンネル間の独立動作/同期動作が可能です。
モード 2	TGR の 1 本を周期レジスタ、他の TGR をデューティレジスタとして使用し、PWM 出力を生成します。各 TGR で設定したコンペアマッチ時出力値は各 TGR に対応する端子に出力されます。周期レジスタとして指定した TGR とのコンペアマッチ時にカウンタがクリアされ、各端子の値はそれぞれの TGR で設定した初期値となります。チャンネル間の独立動作/同期動作が可能です。

## [初期設定動作]

タイマをダイアログで指定した状態に初期設定した直後の動作を[動作開始]、[動作停止]から選択してください。

## [タイマ同期]

チャンネル間の同期動作を設定します。タイマ同期の設定内容を表 4.4.23 に示します。

表4.4.23 [タイマ同期]設定内容

設定	内容
同期しない	他のチャンネルとは独立して動作します。[カウンタクリア]で設定するカウンタクリア要因には、チャンネル内の TGR が指定されます。
タイマ同期の主となる	他のチャンネルと同期して動作します。[カウンタクリア]で設定するカウンタクリア要因は、同期する他のチャンネルのカウンタクリア要因となります。
同期する	他のチャンネルで[タイマ同期の主となる]が指定されている場合のみ選択可能です。カウンタクリア要因は[タイマ同期の主となる]が指定されているチャンネルの設定に従います。[カウンタクリア]で設定するカウンタクリア要因は同期クリアのみ選択できます。

## [出力端子]

設定しません。

## [外部トリガ極性]

設定しません。

## [外部トリガ(カウント開始条件)]

設定しません。

## [出力機能]

設定しません。

## [カウンタクリア]

カウンタクリア要因を指定します。カウンタクリアの設定内容を表 4.4.24 に示します。

表4.4.24 [カウンタクリア]設定内容

設定	内容
GRA のコンペアマッチ	選択したチャンネルの該当 TGR をカウンタクリア要因とします。[タイマ同期]に[タイマ同期の主となる]を選択した場合、指定した TGR は同期動作する他のチャンネルのカウンタクリア要因となります。
GRB のコンペアマッチ	
GRC のコンペアマッチ ※1	
GRD のコンペアマッチ ※1	
同期クリア ※2	[タイマ同期]に[同期する]を選択した場合は常に選択されます。[タイマ同期]に[タイマ同期の主となる]が設定されている他のチャンネルのカウンタクリア要因に従います。

※1 チャンネル 1、チャンネル 2 選択時は指定できません

※2 [タイマ同期]に[同期する]を選択した場合のみ設定可能です

#### [使用チャンネル]

PWM 波形生成に使用する TGR を選択します。[カウンタクリア]で選択した TGR は常にチェックされます。PWM モード 1 使用時は TGRA と TGRB、TGRC と TGRD は連動します。チェックすると対応する TGR の設定タブが開きます。TGR タブで詳細設定を行います。

#### TGR 設定タブ

[使用チャンネル]でチェックした TGR の設定タブが表示されます。以下の項目を設定し、TGR の設定を行います。

- [コンペアマッチまでの時間]

カウント開始からコンペアマッチまでの時間を入力してください。入力すると設定した時間とカウントソースの周波数から適切な TGR の値を計算します。

- [実時間]

TGR 値とカウントソースの周波数より算出した、カウント開始からコンペアマッチまでの実際の時間が表示されます。

- [GRi 設定値](i: A,B,C,D)

[コンペアマッチまでの時間]で入力した時間とカウントソースの周波数より算出した TGR の値が表示されます。値が範囲外の場合は[設定の変更]をクリックしても設定を適用することはできません。設定を元に生成したドライバソースを使用すると、表示されている値がレジスタに設定されます。

- [初期出力]

各 TGR に対応する端子の初期出力値を指定してください。

- [出力波形]

コンペアマッチ時の各 TGR に対応する端子の出力を指定してください。

- [割り込み]

コンペアマッチ割り込みを許可し、割り込み発生の検出を行うために設定します。チェックして割り込みレベルを設定すると、割り込み発生時にユーザ作成の割り込み関数が呼び出されます。

[コンペアマッチ割り込みを許可する]

チェックするとコンペアマッチ割り込みを検出することができます。

[割り込み通知関数名]

コンペアマッチ割り込み発生時に呼び出される通知関数名を指定します。割り込み通知関数を使用する場合は指定した名前関数を追加してください。割り込み関数の宣言は次のようになります。

関数宣言

```
void [設定した通知関数名](void);
```

## [割り込みレベル]

TGR 設定タブの[割り込み]でコンペアマッチ割り込みを許可した場合、コンペアマッチ割り込みレベルを設定してください。複数の TGR で割り込みの設定を行った場合も、割り込みレベルの設定は共通です。

## [出力波形の幅・周期]

以上の設定により生成される PWM 波形の情報を表示します。

## ・ [周期]

出力波の周期が表示されます。

## ・ [周波数]

出力波の周波数が表示されます。

## ・ [デューティ]

使用しません。

## ・ 波形表示領域

生成される PWM 波形を表示します。表示例を図 4.4.4 および図 4.4.5 に示します。

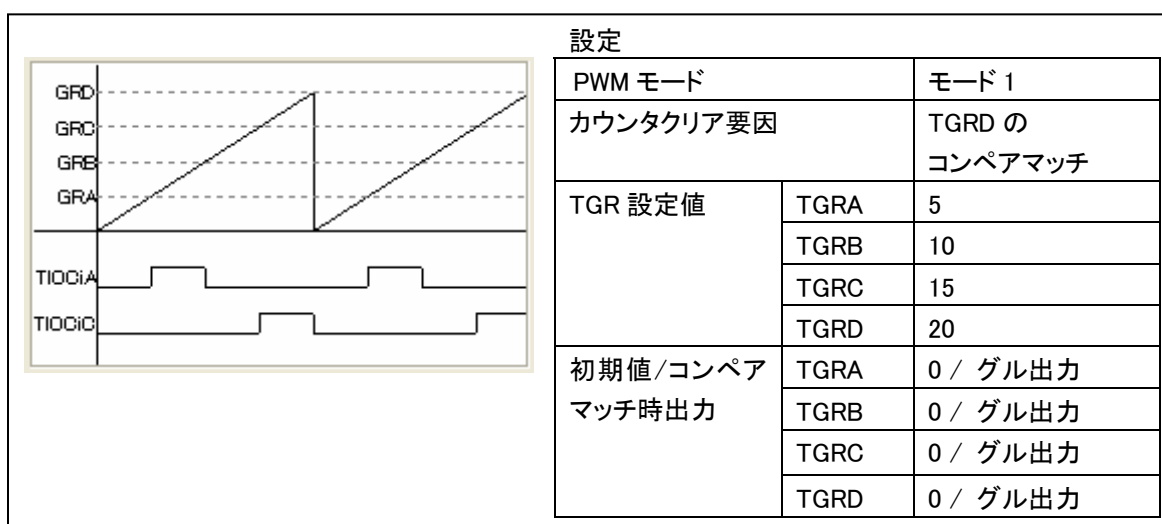


図 4.4.4 PWM波形表示例 (PWMモード1)

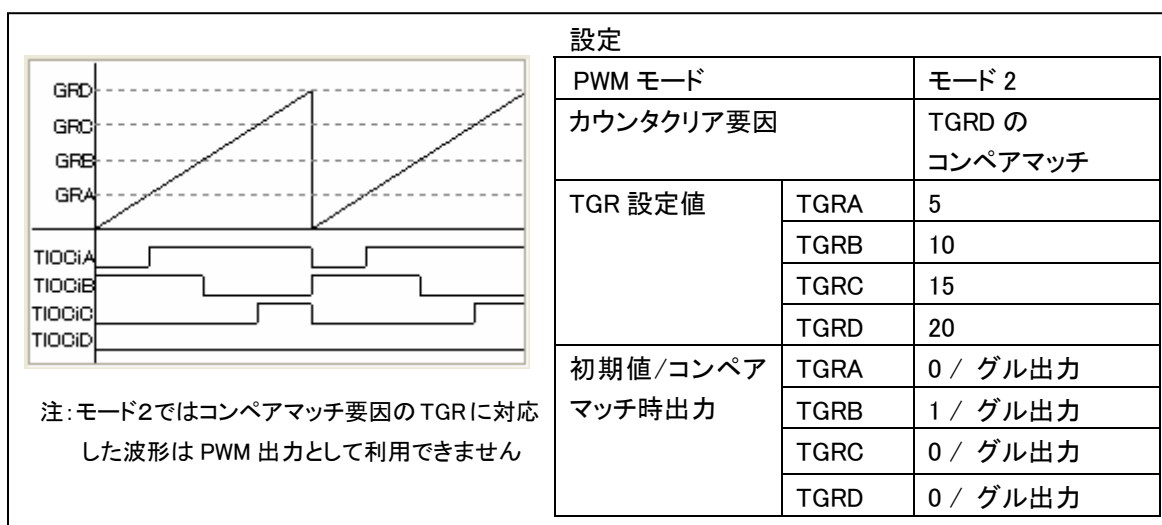


図 4.4.5 PWM波形表示例 (PWMモード2)

## [一括ソース生成をする]

チェックすると、[設定]ボタンクリック時にドライバソースを生成します。

## [設定]

設定した内容を保存し、ダイアログを閉じます。[一括ソース生成をする]にチェックされている場合はドライバソースを生成します。

## [キャンセル]

設定した内容を保存せずにダイアログを閉じます。

・ M16C/60、M16C/Tiny、R8C/Tiny、および H8/300H Tiny

M16C/60、M16C/Tiny、R8C/Tiny、および H8/300H Tiny シリーズのパルス幅変調モードの設定ダイアログを図 4.4.6 に示します。

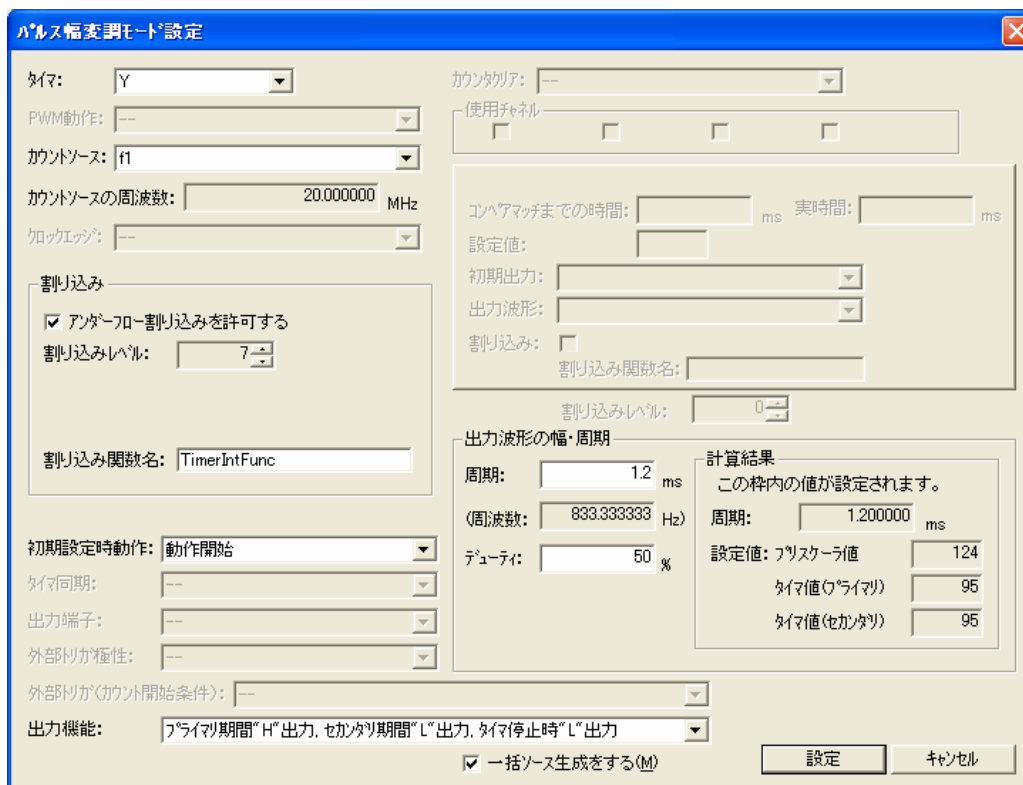


図4.4.6 パルス幅変調モード設定ダイアログ (M16C/60、M16C/Tiny、R8C/Tiny、H8/300H Tiny)

## [タイマ]

設定するタイマリソースを指定してください。[設定なし]を選択すると、リソースを指定しない設定を作成し、後からリソースを割り当てることができます。

## [カウントソース]

カウンタのカウントソースを選択してください。

## [カウントソースの周波数]

[カウントソース]で選択したクロックの周波数が表示されます。

## [クロックエッジ]

設定しません。

## [PWM 動作]

パルス幅変調器の動作を設定します。パルス幅変調器の動作により出力できるパルスが異なります。設定の有無はマイコンにより異なります。表 4.4.25 に PWM 動作の設定内容を示します。



表4.4.25 [PWM動作]設定内容

マイコン	設定	内容
M16C/62P M16C/28,28B,29	16 ビットパルス幅変調器として動作	H 幅、周期は次の通りです。 ・”H”幅 : $n / f_j$ n: Tai レジスタの設定値(i=0~4)※ ・周期 : $(2^{16}-1) / f_j$ 固定 f <sub>j</sub> : カウントソースの周波数(f1、f2、f8、f32、fC32)
	8 ビットパルス幅変調器として動作	H 幅、周期は次の通りです。 ・”H”幅 : $n \times (m+1) / f_j$ n: Tai レジスタの上位番地の設定値※ ・周期 : $(2^8-1) \times (m+1) / f_j$ m: Tai レジスタの下位番地の設定値※

各レジスタ設定値は後述の計算結果で自動算出されます。

## [割り込み]

割り込みを許可し、割り込み発生の検出を行うために設定します。検出する割り込みをチェックし、割り込みレベルを設定すると、割り込み発生時にユーザ作成の割り込み関数が呼び出されます。各項目の設定方法を以下に示します。

## ・ [ &lt;割り込み種別&gt; 割り込みを許可する ]

各割り込み種別に対応した項目をチェックすると、該当の割り込みを検出することができます。マイコンにより検出可能な割り込み種別は異なります。各マイコンの割り込み種別を表 4.4.26 に示します。

表4.4.26 各マイコンの割り込み種別

マイコン	割り込み種別	内容
M16C/62P M16C/28,28B,29	PWM パルス立ち下が り時割り込み	”H”期間が終了し、PWM パルスが立ち下がった時に発行されます
R8C/13 R8C/22-29, R8C/2A-2D	アンダフロー割り込み	セカンダリ期間のアンダフロー時に発行されます。セカンダリ期間終了時の端子出力変化と同時です。
H8/3687,36077 H8/36049,36109	コンペアマッチ A 割り 込み	タイムコンスタントレジスタ A とタイマカウンタの一致時に発行されます
	コンペアマッチ B 割り 込み	タイムコンスタントレジスタ B とタイマカウンタの一致時に発行されます

## ・ [ &lt;割り込み種別&gt; 割り込みレベル ]

チェックした割り込みの優先度を設定します。マイコンまたは割り込み種別によっては優先度を設定できません。

## ・ [割り込み通知関数名]

該当の割り込み発生時に呼び出される通知関数名を指定します。割り込み通知関数を使用する場合は指定した名前関数を追加してください。割り込み関数の宣言は次のようになります。

## 関数宣言

```
void [設定した通知関数名](void);
```

## [初期設定動作]

タイマをダイアログで指定した状態に初期設定した直後の動作を[動作開始]、[動作停止]から選択してください。

## [タイマ同期]

設定しません。

## [出力端子]

パルス出力に使用する端子を選択します。設定の有無はマイコン、選択したシリアル通信リソースにより異なります。[クロック端子選択] の設定内容を表4.4.27 に示します。

表4.4.27 [出力端子] 設定内容

マイコン	設定	内容
R8C/26,27	P3_0	パルス出力に P3_0 端子を使用します
	P1_3	パルス出力に P1_3 端子を使用します

## [外部トリガ極性]

[外部トリガ(カウント開始条件)]で外部トリガを選択した場合、トリガ有効エッジを選択してください。設定の有無、選択項目はマイコンにより異なります。

## [外部トリガ(カウント開始条件)]

カウント開始条件を選択してください。設定の有無、選択項目はマイコンにより異なります。

## [出力機能]

出力波形の設定です。設定内容はマイコンにより異なります。表 4.4.28 に出力機能の設定内容を示します。

表4.4.28 [出力機能]設定内容

シリーズ(グループ)	設定	内容
M16C/62p M16C/28,28B,29	パルス出力あり	パルスを出力します。 プライマリ期間は”H”出力です。 (TAiOUT 端子はパルス出力端子として使用)
	パルス出力なし	パルスを出力しません。 (TAiOUT 端子は入出力ポートとして使用)
R8C/13,22-29 R8C/2A-2D	プライマリ期間”H”出力、 セカンダリ期間”L”出力、 タイマ停止時”L”出力	プライマリ期間、セカンダリ期間、タイマ停止時の出力を設定します。
	プライマリ期間”L”出力、 セカンダリ期間”H”出力、 タイマ停止時”H”出力	
H8/3687,36077 H8/36049,36109	プライマリ期間”H”出力、 セカンダリ期間”L”出力	プライマリ期間、セカンダリ期間の出力を設定します。
	プライマリ期間”L”出力、 セカンダリ期間”H”出力	

## [出力波形の幅・周期]

生成される PWM 波形を設定します。

## ・ [周期]

出力波形の周期を入力してください。

## ・ [周波数]

出力波形の周波数が表示されます。

## ・ [デューティ]

出力波形のデューティを入力してください。設定するデューティは周期に対するプライマリ期間の割合です。

## ・ [計算結果]

[周期]、[デューティ]で入力した値とカウントソースの周波数より最適なレジスタ値を計算し、結果を表示します。

## [周期]

算出したレジスタ値を適用した場合の実際の周期が表示されます。

## [設定値]

入力した[周期]および[デューティ]の値を元に各レジスタの値を算出し、表示します。レジスタ値が設定可能範囲を外れている場合、後述の[設定]ボタンを押しても設定を適用することはできません。設定を元に生成したドライバソースを使用すると、表示されている値がレジスタに設定されます。設定するレジスタはマイコンにより異なります。

## [一括ソース生成をする]

チェックすると、[設定]ボタンクリック時にドライバソースを生成します。

## [設定]

設定した内容を保存し、ダイアログを閉じます。[一括ソース生成をする]にチェックされている場合はドライバソースを生成します。

## [キャンセル]

設定した内容を保存せずにダイアログを閉じます。

## 4.4.4 パルス周期測定モード

パルス周期測定モードの設定が可能な各マイコンのタイマリソースを表4.4.29 に示します。

表4.4.29 パルス周期測定モードに対応する各マイコンのタイマリソース

シリーズ	グループ	タイマリソース
N16C/60	M16C/62p	B0-5
M16C/Tiny	M16C/28,28B,29	B0-2
R8C/Tiny	R8C/13	X
	R8C/22-29, 2A-2D	RA
H8/300H Tiny	H8/3687, 36077	Z0,Z1
	36049	W,Z0,Z1
	36109	RC,RD0-3
SH/Tiny	SH7125	チャンネル 0-5

各マイコンでのパルス周期測定モード設定の概要は次の通りです。

- M16C/62P,M16C/28,28B,29, R8C/13,22-29,2A-2D

タイマ B のパルス周期測定モードにより外部信号のパルス周期を測定します。カウンタを 0 からアップカウントし、測定パルスの有効エッジでリロードレジスタにカウンタ値が転送されます。カウンタは 0 にクリアされ、カウントアップを継続します。転送されたりロードレジスタ値とカウントソースの周期より入力パルスの周期を算出できます。測定パルスの有効エッジ入力時とカウンタオーバフロー時の割り込みを検出できます。

- R8C/13,22-29,2A-2D

タイマ RA (R8C/13 はタイマ X) のパルス周期測定モードにより外部信号のパルス周期を測定します。リロードレジスタの内容をリロードしてダウンカウントを行い、有効エッジが入力されるとカウンタの値を読み出し用バッファに格納します。バッファの値とリロード値、カウントソースの周期より入力パルスの周期を算出できます。アンダフロー時、またはリロード時のタイマ RA 割り込み (R8C/13 はタイマ X 割り込み) を検出できます。

- H8/3687,36049,36077,36109

基本的な動作はタイマ Z、タイマ W、タイマ RC、タイマ RD で共通です。各タイマのインプットキャプチャにより外部信号のパルス周期を測定します。有効エッジ検出時のカウンタ値がそれぞれの端子に対応したジェネラルレジスタに転送されます。本設定ではカウンタクリア要因を指定できません。カウンタはクリアされずアップカウントを継続します。次の有効エッジで再びカウンタ値がジェネラルレジスタに転送されます。2 回の検出エッジで取得したカウンタ値の差分とカウントソースの周期より入力パルスの周期を算出できます。カウンタのオーバフロー割り込みを検出できます。

- SH7125

外部信号の入力端子に対応したジェネラルレジスタに、有効エッジ検出時のカウンタ値が転送されます。カウンタクリア要因をパルス入力端子に対応するジェネラルレジスタのインプットキャプチャとすることにより、有効エッジ検出時にカウンタがクリアされます。この場合、取得したジェネラルレジスタ値とカウントソースの周期より入力パルスの周期を算出できます。カウンタのオーバフロー割り込みと、測定する端子のインプットキャプチャ割り込みを検出できます。

パルス周期測定モードの設定ダイアログを図 4.4.7 に示します。

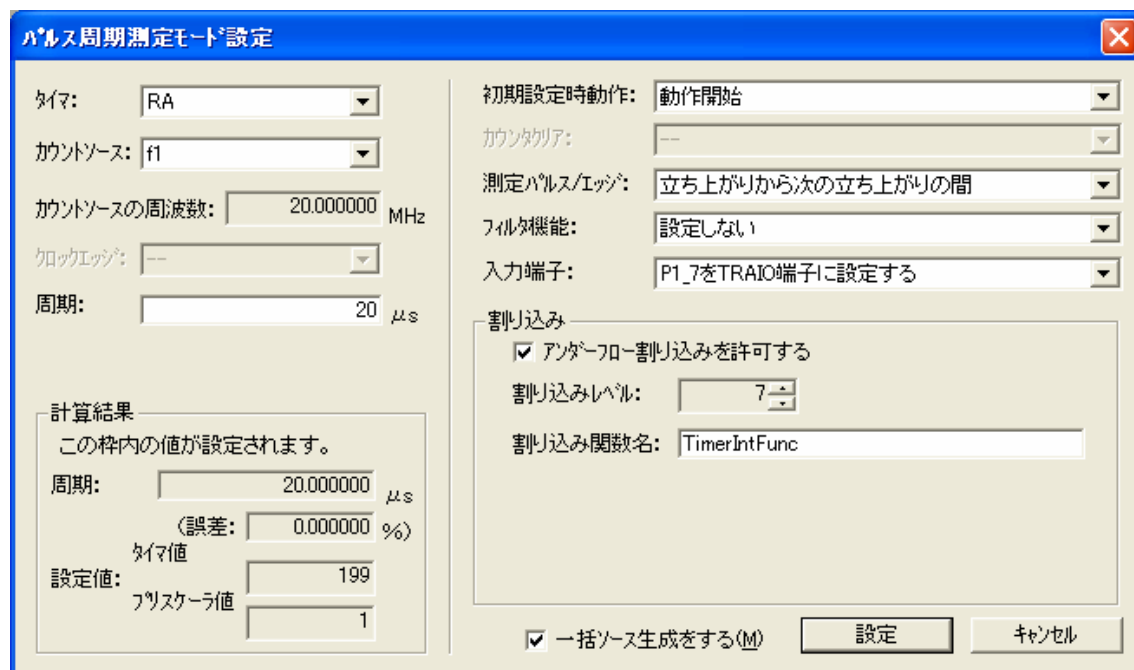


図4.4.7 パルス周期測定モード設定ダイアログ

#### [タイマ]

設定するタイマリソースを指定してください。[設定なし]を選択すると、リソースを指定しない設定を作成し、後からリソースを割り当てることができます。ただし SH7125 では[設定なし]を選択できません。

#### [カウントソース]

カウンタのカウントソースを選択してください。

#### [カウントソースの周波数] :

選択したカウントソースの周波数が表示されます。

#### [クロックエッジ]

カウントに使用するカウントソースクロックのエッジを選択してください。設定の有無はマイコンにより異なります。

#### [周期]

カウンタのアンダフロー周期を指定してください。設定の有無はマイコンにより異なります。周期の設定内容を表 4.4.30 に示します。

表4.4.30 [周期]設定内容

マイコン	内容
R8C/13,22-29,2A-2D	カウンタのアンダフロー周期を設定してください。指定した値とカウントソースの周波数から適切なプリスケアラとタイマの各レジスタ設定値が算出され、後述の[計算結果]に表示されます。アンダフロー時または有効エッジ検出時、算出された値がカウンタにリロードされてダウンカウントを続けます。

#### [計算結果]

[周期]で指定した値から適切な各レジスタ設定値を算出し表示します。表示の有無は CPU により異なります。

#### [周期]

指定した周期に対して算出したレジスタを適用した場合の実際の時間が表示されます。

## [誤差]

指定した周期に対する実際の時間の誤差を表示します。

## [設定値]

指定した周期に適した各レジスタの値を算出し、表示します。表示するレジスタはマイコンにより異なります。各レジスタの設定内容を表 4.4.31 に示します。

表4.4.31 周期レジスタ設定内容

マイコン	設定値	内容
R8C/13,22-29,2A-2D	プリスケアラ	カウントアップに用いるプリスケアラの値です。測定パルスはプリスケアラ周期の 2 倍より長くしてください。
	タイマ	カウンタのアンダフローおよび有効エッジ検出時にリロードされる値です。

## [初期設定動作]

タイマをダイアログで指定した状態に初期設定した直後の動作を[動作開始]、[動作停止]から選択してください。

## [カウンタクリア]

カウンタのクリア要因を選択します。設定の有無はマイコンにより異なります。カウンタクリアの設定内容を表 4.4.32 に示します。

表4.4.32 [カウンタクリア]設定内容

マイコン	設定	内容
SH7125	GRxのインプットキャプチャ (x:A~D,U,V,W)	選択したジェネラルレジスタインプットキャプチャをカウンタクリア要因とします。後述の[入力端子]で対応する端子を指定すると、測定するパルスの有効エッジ検出時にカウンタ値がジェネラルレジスタに退避された後 0 にクリアされます。

## [測定パルス]

パルス周期の測定区間を[立ち上がりエッジから次の立ち上がりエッジまで]または[立ち下がりエッジから次の立ち下がりエッジまで]から選択してください。

## [フィルタ機能]

フィルタ機能を使用する場合にフィルタのサンプリングを選択してください。設定の有無はマイコンにより異なります。

表4.4.33 [フィルタ機能]設定内容

マイコン	設定	内容
R8C/22-29	設定しない	フィルタ機能を使用しません
R8C/2A-2D	f1 でサンプリング	選択した信号で外部信号をサンプリングし、3 回連続で同じ値をサンプリングすると入力信号と認識します
	f8 でサンプリング	
	f32 でサンプリング	
H8/36109	f1 でサンプリング	選択した信号で外部信号をサンプリングし、3 回連続で同じ値をサンプリングすると入力信号と認識します
	f8 でサンプリング	
	f32 でサンプリング	
	カウントソースと同じ	

## [入力端子]

パルスを入力端子を選択します。設定の有無はマイコンにより異なります。入力端子の設定内容を表 4.4.34 に示します。

表4.4.34 [入力端子]設定内容

マイコン	設定	内容
R8C/22-29 R8C/2A-2D	P1_7 を TRAI0 端子に設定する	選択した端子のパルス周期を測定します。
	P1_5 を TRAI0 端子に設定する	
H8/3687 H8/36077 H8/36049 H8/36109	FTIOA(i)端子からの入力を測定する	FTIOA(i) (i:チャンネル) 端子のパルス周期を測定します。有効エッジ検出時のカウンタ値は GRA に転送されます
	FTIOB(i)端子からの入力を測定する	FTIOB(i) (i:チャンネル) 端子のパルス周期を測定します。有効エッジ検出時のカウンタ値は GRB に転送されます
SH7125	TIOCixA からの入力を測定する (GRx) (x: A~D,U,V,W)	選択した端子のパルス周期を測定します。有効エッジ検出時のカウンタ値は対応するジェネラルレジスタ(GRx)に転送されます

## [割り込み]

割り込みを許可し、割り込み発生の検出を行うために設定します。検出する割り込みをチェックし、割り込みレベルを設定すると、割り込み発生時にユーザ作成の割り込み関数が呼び出されます。各項目の設定方法を以下に示します。

## ・ [ &lt;割り込み種別&gt; 割り込みを許可する ]

各割り込み種別に対応した項目をチェックすると、該当の割り込みを検出することができます。マイコンにより検出可能な割り込み種別は異なります。各マイコンの割り込み種別を表 4.4.35 に示します。

表4.4.35 各マイコンの割り込み種別

マイコン	割り込み種別	内容
M16C/62P M16C/28,28B,29	オーバフロー/有効エッジ入力時割り込み	測定パルスの有効エッジ入力時とカウンタオーバフロー時に発行されます
R8C/13,22-29, R8C/2A-2D	アンダフロー割り込み	アンダフロー時、または有効エッジ入力時に発行されます
H8/3687,36077 H8/36049,36109	オーバフロー割り込み	カウンタオーバフロー時に発行されます
SH7125	オーバフロー割り込み	カウンタオーバフロー時に発行されます
	インプットキャプチャ割り込み	測定端子のインプットキャプチャ時(有効エッジ入力時)に発行されます

## ・ [ &lt;割り込み種別&gt; 割り込みレベル ]

チェックした割り込みの優先度を設定します。マイコンまたは割り込み種別によっては優先度を設定できません。

## ・ [割り込み通知関数名]

該当の割り込み発生時に呼び出される通知関数名を指定します。割り込み通知関数を使用する場合は指定した名前に関数を追加してください。割り込み関数の宣言は次のようになります。

## 関数宣言

```
void [設定した通知関数名](void);
```



## [一括ソース生成をする]

チェックすると、[設定]ボタンクリック時にドライバソースを生成します。

## [設定]

設定した内容を保存し、ダイアログを閉じます。[一括ソース生成をする]にチェックされている場合はドライバソースを生成します。

## [キャンセル]

設定した内容を保存せずにダイアログを閉じます。

#### 4.4.5 パルス幅測定モード

パルス幅測定モードの設定が可能な各マイコンのタイマリソースを表4.4.36 に示します。

表4.4.36 パルス周期測定モードに対応する各マイコンのタイマリソース

シリーズ	グループ	タイマリソース
N16C/60	M16C/62p	B0-5
M16C/Tiny	M16C/28,28B,29	B0-2
R8C/Tiny	R8C/13	X
	R8C/22-29, 2A-2D	RA
H8/300H Tiny	H8/3687, 36077	Z0,Z1
	36049	W,Z0,Z1
	36109	RC,RD0-3
SH/Tiny	SH7125	チャンネル 0-5

各マイコンでのパルス幅測定モード設定の概要は次の通りです。

- M16C/62P,M16C/28,28B,29, R8C/13,22-29,2A-2D

タイマ B のパルス幅測定モードにより外部信号のパルス幅を測定します。測定パルスの有効エッジでリロードレジスタにカウンタ値が転送され、カウンタは 0 クリアからカウントアップを続けます。転送されたリロードレジスタ値とカウントソースの周期より入力パルスの幅を算出できます。有効エッジは立ち上がりエッジと立ち下がりエッジの両方です。"H"幅、"L"幅を指定した測定はできません。"H"幅、"L"幅を測定するには入力端子のレベルを確認してください。測定パルスの有効エッジ入力時とカウンタオーバフロー時の割り込みを検出できます。

- R8C/13,22-29,2A-2D

タイマ RA (R8C/13 はタイマ X) のパルス幅測定モードにより外部信号のパルス幅を測定します。"H"期間または"L"期間を指定した測定が可能です。リロードレジスタの内容をリロードし、入力パルスが指定されたレベルにある間ダウンカウントを行います。入力パルスの立ち上がりまたは立ち下がりで測定期間が終了すると割り込みが発生し、ダウンカウントが停止します。取得したバッファの値とリロード値、カウントソースの周期より入力パルスの幅を算出できます。アンダフロー時、または測定期間終了時のタイマ RA 割り込み (R8C/13 はタイマ X 割り込み) を検出できます。

- H8/3687,36049,36077,36109

基本的な動作はタイマ Z、タイマ W、タイマ RC、タイマ RD で共通です。各タイマのインプットキャプチャにより外部信号のパルス幅を測定します。有効エッジ検出時のカウンタ値がそれぞれの端子に対応したジェネラルレジスタに転送されます。本設定ではカウンタクリア要因を指定できません。カウンタはクリアされずアップカウントを続けます。次の有効エッジで再びカウンタ値がジェネラルレジスタに転送されます。2回の検出エッジで取得したカウンタ値の差分とカウントソースの周期より入力パルス幅を算出できます。"H"幅、"L"幅を指定した測定はできません。"H"幅、"L"幅を測定するには入力端子のレベルを確認してください。カウンタのオーバフロー割り込みを検出できます。

- SH7125

外部信号の入力端子に対応したジェネラルレジスタに、有効エッジ検出時のカウンタ値が転送されます。カウンタクリア要因をパルス入力端子に対応するジェネラルレジスタのインプットキャプチャとすることにより、有効エッジ検出時にカウンタがクリアされます。この場合、取得したジェネラルレジスタ値とカウントソースの周期より入力パルスの幅を算出できます。有効エッジは立ち上がりエッジと立ち下がりエッジの両方です。本設定では"H"幅、"L"幅を指定した測定はできません。"H"幅、"L"幅を測定するには入力端子のレベルを確認してください。カウンタのオーバフロー割り込みと、測定する端子のインプットキャプチャ割り込みを検出できます。

パルス幅測定モードの設定ダイアログを図 4.4.8 に示します。

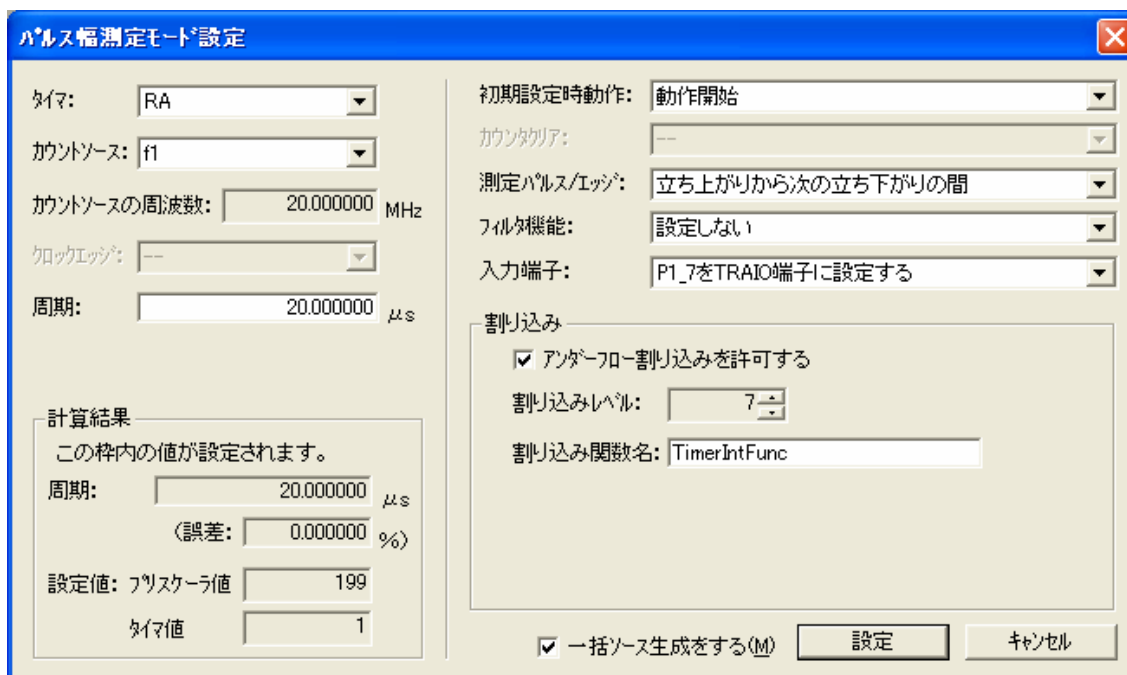


図4.4.8 パルス幅測定モード設定ダイアログ

パルス周期測定モードと異なる設定は以下です。それ以外の項目は「4.4.4 パルス周期測定モード」を参照してください。

#### [測定パルス]

パルス幅の測定区間を[立ち上がりエッジから次の立ち下がりエッジまで]または[立ち下がりエッジから次の立ち上がりエッジまで]から選択してください。設定の有無はマイコンにより異なります。測定パルスの設定内容を表 4.4.37 に示します。

表4.4.37 [測定パルス]設定内容

マイコン	設定	内容
R8C/13,22-29 R8C/2A-2D	立ち上がりから次の立ち下がりの間	“H”レベルの期間を測定します。“H”レベルの期間のみダウンカウントします。
	立ち下がりから次の立ち上がりの間	“L”レベルの期間を測定します。“L”レベルの期間のみダウンカウントします。

#### 4.4.6 インプットキャプチャモード

インプットキャプチャモードの設定が可能な各マイコンのタイマリソースを表4.4.38に示します。

表4.4.38 パルス周期測定モードに対応する各マイコンのタイマリソース

シリーズ	グループ	タイマリソース
N16C/60	M16C/62p	-
M16C/Tiny	M16C/28,28B,29	S
R8C/Tiny	R8C/13	C
	R8C/22-25	RD0-1
	R8C/26-29	RC
	R8C/2A-2D	RC,RD0-1,RF
H8/300H Tiny	H8/3687, 36077	Z0,Z1
	H8/36049	W,Z0,Z1
	H8/36109	RC,RD0-3
SH/Tiny	SH7125	チャンネル 0-5

各マイコンでのパルス幅測定モード設定の概要は次の通りです。

- M16C/28,28B,29

タイマ S の時間計測機能を使用し、外部信号変化の間隔を測定します。外部信号の有効エッジでベースタイマの値を G1TMj(j=0~7)レジスタに転送します。8チャンネルそれぞれに有効エッジ(立ち上がり/立ち下がり/両エッジ)、割り込み、フィルタ機能を設定できます。G1TMj(j=0~7)レジスタにより取得した時間計測開始時点および終了時点のベースタイマ値とカウントソースの周期より、時間を算出することができます。本設定では常にアップカウントモード、カウンタクリア無効となります。ベースタイマオーバフロー割り込みと、コンペアマッチ割り込み 0 および 1 を検出できます。

- R8C/13

タイマ C のインプットキャプチャモードで外部信号の間隔を測定します。外部信号の有効エッジでカウンタの値を TM0 レジスタに格納します。カウンタは 0 からオーバフローまでカウントします。カウンタクリアはできません。TM0 レジスタにより取得した時間計測開始時点および終了時点のベースタイマ値とカウントソースの周期より、時間を算出することができます。オーバフロー割り込みを検出することができます。

- R8C/22-29,2A-2D

各タイマのインプットキャプチャ機能により外部信号の間隔を測定します。有効エッジ検出時のカウンタ値がそれぞれの端子に対応したレジスタに転送されます。レジスタにより取得した時間計測開始時点および終了時点のベースタイマ値とカウントソースの周期より、時間を算出することができます。カウンタは 0 からオーバフローまでカウントしますが、R8C/22-25,2A-2D のタイマ RD はインプットキャプチャによるカウンタクリアを指定できます。カウンタのオーバフロー割り込みと、各端子のインプットキャプチャ割り込みを検出できます。

- H8/3687,36049,36077,36109

基本的な動作はタイマ Z、タイマ W、タイマ RC、タイマ RD で共通です。各タイマのインプットキャプチャにより外部信号の間隔を測定します。有効エッジ検出時のカウンタ値がそれぞれの端子に対応したジェネラルレジスタに転送されます。カウンタは 0 からオーバフローまでカウントしますが、H8/3687, 36077, 36049 のタイマ Z、H8/36109 のタイマ RD はインプットキャプチャによるカウンタクリアを指定できます。カウンタのオーバフロー割り込みと、各端子のインプットキャプチャ割り込みを検出できます。

・ SH7125

外部信号の入力端子に対応したジェネラルレジスタに、有効エッジ検出時のカウンタ値が転送されます。カウンタクリア要因をパルス入力端子に対応するジェネラルレジスタのインプットキャプチャとすることにより、有効エッジ検出時にカウンタがクリアされます。この場合、取得したジェネラルレジスタ値とカウントソースの周期より外部信号の間隔を算出できます。カウンタのオーバーフロー割り込みと、測定する端子のインプットキャプチャ割り込みを検出できます。

インプットキャプチャモードの設定ダイアログを図 4.4.9 に示します。

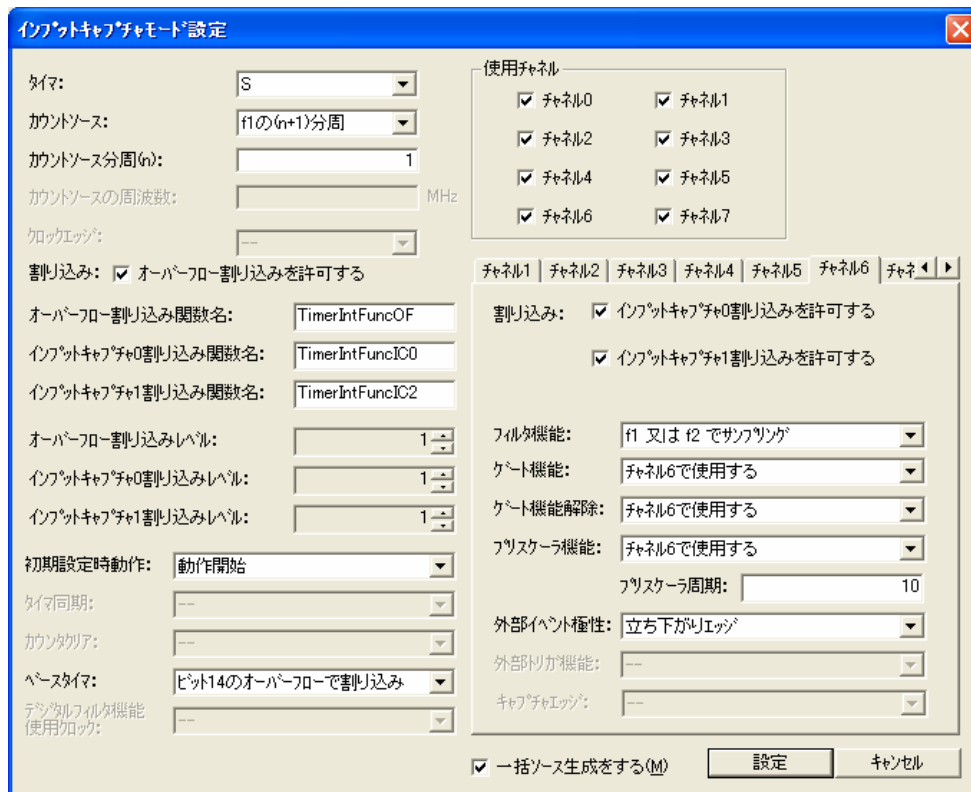


図4.4.9 インプットキャプチャモード設定ダイアログ

[タイマ]

設定するタイマリソースを指定してください。[設定なし]を選択すると、リソースを指定しない設定を作成し、後からリソースを割り当てることができます。ただし SH7125 では[設定なし]を選択できません。

[カウントソース]

カウンタのカウントソースを選択してください。

[カウントソース分周(n)]

カウントソース分周レジスタの値を設定します。設定の有無はマイコンにより異なります。カウントソース分周の設定内容を表 4.4.39 に示します。

表4.4.39 カウントソース分周設定内容

マイコン	内容
M16C/28,28B,29	設定値を $n$ とすると、カウントソースを $(n+1)$ 分周した周期でカウントします。設定可能な値は $0 \sim 255$ です。0 の場合は分周しません。

## [カウントソースの周波数]

選択したカウントソースの周波数が表示されます。表示の有無はマイコンにより異なります。16C/Tiny シリーズでは表示されません。

## [クロックエッジ]

カウントに使用するカウントソースクロックのエッジを選択してください。設定の有無はマイコンにより異なります。

表4.4.40 [クロックエッジ]設定内容

マイコン	設定	内容
SH7125	立ち上がりエッジ	カウントソースの立ち上がりエッジでカウントします
	立ち下がりエッジ※	カウントソースの立ち下がりエッジでカウントします
	両エッジ※	カウントソースの立ち上がり/立ち下がり両エッジでカウントします

カウントソースに MP  $\phi/1$  を指定した場合は選択できません

## [割り込み]

各種割り込み発生を検出を行うために設定します。検出する割り込みをチェックし、割り込みレベルを設定すると、割り込み発生時に[割り込み関数名]で指定した名前の関数が呼び出されます。[<割り込み種別>割り込みを許可する]をチェックすると、割り込み通知関数名の設定が可能となります。設定可能な割り込みの種別はマイコンにより異なります。各項目の設定方法を以下に示します。

## [オーバーフロー割り込みを許可する]

オーバーフロー割り込みを検出する場合にチェックします。インプットキャプチャ割り込みはチャンネルまたはジェネラルレジスタごとに設定できます。インプットキャプチャ割り込みを許可するには、後述のチャンネル、ジェネラルレジスタタブで設定してください。

## [&lt;割り込み種別&gt;割り込みレベル]

<割り込み種別>で示されている割り込みの優先度を設定します。割り込みレベル設定の有無はタイマにより異なります。

## [&lt;割り込み種別&gt;割り込み関数名]

<割り込み種別>で示されている割り込み発生時に呼び出される通知関数名を指定します。割り込み通知関数を使用する場合は指定した名前関数を追加してください。<割り込み種別>が示されていない場合、指定した関数は許可した全ての割り込み発生時に呼び出されます。関数の宣言は次のようになります。

関数宣言
void [設定した通知関数名](void);

## [初期設定時動作]

タイマをダイアログで指定した状態に初期設定した直後の動作を[動作開始]、[動作停止]から選択してください。選択できる項目はマイコンとタイマリソースにより異なります。

## [タイマ同期]

チャンネル間の同期動作を設定します。設定の有無はマイコンにより異なります。タイマ同期の設定内容を表 4.80 に示します。

表4.4.41 タイマ同期設定内容

マイコン	設定	内容
R8C/22-25 R8C/2A-2D H8/3687, 36077 H8/36049, 36109	チャンネル 0 とチャンネル 1 でタイマ同期しない	チャンネル 0 とチャンネル 1 のカウンタは独立して動作します。
	チャンネル 0 とチャンネル 1 でタイマ同期する	チャンネル 0 とチャンネル 1 のカウンタが同期プリセットされます。片方のチャンネル(RD0 または RD1)のカウンタに値を書き込むと、他のチャンネルのカウンタにも値が書き込まれます。2つのチャンネルを同期動作させる場合、チャンネル 0 とチャンネル 1 の両方の設定で[チャル 0 とチャンネル 1 でタイマ同期する]を設定してください。 同期クリアする場合は[チャンネル 0 とチャンネル 1 でタイマ同期する]を選択した後、[カウンタクリア]で[同期クリア]を選択してください。
SH7125	同期しない	他のチャンネルとは独立して動作します。[カウンタクリア]で設定するカウンタクリア要因には、チャンネル内の TGR が指定されます。
	タイマ同期の主となる	他のチャンネルと同期して動作します。[カウンタクリア]で設定するカウンタクリア要因は、同期する他のチャンネルのカウンタクリア要因となります。
	同期する	他のチャンネルで[タイマ同期の主となる]が指定されている場合のみ選択可能です。カウンタクリア要因は[タイマ同期の主となる]が指定されているチャンネルの設定に従います。[カウンタクリア]で設定するカウンタクリア要因は同期クリアのみ選択できます。

## [カウンタクリア]

カウンタのクリア要因を選択します。設定の有無 CPU により異なります。カウンタクリアの設定内容を表 4.4.42 に示します。

表4.4.42 カウンタクリア設定内容

マイコン	設定	内容
R8C/22-25 R8C/2A-2D H8/3687, 36077 H8/36049, 36109	カウンタクリアなし	カウンタクリアしません。
	<ジェネラルレジスタ名>のイ ンพุットキャプチャ	指定したジェネラルレジスタのインพุットキャプチャでクリアします。
	同期クリア	別のチャンネルのカウンタクリアと同時にクリアされます。別のチャンネルの設定でクリア要因を指定してください。
SH7125	<ジェネラルレジスタ名>のイ ンพุットキャプチャ	指定したジェネラルレジスタのインพุットキャプチャでクリアします。

## [ベースタイマ]

ベースタイマオーバフロー割り込みのタイミングを指定してください。設定の有無はマイコンにより異なります。ベースタイマの設定内容を表 4.4.43 に示します。



表4.4.43 ベースタイマ設定内容

マイコン	設定	内容
M16C/28,28B,29	ビット14のオーバーフローで割り込み	ベースタイマレジスタのビット14のオーバーフローでオーバーフロー割り込みが発生します
	ビット15のオーバーフローで割り込み	ベースタイマレジスタのビット15のオーバーフローでオーバーフロー割り込みが発生します

## [デジタルフィルタ機能使用クロック]

デジタルフィルタ機能に使用するクロックを選択してください。設定の有無はマイコンにより異なります。デジタルフィルタ機能を使用する場合は、後述の各チャンネルまたはジェネラルレジスタ設定タブの[フィルタ機能]で[デジタルフィルタ機能あり]を指定してください。入力端子ごとにデジタルフィルタ機能使用クロックを選択できるマイコンでは本項目は無効です。各チャンネルまたはジェネラルレジスタ設定タブの[フィルタ機能]でクロックを選択してください。

## [使用チャンネル]

使用可能なチャンネルまたはジェネラルレジスタが表示されます。使用するチャンネルまたはジェネラルレジスタをチェックしてください。複数選択することが可能です。チェックするとそのチャンネルまたはジェネラルレジスタに対応したタブが開き、それぞれの詳細設定が可能となります。

## チャンネル、ジェネラルレジスタタブ

[使用チャンネル]で選択したチャンネルまたはジェネラルレジスタを設定してください。

## [インプットキャプチャ割り込みを許可する]

チャンネルまたはジェネラルレジスタのインプットキャプチャ割り込みをを許可し、割り込み発生を検出を行うために設定します。チェックすると、登録した通知関数が割り込み発生時に呼び出されます。マイコンによっては通知関数と割り込みレベルの設定が、オーバーフロー割り込みと共通です。前述の[<割り込み種別>割り込みレベル]、[<割り込み種別>割り込み関数名]を設定してください。

## [フィルタ機能]

フィルタ機能を使用する場合に設定します。設定の有無はマイコンにより異なります。[デジタルフィルタ機能使用クロック]で選択したクロックでフィルタ機能を使用する場合は、[デジタルフィルタ機能あり]を設定してください。チャンネルごとに使用クロックを設定できるマイコンでは個別に使用クロックを選択してください。

## [ゲート機能]

ゲート機能を使用する場合に設定してください。設定の有無はマイコンにより異なります。ゲート機能の設定内容を表 4.4.44 に示します。

表 4.4.44 ゲート機能設定内容

マイコン	設定	内容
M16C/28,28B,29 ※	使用しない	ゲート機能を使用しません。
	チャンネル i で使用する	ゲート機能を使用します。最初のトリガ入力による時間計測後、トリガ入力の受け付けを禁止します。ゲート機能の解除については、[ゲート機能解除]を設定してください。

タイマ S チャンネル 6,7 でのみ設定可能です

## [ゲート機能解除]

ゲート機能を使用する場合に設定してください。設定の有無はマイコンにより異なります。ゲート機能解除の設定内容を表 4.4.45 に示します。

表4.4.45 ゲート機能解除設定内容

マイコン	設定	内容
M16C/28,28B,29 ※	使用しない	[ゲート機能]で[使用しない]に設定した場合に選択してください。
	チャンネル i で使用する	ゲート機能を使用する場合、トリガを再び受け付けるために設定します。設定すると、G1Pop レジスタ(チャンネル 6 の時 p=4、チャンネル 7 の時 p=5)とベースタイマー一致で再度トリガ入力を受け付けられます。ゲート機能解除を使用する場合、[ゲート機能]は[チャンネル i で使用する]を選択してください。

タイマ S チャンネル 6,7 でのみ設定可能です

## [プリスケータ機能]

プリスケータ機能を使用する場合に設定してください。設定の有無はマイコンにより異なります。プリスケータ機能の設定内容を表 4.4.46 に示します。

表 4.4.46 プリスケータ機能設定内容

マイコン	設定	内容
M16C/28,28B,29 ※	使用しない	プリスケータ機能を使用しません。トリガ入力ごとに時間が計測されます。
	チャンネル i で使用する	プリスケータ機能を使用します。時間計測プリスケータレジスタ値+1 回のトリガ入力ごとに時間が計測されます。プリスケータレジスタの値は[プリスケータ周期]で設定してください。

タイマ S チャンネル 6,7 でのみ設定可能です

## [プリスケータ周期]

プリスケータ機能を使用する場合のプリスケータ周期を設定してください。設定の有無はマイコンにより異なります。プリスケータ周期の設定内容を表 4.4.47 に示します。

表4.4.47 プリスケータ周期設定内容

マイコン	内容
M16C/28,28B,29	プリスケータレジスタ値を設定してください。プリスケータ機能を使用すると、時間計測プリスケータレジスタ値 +1 回のトリガ入力ごとに時間が計測されます。

## [外部イベント極性]

計測に使用するトリガのエッジを立ち上がりエッジ、立ち下がりエッジおよび両エッジから選択してください。

## [外部トリガ機能]

入力パルスを選択できる場合に有効になります。キャプチャするパルスを選択してください。設定の有無、選択項目はマイコン、チャンネルにより異なります。

## [一括ソース生成をする]

チェックすると、[設定]ボタンクリック時にドライバソースを生成します。

## [設定]

設定した内容を保存し、ダイアログを閉じます。[一括ソース生成をする]にチェックされている場合はドライバソースを生成します。

## [キャンセル]

設定した内容を保存せずにダイアログを閉じます。

#### 4.4.7 アウトプットコンペアモード

アウトプットコンペアモードの設定が可能な各マイコンのタイマリソースを表 4.4.48 に示します。

表4.4.48 アウトプットコンペアモードに対応する各マイコンのタイマリソース

シリーズ	グループ	タイマリソース
N16C/60	M16C/62p	-
M16C/Tiny	M16C/28,28B,29	S
R8C/Tiny	R8C/13	C
	R8C/22-25	RD0-1,RE
	R8C/26-29	RC,RE
	R8C/2A-2D	RC,RD0-1,RE
H8/300H Tiny	H8/3687, 36077	Z0,Z1
	H8/36049	W,Z0,Z1
	H8/36109	RC,RD0-3
SH/Tiny	SH7125	チャンネル 0-4

各マイコンでのパルス幅測定モード設定の概要は次の通りです。

- M16C/28,28B,29,

タイマ S の波形生成機能により、最大 8 チャンネルの波形を生成します。単相波形出力モード、反転波形出力モード、セット/リセット波形出力(SR 波形出力)モードの設定が可能です。0~7 の各チャンネルの波形生成レジスタの値がアップカウントモードで動作するベースタイマ値と一致した際に、各チャンネルの出力レベルを制御します。オーバフロー割り込みと、コンペアマッチ割り込み 0 および 1 を検出できます。

- R8C/13

タイマ C のアウトプットコンペアモードで波形を生成します。コンペア 0 レジスタまたはコンペア 0 レジスタの値がアップカウントするカウンタ値と一致した際に、出力レベルを制御します。オーバフロー割り込みと各コンペアレジスタのコンペアマッチ割り込みを検出できます。

- R8C/22-29,2A-2D

各タイマのアウトプットコンペア機能により波形を生成します。ジェネラルレジスタまたはコンペアレジスタの値とアップカウントするカウンタ値が一致した際に、各レジスタに対応する端子の出力レベルを制御します。タイマ RC および RD はコンペアマッチによるカウンタクリアを設定可能です。タイマ RE はコンペアレジスタとカウンタとの一致で常にカウンタが 0 にクリアされます。オーバフロー割り込みおよび各レジスタのコンペアマッチ割り込みを検出できます。

- H8/3687,36049,36077,36109

各タイマのジェネラルレジスタ値とアップカウントするカウンタ値が一致した際に、ジェネラルレジスタに対応する端子の出力レベルを制御します。コンペアマッチによるカウンタクリアを設定可能です。オーバフロー割り込みおよび各ジェネラルレジスタのコンペアマッチ割り込みを検出できます。

- SH7125

各チャンネルのジェネラルレジスタ値とアップカウントするカウンタ値が一致した際に、ジェネラルレジスタに対応する端子の出力レベルを制御します。コンペアマッチによるカウンタクリアを設定可能です。オーバフロー割り込みおよび各ジェネラルレジスタのコンペアマッチ割り込みを検出できます。

アウトプットコンペアモードの設定ダイアログを図 4.4.10 に示します。

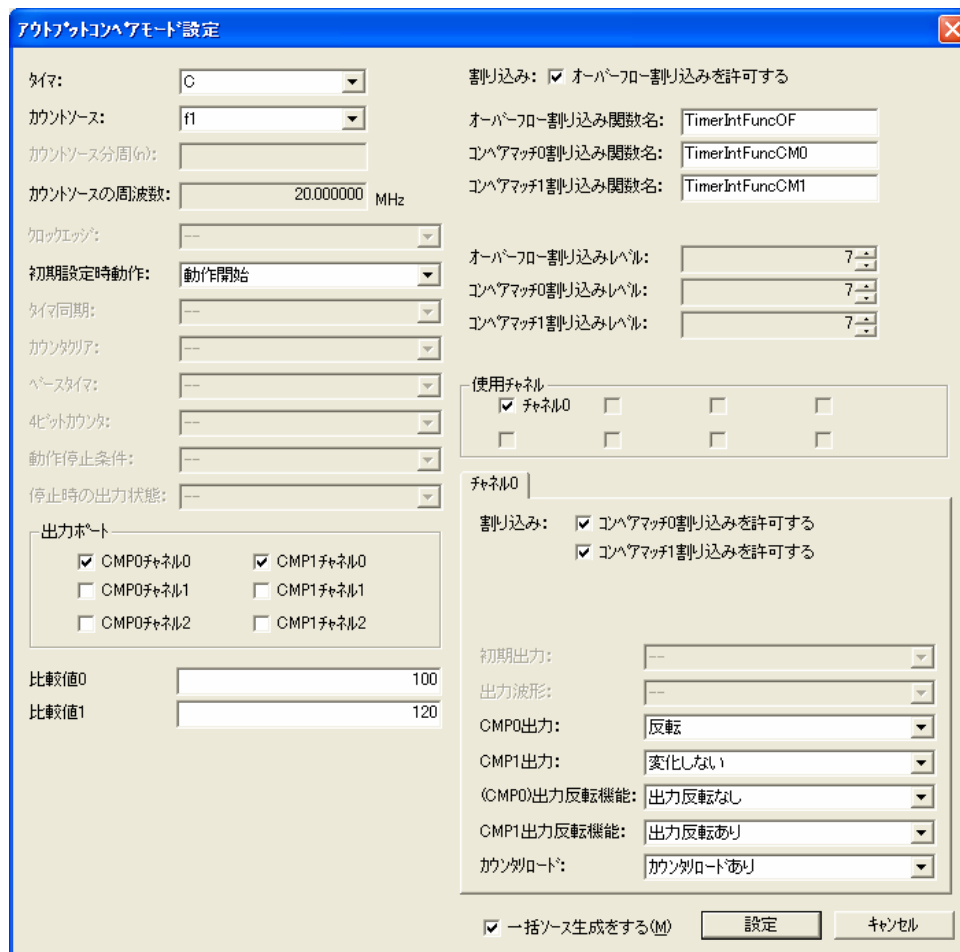


図4.4.10 アウトプットモード設定ダイアログ

#### [タイマ]

設定するタイマリソースを指定してください。[設定なし]を選択すると、リソースを指定しない設定を作成し、後からリソースを割り当てることができます。ただし SH7125 では[設定なし]を選択できません。

#### [カウントソース]

カウンタのカウントソースを選択してください。

#### [カウントソース分周(n)]

カウントソース分周レジスタの値を設定します。設定の有無は CPU により異なります。カウントソース分周の設定内容を表 4.4.49 に示します。

表4.4.49 カウントソース分周(n)設定内容

マイコン	内容
M16C/28,28B,29	設定値を $n$ とすると、カウントソースを $(n+1)$ 分周した周期でカウントします。設定可能な値は 0~255 です。0 の場合は分周しません。

#### [カウントソースの周波数]

選択したカウントソースの周波数が表示されます。表示の有無はマイコンにより異なります。

## [初期設定動作]

タイマをダイアログで指定した状態に初期設定した直後の動作を[動作開始]、[動作停止]から選択してください。選択できる項目はマイコンとタイマリソースにより異なります。

## [クロックエッジ]

カウントに使用するカウントソースクロックのエッジを選択してください。設定の有無はマイコンにより異なります。

## [タイマ同期]

チャンネル間の同期動作を設定します。設定の有無はマイコンにより異なります。タイマ同期の設定内容を表 4.4.50 に示します。

表4.4.50 タイマ同期設定内容

マイコン	設定	内容
R8C/22-25 R8C/2A-2D H8/3687, 36077 H8/36049, 36109	チャンネル 0 とチャンネル 1 でタイマ同期しない	チャンネル 0 とチャンネル 1 のカウンタは独立して動作します。
	チャンネル 0 とチャンネル 1 でタイマ同期する	チャンネル 0 とチャンネル 1 のカウンタが同期プリセットされます。片方のチャンネル(RD0 または RD1)のカウンタに値を書き込むと、他のチャンネルのカウンタにも値が書き込まれます。2 つのチャンネルを同期動作させる場合、チャンネル 0 とチャンネル 1 の両方の設定で[チャル 0 とチャンネル 1 でタイマ同期する]を設定してください。 同期クリアする場合は[チャンネル 0 とチャンネル 1 でタイマ同期する]を選択した後、[カウンタクリア]で[同期クリア]を選択してください。
SH7125	同期しない	他のチャンネルとは独立して動作します。[カウンタクリア]で設定するカウンタクリア要因には、チャンネル内の TGR が指定されます。
	タイマ同期の主となる	他のチャンネルと同期して動作します。[カウンタクリア]で設定するカウンタクリア要因は、同期する他のチャンネルのカウンタクリア要因となります。
	同期する	他のチャンネルで[タイマ同期の主となる]が指定されている場合のみ選択可能です。カウンタクリア要因は[タイマ同期の主となる]が指定されているチャンネルの設定に従います。[カウンタクリア]で設定するカウンタクリア要因は同期クリアのみ選択できます。

## [カウンタクリア要因]

カウンタのクリア要因を選択します。設定の有無 CPU により異なります。カウンタクリアの設定内容を表 4.4.51 に示します。

表4.4.51 カウンタクリア要因設定内容

マイコン	設定	内容
R8C/22-25 R8C/2A-2D H8/3687, 36077 H8/36049, 36109 SH7125	カウンタクリアなし	カウンタクリアしません。
	<ジェネラルレジスタ名>の コンペアマッチ	指定したジェネラルレジスタのコンペアマッチでクリアします。
	同期クリア	設定するチャンネルと別のチャンネルのカウンタクリアと同時にクリアされます。別のチャンネルの設定でクリア要因を指定してください。
SH7125	<ジェネラルレジスタ名>の インプットキャプチャ	指定したジェネラルレジスタのインプットキャプチャでクリアします。

## [ベースタイマ]

ベースタイマオーバフロー割り込みのタイミングを指定してください。設定の有無はマイコンにより異なります。ベースタイマの設定内容を表 4.4.52 に示します。

表4.4.52 ベースタイマ設定内容

マイコン	設定	内容
M16C/28,28B,29	ビット 14 のオーバフロー で割り込み	ベースタイマレジスタのビット 14 のオーバフローでオーバフロー割り込みが発生します
	ビット 15 のオーバフロー で割り込み	ベースタイマレジスタのビット 15 のオーバフローでオーバフロー割り込みが発生します

## [4 ビットカウンタ]

4 ビットカウンタの使用を選択します。設定の有無はマイコンにより異なります。4 ビットカウンタの設定内容を表 4.4.53 に示します。

表4.4.53 4ビットカウンタ設定内容

マイコン	設定	内容
R8C/22-29※ R8C/2A-2D※	使用する	カウントソースを 2 分周したものを 4 ビットカウンタと 8 ビットカウンタでカウントします。比較値(TREMIN レジスタ値)と一致するまでの周期は次のようになります。 $1/f_i \times 32 \times (n+1)$ fi:カウントソース周波数 n:比較値(TREMIN レジスタ値)
	使用しない	カウントソースを 2 分周したものを 8 ビットカウンタでカウントします。比較値(TREMIN レジスタ値)と一致するまでの周期は次のようになります。 $1/f_i \times 2 \times (n+1)$ fi:カウントソース周波数 n:比較値(TREMIN レジスタ値)

タイマ RE 選択時のみ設定可能です



## [動作停止条件]

動作停止条件を選択します。設定の有無 CPU により異なります。動作停止条件の設定内容を表 4.4.54 に示します。

表4.4.54 動作停止条件設定内容

マイコン	設定	内容
R8C/22-25 ※ R8C/2A-2D ※	ソフトによる停止 (TSTART ビット)	タイマ RD スタートレジスタ(TRDSTR)の TRDi カウント開始フラグ(TSTARTi ビット)(i=0,1)への“0”(カウント停止)書き込みで動作停止します。アウトプットコンペア出力端子はカウント停止前の出力レベルを保持します。
	GRA のコンペアマッチで停止	タイマ RD ジェネラルレジスタ Ai (TRDGRAi) (i=0,1)とのコンペアマッチでカウント停止します。アウトプットコンペア出力端子はコンペアマッチによる出力変化後のレベルを保持します。

タイマ RD 選択時のみ設定可能です

## [停止時の出力状態]

停止時の出力状態を選択します。設定の有無 CPU により異なります。停止時の出力状態の設定内容を表 4.4.55 に示します。

表4.4.55 停止時の出力状態設定内容

マイコン	設定	内容
R8C/2A-2D※	L 出力	カウント停止時の出力レベルを”L”にします。
	H 出力	カウント停止時の出力レベルを”H”にします。
	出力保持	カウント停止時、出力レベルを保持します。

タイマ RF 選択時のみ設定可能です

## [出力ポート]

出力端子を選択できる場合に有効です。設定の有無 CPU により異なります。出力ポートの設定内容を表 4.4.56 に示します。

表4.4.56 出力ポート設定内容

マイコン	設定	内容
R8C/13 ※1	CMP0 チャンネル 0	CMP00 出力許可する場合にチェックしてください
	CMP0 チャンネル 1	CMP01 出力許可する場合にチェックしてください
	CMP0 チャンネル 2	CMP02 出力許可する場合にチェックしてください
	CMP1 チャンネル 0	CMP10 出力許可する場合にチェックしてください
	CMP1 チャンネル 1	CMP11 出力許可する場合にチェックしてください
	CMP1 チャンネル 2	CMP12 出力許可する場合にチェックしてください
R8C/2A-2D※2	CMP0 チャンネル 0	TRFO00 出力許可する場合にチェックしてください
	CMP0 チャンネル 1	TRFO01 出力許可する場合にチェックしてください
	CMP0 チャンネル 2	TRFO02 出力許可する場合にチェックしてください
	CMP1 チャンネル 0	TRFO10 出力許可する場合にチェックしてください
	CMP1 チャンネル 1	TRFO11 出力許可する場合にチェックしてください
	CMP1 チャンネル 2	TRFO12 出力許可する場合にチェックしてください

※1 タイマ C 選択時のみ設定可能です

※2 タイマ RF 選択時のみ設定可能です

## [比較値]

カウンタ値とのコンペアマッチを発生させるための比較値を設定してください。設定項目はマイコンにより異なります。比較値の設定内容を表 4.4.57 に示します。

表4.4.57 比較値設定内容

マイコン	比較値設定項目	内容
M16C/28,28B,29 タイマS	比較値 0-7	0~7チャンネルの波形生成レジスタ(G1POj j=0~7)の値に対応します。ベースタイマ値と一致すると、各チャンネルに対応する端子からの波形出力と、コンペアマッチ割り込み0および1の発生が可能です。出力波形および割り込みについては各チャンネルの設定で指定してください。
R8C/Tiny (R8C/13) タイマC	比較値 0-1	コンペア0レジスタおよびコンペア1レジスタに対応します。それぞれカウンタと一致するとコンペア i 割り込み(i=0,1)およびCMPi(i=0,1)からの波形出力が可能です。出力波形および割り込みについては各チャンネルの設定で指定してください。
R8C/26-29※ R8C/2A-2D※ タイマRC R8C/22-25※ R8C/2A-2D※ ※タイマRD	比較値 A-D	ジェネラルレジスタA~Dに対応します。カウンタ値と一致すると、各ジェネラルレジスタに対応する端子への波形出力と、コンペアマッチ割り込みの発生が可能です。出力波形および割り込みについては各チャンネルの設定で指定してください。
R8C/22-25※ R8C/2A-2D※ ※タイマRE	比較値	TREMIN レジスタ値に対応します。8ビットカウンタ値と一致すると、TREO の出力極性反転と、コンペア一致割り込みの発生が可能です。TREO 出力および割り込みについては各チャンネルの設定で指定してください。
R8C/2A-2D※ ※タイマRF	比較値 0-1	コンペア0レジスタおよびコンペア1レジスタに対応します。それぞれカウンタと一致するとコンペア i 割り込み(i=0,1)およびTRFO00~TRFO02, TRFO10~TRFO12 からの波形出力が可能です。出力波形および割り込みについては各チャンネルの設定で指定してください。
H8/3687, 36077 H8/36049, 36109 SH7125	比較値 A-D	ジェネラルレジスタA~Dに対応します。カウンタ値と一致すると、各ジェネラルレジスタに対応する端子への波形出力と、コンペアマッチ割り込みの発生が可能です。出力波形および割り込みについては各チャンネルの設定で指定してください。

**[割り込み]**

各種割り込み発生の検出を行うために設定します。検出する割り込みをチェックし、割り込みレベルを設定すると、割り込み発生時に[割り込み関数名]で指定した名前前の関数が呼び出されます。[<割り込み種別>割り込みを許可する]をチェックすると、割り込み通知関数名の設定が可能となります。設定可能な割り込みの種別はマイコンにより異なります。各項目の設定方法を以下に示します。

**[オーバーフロー割り込みを許可する]**

オーバーフロー割り込みを検出する場合にチェックします。コンペアマッチ割り込みはチャンネルまたはジェネラルレジスタごとに設定できます。コンペアマッチ割り込みを許可するには、後述のチャンネル、ジェネラルレジスタタブで設定してください。

**[<割り込み種別>割り込みレベル]**

<割り込み種別>で示されている割り込みの優先度を設定します。割り込みレベル設定の有無はタイマにより異なります。

**[<割り込み種別>割り込み関数名]**

<割り込み種別>で示されている割り込み発生時に呼び出される通知関数名を指定します。割り込み通知関数を使用する場合は指定した名前前で関数を追加してください。<割り込み種別>が示されていない場合、指定した関数は許可した全ての割り込み発生時に呼び出されます。関数の宣言は次のようになります。

関数宣言
void [設定した通知関数名](void);

**[使用チャンネル]**

使用可能なチャンネルまたはジェネラルレジスタが表示されます。使用するチャンネルまたはジェネラルレジスタをチェックしてください。複数選択することが可能です。チェックするとそのチャンネルまたはジェネラルレジスタに対応したタブが開き、それぞれの詳細設定が可能となります。

**チャンネル、ジェネラルレジスタタブ**

[使用チャンネル]で選択したチャンネルまたはジェネラルレジスタを設定してください。

**[コンペアマッチ割り込みを許可する]**

チャンネルまたはジェネラルレジスタでコンペアマッチ割り込みをを許可し、割り込み発生の検出を行うために設定します。チェックすると、割り込み発生時に登録した通知関数が呼び出されます。複数のチャンネルまたはジェネラルレジスタで割り込みを許可しても、割り込みレベルと通知関数は共通です。マイコンによっては通知関数と割り込みレベルの設定が、オーバーフロー割り込みと共通です。前述の[割り込み]から[割り込みレベル]と[割り込み関数]を指定してください。

**[初期出力]**

初期出力レベルの設定です。初期出力を 0 出力、1 出力から選択してください。設定の有無はマイコン、タイマリソースにより異なります。

**[出力波形]**

出力波形の設定です。選択の有無、選択可能項目はマイコン、タイマリソースにより異なります。出力波形の設定内容を表 4.4.58 に示します。

表4.4.58 出力波形設定内容

マイコン	比較値設定項目	内容
M16C/28,28B,29 ※ ※タイマS	単層波形	出力反転しない場合、ベースタイマと G1POj レジスタ(j=0~7)値が一致すると“H”レベルを出力します。ベースタイマが“000016”になると“L”レベルを出力します。
	反転波形	ベースタイマの値と一致するごとに出力レベルを反転します。
	SR 波形	出力反転しない場合、ベースタイマと G1POj レジスタ(j=0~7)値が一致すると、OUTC1j 端子から“H”レベルを出力します。ベースタイマの値が G1POk( k=j+1)レジスタの値と一致した場合、OUTC1j 端子から“L”レベルを出力します。
R8C/26-29 ※ R8C/2A-2D ※ ※タイマRC R8C/22-25 ※ R8C/2A-2D ※ ※タイマRD H8/3687, 36077 H8/36049, 36109	0 出力	コンペアマッチ時、0 を出力します
	1 出力	コンペアマッチ時、1 を出力します。
	トグル出力	コンペアマッチ時、トグル出力します。
R8C/22-25 ※ R8C/2A-2D ※ ※タイマRE	出力禁止	TREO 端子からのクロック出力を禁止します。
	F2 出力	TREO 端子から f2 を出力します。
	F4 出力	TREO 端子から f4 を出力します。
	F8 出力	TREO 端子から f8 を出力します。
	コンペア出力	TREO 端子をコンペア出力に設定します。コンペアマッチするごとに、出力極性が反転します。
R8C/Tiny R8C/13 ※ ※タイマC R8C/2A-2D ※ ※タイマRF	設定不可	出力波形は後述の[CMP0 出力]および[CMP1 出力]にて設定してください。
SH7125	0 出力	コンペアマッチ時、0 を出力します
	1 出力	コンペアマッチ時、1 を出力します。
	トグル出力	コンペアマッチ時、トグル出力します。

## [CMP0 出力]、[CMP1 出力]

コンペアマッチ発生時の出力を設定します。設定の有無はマイコン、タイマリソースにより異なります。CMP0 出力および CMP1 出力の内容を表 4.4.59 に示します。

表4.4.59 CMP0出力、CMP1出力設定内容

マイコン	項目	内容
R8C/13 ※ ※タイマC R8C/2A-2D ※ ※タイマRF	変化しない	コンペア i(i=0,1)一致時に出力を変化させません。
	反転	コンペア i(i=0,1)一致時に出力を反転させます。
	“L”に設定	コンペア i(i=0,1)一致時に出力を“L”にします。
	“H”に設定	コンペア i(i=0,1)一致時に出力を“H”にします。

## [(CMP0)出力反転機能]

出力反転機能の設定です。機能の有無はマイコン、タイマリソースにより異なります。(CMP0)出力反転機能の設定内容を表 4.4.60 に示します。

表4.4.60 (CMP0)出力反転機能設定内容

マイコン	項目	内容
M16C/28,28B,29 タイマ S	出力反転あり	出力反転します。
	出力反転なし	出力を反転しません。
R8C/13 タイマC	出力反転あり	CMP00～CMP02 からの CMP 出力を反転します。
	出力反転なし	CMP00～CMP02 からの CMP 出力を反転しません。
R8C/2A-2D タイマRF	出力反転あり	TRFO00～TRFO02 からの CMP 出力を反転します。
	出力反転なし	TRFO00～TRFO02 からの CMP 出力を反転しません。

## [(CMP1 出力反転機能)]

出力反転機能の設定です。機能の有無は CPU、タイマにより異なります。CMP1 出力反転機能の設定内容を表 4.4.61 に示します。

表4.4.61 CMP1出力反転機能設定内容

マイコン	項目	内容
R8C/13 タイマC	出力反転あり	CMP10～CMP12 からの CMP 出力を反転します。
	出力反転なし	CMP10～CMP12 からの CMP 出力を反転しません。
R8C/2A-2D タイマRF	出力反転あり	TRFO10～TRFO12 からの CMP 出力を反転します。
	出力反転なし	TRFO10～TRFO12 からの CMP 出力を反転しません。

## [カウンタリロード]

カウンタリロードの設定です。機能の有無は CPU、タイマにより異なります。カウンタリロードの設定内容を表 4.4.62 に示します。

表4.4.62 カウンタリロード設定内容

マイコン	項目	内容
M16C/28,28B,29 タイマS	書き込み時リロード	各チャネルの波形生成レジスタ(G1POj j=0～7)に値を書き込んだ直後に値が G1POj レジスタにリロードされ出力波形等に反映されます。
	ベースタイマセット時にリロード	各チャネルの波形生成レジスタ(G1POj j=0～7)に値を書き込むと、ベースタイマリセット時に値がリロードされます。
R8C/13	カウンタリロードなし	コンペア 1 一致時にカウンタの値をリセットしません。
	カウンタリロードあり	コンペア 1 一致時にカウンタの値を 0 にします。

## [一括ソース生成をする]

チェックすると、[設定]ボタンクリック時にドライバソースを生成します。

## [設定]

設定した内容を保存し、ダイアログを閉じます。[一括ソース生成をする]にチェックされている場合はドライバソースを生成します。

## [キャンセル]

設定した内容を保存せずにダイアログを閉じます。

## 4.5 割り込みの設定

各マイコンで設定可能な割り込み種別を表 4.5.1 に示します。

表4.5.1 各マイコンの設定可能割り込み種別

シリーズ	グループ	割り込み
N16C/60 M16C/Tiny	M16C/62p M16C/28,28B,29	INT0~INT5 割り込み キー入力割り込み (KI0~KI3)
R8C/Tiny	R8C/13 R8C/22-25, 2A-2D	INT0~INT3 割り込み キー入力割り込み (KI0~KI3)
	R8C/26-29 R8C/2A-2D	INT0、INT1、INT3 割り込み キー入力割り込み (KI0~KI3)
H8/300H Tiny	H8/3687, 36077 H8/36049, 36109	IRQ0~IRQ3 割り込み WKP 割り込み (WKP0~WKP5)
SH/Tiny	SH7125	IRQ0~IRQ3 割り込み NMI 割り込み

割り込みの設定ダイアログを図 4.5.1 に示します。

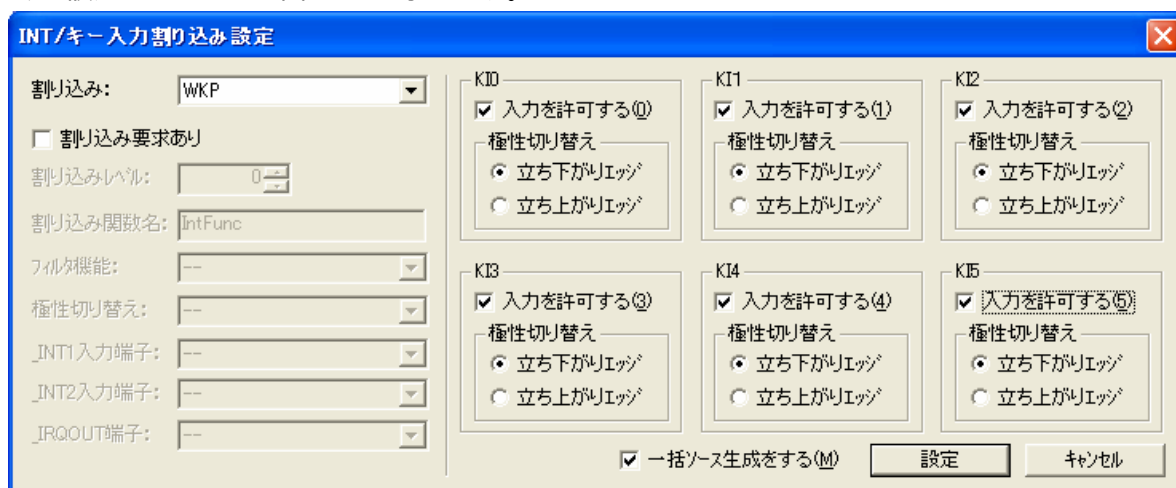


図 4.5.1 割り込み設定ダイアログ

割り込み設定ダイアログでは、割り込み発生を検出を行うための設定を行います。検出する割り込みを選択して設定すると、割り込み発生時に[割り込み関数名]で指定した名前前の関数が呼び出されます。

### [割り込み]

設定する割り込みを選択してください。

### [割り込み要求あり]

[割り込み]で指定した割り込み要求をイネーブルにします。設定の有無はマイコンにより異なります。

[割り込み要求あり]の設定内容を表 4.5.2 に示します。

表4.5.2 [割り込み要求あり] 設定内容

マイコン	内容
H8/3687, 36077 H8/36049, 36109	非チェック時: [割り込み]で選択した割り込み要求を禁止します チェック時: [割り込み]で選択した割り込み要求を許可します



## [割り込みレベル]

割り込み優先レベルを設定してください。割り込みレベルの値を 1 以上にすると、割り込み関数の設定が可能です。設定の有無はマイコンにより異なります。

## [割り込み関数名]

割り込み発生時に呼び出される通知関数名を指定します。プログラムに指定した名前で関数を追加してください。関数の宣言は次のようになります。

関数宣言
void [設定した通知関数名](void);

## [フィルタ機能]

フィルタ機能の設定です。設定の有無はマイコン、割り込み種別により異なります。フィルタ機能の設定内容を表 4.5.3 に示します。

表4.5.3 フィルタ機能設定内容

マイコン	項目	内容
R8C/13 (INT0,3割り込み)	フィルタなし	フィルタ機能を使用しません
R8C/22-25 (INT0~3割り込み)	f1 でサンプリング	サンプリングクロックごとにレベルをサンプリングし、レベルが 3 度一致した時点で、割り込み要求ありになります。
R8C/2A-2D (INT0~3割り込み)	f8 でサンプリング	
R8C/26-29 (INT0,1,3割り込み)	f32 でサンプリング	

## [極性切り替え]

割り込み入力極性を設定します。設定の有無はマイコン、割り込み種別により異なります。

## [\_INT1 入力端子]

[割り込み]で \_INT1 を選択した場合に INT1 割り込み入力端子を選択してください。設定の有無はマイコンにより異なります。\_INT1 入力端子の設定内容を表 4.5.4 に示します。

表4.5.4 \_INT1入力端子設定内容

シリーズ(グループ)	項目	内容
R8C/Tiny (26-27) (2A-2D)	P1_5 端子	INT1 割り込み入力端子を選択してください
	P1_7 端子	
	P3_6 端子	
R8C/Tiny (28-29)	P1_5 端子	
	P1_7 端子	

## [\_INT2 入力端子]

[割り込み]で \_INT2 を選択した場合に INT2 割り込み入力端子を選択してください。設定の有無は CPU により異なります。\_INT2 入力端子の設定内容を表 4.5.5 に示します。

表4.5.5 \_INT2入力端子設定内容

マイコン	項目	内容
R8C/2A-2D	P3_2 端子	INT2 割り込み入力端子を選択してください
	P6_6 端子	



## [\_IRQOUT 端子]

[割り込み]で IRQ 割り込みを選択した場合に IRQOUT 端子機能を選択してください。設定の有無はマイコンにより異なります。\_IRQOUT 端子の設定内容を表 4.5.6 に示します。

表4.5.6 \_IRQOUT端子設定内容

マイコン	項目	内容
SH7125	割り込み要求受出力	_IRQOUT 端子から割り込みの発生を知らせる信号を出力します
	常にハイレベル出力	_IRQOUT 端子から常にハイレベルを出力します

## [KI0] ~ [KI5]

キー入力割り込み、または WKP 割り込みを使用する場合、使用する端子をチェックし、各端子の設定を行います。対応する CPU を表 4.5.7 に示します。

表4.5.7 KI0~KI5設定内容

マイコン	項目	内容
R8C/ 13,22-29, 2A-2D M16C/28,28B,29 M16C/62P	KI0~KI3	KI0~KI3 端子に対応します。使用する端子の[入力を許可する]をチェックしてください。
H8/3687, 36077,36049, 36109	KI0~KI5	WKP0~WKP5 端子に対応します。使用する端子の[入力を許可する]をチェックしてください。

## [極性切り替え]

キー入力割り込み、または WKP 割り込みの入力極性を指定してください。

## [一括ソース生成をする]

チェックすると、[設定]ボタンクリック時にドライバソースを生成します。

## [設定]

設定した内容を保存し、ダイアログを閉じます。[一括ソース生成をする]にチェックされている場合はドライバソースを生成します。

## [キャンセル]

設定した内容を保存せずにダイアログを閉じます。

## 4.6 A/D変換の設定

A/D 変換で設定可能な各モードの内容は次の通りです。

- 単発モード  
選択した 1 本の端子の入力電圧を 1 回 A/D 変換するモードです。
- 繰り返しモード  
選択した 1 本の端子の入力電圧を繰り返し A/D 変換するモードです。
- 単掃引モード  
選択した複数の端子の入力電圧を 1 回ずつ A/D 変換するモードです。
- 繰り返し掃引モード 0  
選択した複数の端子の入力電圧を繰り返し A/D 変換するモードです。
- 繰り返し掃引モード 1  
選択した端子に重点をおいて全端子の入力電圧を繰り返し A/D 変換するモードです。
- 同時サンプル掃引モード  
選択した端子の入力電圧を 1 回ずつ A/D 変換するモードです。その際、サンプル&ホールド回路を 2 回路用いて、AN0 と AN1 の入力電圧のサンプリングを同時に行います。
- 遅延トリガモード 0  
M16C/28,28B,29 のみ対応します。タイマ B0 のアンダフローにより単掃引変換を開始します。AN0 端子の変換終了後は、タイマ B1 のアンダフローが発生するまで AN1 端子のサンプリングおよび変換を行いません。タイマ B1 アンダフローが発生すると、AN1 端子以降の単掃引変換を再開します。
- 遅延トリガモード 1  
M16C/28,28B,29 のみ対応します。ADTRG 端子の入力が “H” から “L” に変化(立ち下がり)したとき単掃引変換を開始します。AN0 端子の変換終了後は、2 回目の ADTRG 端子の立ち下がりが発生するまで AN1 端子のサンプリングおよび変換を行いません。2 回目の ADTRG 端子の立ち下がりが発生すると、AN1 端子以降の単掃引変換を再開します。
- 2 チャンルスキャンモード  
SH7125 の各 A/D モジュール(A/D モジュール 0 または A/D モジュール 1) の 4 チャンルのアナログ入力をグループ 0 とグループ 1 に分け、グループ 0 とグループ 1 に個別のトリガによる起動要因を選択できます。ADI 割り込み要求の発行はグループ 0 トリガ、グループ 1 トリガそれぞれの変換終了時と、グループ 0 トリガ、グループ 1 トリガ両方の変換終了時から選択できます。選択した入力のア/D 変換が終了すると、A/D 変換器は待機状態になります。
- 2 チャンル連続スキャンモード  
2 チャンルスキャンモードのグループ 0 トリガとグループ 1 トリガの変換終了後も引き続きトリガの変換を行います。
- 4 チャンルスキャンモード  
SH7125 の A/D モジュール 0 (AN0 端子 ~ AN3 端子) または A/D モジュール 1 (AN4 端子 ~ AN7 端子) より 1 ~ 4 チャンルを選択し、選択したアナログ入力チャンネルを番号の小さい順に A/D 変換します。A/D 変換結果は順次そのチャンネルに対応する A/D データレジスタに転送されます。選択されたすべてのチャンネルの A/D 変換が終了すると、ADI 割り込み要求を発行し、A/D 変換器は待機状態になります。
- 4 チャンル連続スキャンモード  
4 チャンルスキャンモードと同様、選択した最大 4 チャンルのアナログ入力チャンネルを番号の小さい順に A/D 変換します。選択されたすべてのチャンネルの A/D 変換が終了すると、ADI 割り込み要求を発生し、再び第 1 チャンルから A/D 変換を開始します。

各マイコンが対応する A/D 変換モードを表 4.6.1 に示します。

表4.6.1 各マイコンのA/D変換モード

	M16C/60	M16C/Tiny	R8C/Tiny	H8/300H Tiny	SH/Tiny
	M16C/62p	M16C/28 M16C/28B M16C/29	R8C/13 R8C/22-29 R8C/2A-2D	H8/3687, H8/36077, H8/36049, H8/36109	SH7125
単発モード	●	●	●	●	●
繰り返しモード	●	●	●	●	-
単掃引モード	●	●	-	-	-
繰り返し掃引モード0	●	●	-	-	-
繰り返し掃引モード1	●	●	-	-	-
同時サンプル掃引モード	-	●	-	-	-
遅延トリガモード0	-	●	-	-	-
遅延トリガモード1	-	●	-	-	-
2チャンネルスキャンモード	-	-	-	-	●
2チャンネル連続スキャンモード	-	-	-	-	●
4チャンネルスキャンモード	-	-	-	-	●
4チャンネル連続スキャンモード	-	-	-	-	●

各モードで設定ダイアログは共通です。A/D変換の設定ダイアログを図4.24に示します。

図4.6.1 A/D変換設定ダイアログ

## [入力グループ]

設定するアナログ入力端子の含まれるグループを指定してください。設定方法はマイコンにより異なります。入力グループの設定方法を表 4.6.2 に示します。

表4.6.2 入力グループ設定方法

マイコン	設定方法
M16C/62P	ANEXグループまたはアナログ入力端子と共用するポートのグループをPiグループ(i=0,2,10)より選択してください。
M16C/28,28B	アナログ入力端子と共用するポートのグループを P0 グループ、P10 グループ、P1/P9 グループより選択してください。
M16C/29	アナログ入力端子と共用するポートのグループを P0 グループ、P10 グループ、P1/P9 グループ、P9 グループより選択してください。
R8C/13,22-27, 2A-2B	アナログ入力端子と共用するポートのグループを P0 グループ、P1 グループより選択してください。
R8C/28-29	アナログ入力端子グループがありません。常に“-”(グループ指定なし)が選択されます。
R8C/2C-2D	アナログ入力端子と共用するポートのグループをPiグループ(i=0,1,7,12,16)より選択してください。
H8/3687, 36077,36049	アナログ入力端子グループをAN0グループ、AN4グループより選択してください。
H8/36109	アナログ入力端子グループをAniグループ(i=0,4,8,12)より選択してください。
SH7125	アナログ入力端子グループがありません。常に“-”(グループ指定なし)が選択されます。

## [入力端子]

設定するアナログ入力端子を選択してください。モードにより選択可能な端子が異なります。複数の端子を使用するモードでは複数端子を指定可能です。

表 4.6.3 に、SH7125 の 2 チャンルスキャンモードおよび 2 チャンル連続スキャンモードでの[入力端子]の設定内容を示します。

表4.6.3 [入力端子] 設定内容 (SH7125 2チャンネルスキャンモード、2チャンネル連続スキャンモード)

マイコン	項目	モジュール	内容
SH7125	AN0	A/D_0	AN0 を選択します
	AN0-AN1	A/D_0	AN0 と AN1 を選択します
	AN2	A/D_0	AN2 を選択します
	AN2-AN3	A/D_0	AN2 と AN3 を選択します
	AN4	A/D_1	AN4 を選択します
	AN4-AN5	A/D_1	AN4 と AN5 を選択します
	AN6	A/D_1	AN6 を選択します
	AN6-AN7	A/D_1	AN6 と AN7 を選択します
	AN0/AN2	A/D_0	AN0 をグループ 0 に、AN2 をグループ 1 に選択します
	AN0_1/AN2_3	A/D_0	AN0 と AN1 をグループ 0 に、AN2 と AN3 をグループ 1 に選択します
	AN4/AN6	A/D_1	AN4 をグループ 0 に、AN6 をグループ 1 に選択します
	AN4_5/AN6_7	A/D_1	AN4 と AN5 をグループ 0 に、AN6 と AN7 をグループ 1 に選択します

## [モジュール]

設定する A/D 変換モジュールが表示されます。表示内容を表 4.6.4 に示します。

表4.6.4 モジュール表示内容

マイコン	表示内容
SH7125	[入力端子]で選択した端子に対応する A/D 変換モジュール(AD_0または AD_1)を表示します。各 A/D 変換モジュールで 1 つの設定を作成することが可能です。例えば最初に AD_0 の設定をした場合、次に A/D 変換設定ダイアログを開くと、[入力端子]には AD_1 に対応する端子のみ表示されます。

## [サンプル&amp;ホールド]

サンプル&ホールド機能の設定です。サンプル&ホールド機能を使用する場合は[サンプル&ホールドあり]を選択してください。

## [変換速度]

A/D 変換速度の設定です。動作クロック又は A/D 変換時間を選択してください。

## [変換動作]

ダイアログで指定した状態に A/D 変換を初期設定した直後の動作を[変換開始]、[変換停止]から選択します。[トリガ]で[ソフトウェアトリガ]を選択した状態で[変換開始]を選択した場合、初期設定とともに変換を開始します。[トリガ]で[ソフトウェアトリガ]以外を選択し、[変換開始]を選択した場合、初期設定後のトリガで変換を開始します。[変換停止]を選択した場合、[トリガ]の選択にかかわらず変換を行いません。

## [分解能]

分解能をビット数で指定してください。選択できる項目はマイコンにより異なります。

## [外部オペアンプ接続]

外部オペアンプによりアナログ入力を増幅する場合の設定です。設定の有無はマイコンにより異なります。表 4.6.5 に外部オペアンプ接続の設定内容を示します。

表4.6.5 外部オペアンプ接続設定内容

マイコン	設定	内容
M16C/62P	ANEX0,ANEX1 は使用しない	[入力]グループに P10、P0 または P2 グループを指定し、指定した端子のアナログ入力を増幅せずに変換する場合に選択します。
	外部オペアンプ接続モード	[入力]グループに P10、P0 または P2 グループを指定し、指定した端子のアナログ入力を外部オペアンプにより増幅する場合に選択します。
	ANEX0 入力を A/D 変換	アナログ入力端子に ANEX0 を指定した場合は常に選択されます。
	ANEX1 入力を A/D 変換	アナログ入力端子に ANEX1 を指定した場合は常に選択されます

## [(グループ 0)トリガ]

A/D 開始条件を指定してください。SH7125 の 2 チャネルスキャンモードおよび 2 チャネル連続スキャンモードでは、グループ 0 のトリガです。選択項目はマイコンにより異なります。設定内容を表 4.6.6 に示します。

表4.6.6 (グループ0)トリガ設定内容

マイコン	項目	内容
全機種	ソフトウェアトリガ	レジスタの A/D 変換開始ビットにより変換を開始します。後述の[変換動作]で[変換開始]を指定した場合、A/D変換の初期設定とともに変換を開始します。
M16C/62P	ハードウェアトリガ	ADTRG 端子の入力の立ち下がりを変換開始トリガとします。
M16C/28,28B	ハードウェアトリガ	ADTRG 端子の入力の立ち下がりを変換開始トリガとします。
M16C/29	タイマB0アンダフロー	同時サンプル掃引モード時のみ選択可能です。使用する場合はタイマを設定してください。
	タイマB2割り込み	
	B2割り込み発生頻度設定 カウンタアンダフロー	
R8C/22-25 R8C/2A-2D	タイマ RD (相補 PWM モード)	タイマ RD 割り込みを変換開始トリガとします。
H8/3687,36077 H8/36049,36109	ハードウェアトリガ	ADTRG 端子の入力を変換開始トリガとします。後述の[トリガ極性]により立ち上がりまたは立ち下がりエッジを選択可能です。
SH7125	外部トリガの入力	外部トリガ端子 (ADTRG) の入力をトリガとします。
	TRGAN	MTU2 各チャンネルの TRGA のインプットキャプチャ/コンペアマッチ、相補 PWM モード時、TCNT_4 のアンダフローをトリガとします。
	TRG0N	MTU2 チャンネル 0 コンペアマッチ(TRG0N)をトリガとします。
	TRG4AN	MTU2 A/D 変換開始要求ディレイド(TRG4AN)をトリガとします。
	TRG4BN	MTU2 A/D 変換開始要求ディレイド(TRG4BN)をトリガとします。

## [グループ 1 トリガ]

SH7125 の 2 チャンネルスキャンモードおよび 2 チャンネル連続スキャンモードのグループ 1 のトリガです。グループ 1 トリガの設定内容を表 4.6.7 にします。

表4.6.7 グループ1トリガ設定内容

マイコン	項目	内容
SH7125	ソフトウェアトリガ	各項目の内容は表 xxx と同様です。[(グループ0)トリガ]とは違う要因を選択してください。
	TRGAN	
	TRG0N	
	TRG4AN	
	TRG4BN	

## [トリガ極性]

外部入力を A/D 変換開始トリガに設定した際のトリガエッジを選択します。設定の有無はマイコンにより異なります。トリガ極性の設定内容を表 4.6.8 に示します。

表4.6.8 トリガ極性設定内容

マイコン	項目	内容
H8/3687 ※ H8/36077 ※	立ち上がりエッジ	ADTRG 端子の立ち上がりエッジを変換開始トリガとします。
H8/36049 ※ H8/36109 ※	立ち下がりエッジ	ADTRG 端子の立ち下がりエッジを変換開始トリガとします。
SH7125	立ち上がりエッジ	常に立ち上がりエッジが表示されますが、トリガ極性としては適用されません。

※ [トリガ]に[ハードウェアトリガ]を指定した場合のみ設定可能です。

## [ADF コントロール]

SH7125 の 2 チャンネルスキャンモードおよび 2 チャンネル連続スキャンモード選択時、グループ 0、1 それぞれの端子を選択した場合に、A/D エンドフラグ(ADF)をセットし ADI 割り込みを発生させるタイミングを指定してください。

表4.6.9 [ADFコントロール] 設定内容

マイコン	項目	内容
SH7125	グループ 0/1 それぞれ変換終了時に ADF セット	グループ 0 トリガ、グループ 1 トリガそれぞれの変換終了時に ADF がセットされ、ADI 割り込みが発生します。
	グループ 0/1 両方の変換終了時に ADF セット	グループ 0 トリガ、グループ 1 トリガ両方の変換終了時に ADF がセットされ、ADI 割り込みが発生します。なお、トリガの順番には影響されません。

## [割り込み]

A/D 変換終了時割り込みの検出を行うために設定します。[割り込みを許可する]をチェックし、割り込みレベルを設定すると、割り込み発生時に[割り込み関数名]で指定した名前の関数が呼び出されます。[<割り込み種別>割り込みを許可する]をチェックすると、割り込み通知関数名の設定が可能となります。各項目の設定方法を以下に示します。

## [割り込みを許可する]

A/D 変換終了時割り込みを検出する場合にチェックします。

## [&lt;割り込み種別&gt;割り込みレベル]

A/D 変換終了時割り込みの優先度を設定します。設定可能な割り込みレベルの値はマイコンにより異なります。

## [&lt;割り込み種別&gt;割り込み関数名]

A/D 変換終了時割り込み発生時に呼び出される通知関数名を指定します。割り込み通知関数を使用する場合は指定した名前に関数を追加してください。関数の宣言は次のようになります。

関数宣言
void [設定した通知関数名](void);

## [一括ソース生成をする]

チェックすると、[設定]ボタンクリック時にドライバソースを生成します。

## [設定]

設定した内容を保存し、ダイアログを閉じます。[一括ソース生成をする]にチェックされている場合はドライバソースを生成します。

## [キャンセル]

設定した内容を保存せずにダイアログを閉じます。



## 5. 生成関数リファレンス

### 5.1 M16C/60, M16C/Tiny, R8C/Tiny, H8/300H Tinyの生成関数

M16C/60 シリーズ(M16C/62P)、R8C/Tiny シリーズ(R8C/13,22-29.2A-2D)、H8/300H Tiny シリーズ(H8/3687,36077,36049,36109)の生成関数を表 5.1 に示します。

表5.1 M16C/60シリーズ、R8C/Tinyシリーズ、H8/300H Tinyシリーズの生成関数

No	機能	生成関数	内容
1-1	シリアル	_OpenSerialDriver	指定されたシリアル I/F を初期化する
1-2		_CloseSerialDriver	指定されたシリアル I/F をクローズする
1-3		_ConfigSerialDriverNotify	通知関数の登録する
1-4		_SetSerialFormat	シリアル設定を変更する
1-5		_SetSerialInterrupt	シリアル割り込みを設定する
1-6		_StartSerialReceiving	受信を開始する
1-7		_StartSerialSending	送信を開始する
1-8		_StopSerialReceiving	受信を停止する
1-9		_StopSerialSending	送信を停止する
1-10		_PollingSerialReceiving	ポーリング受信
1-11		_PollingSerialSending	ポーリング送信
2-1	割り込み	_SetInterrupt	外部割り込みの設定
2-2		_EnableInterrupt	外部割り込みの制御
2-3		_DisableInterrupt	外部割り込みの制御
2-4		_GetInterruptFlag	外部割り込みフラグの状態取得
2-5		_ClearInterruptFlag	外部割り込みフラグのクリア
3-1	A/D変換	_CreateADC	A/D変換器の設定
3-2		_EnableADC	A/D変換器の動作制御
3-3		_DisableADC	A/D変換器の動作制御
3-4		_DestroyADC	A/D変換器の設定破棄
3-5		_GetADC	A/D変換値の取得
3-6		_GetADCAI	A/D変換値の取得(全レジスタ)
3-7		_GetADCStatus	A/D変換器の状態の取得
3-8		_ClearADCStatus	A/D変換器の状態のクリア
4-1	I/Oポート	_SetIOPort	I/Oポートの設定
4-2		_ReadIOPort	I/Oポートからの読み出し
4-3		_WriteIOPort	I/Oポートへの書き込み
4-4		_ReadIOPortRegister	I/Oポートレジスタからの読み出し
4-5		_WriteIOPortRegister	I/Oポートレジスタへの書き込み
5-1	タイマ/タイマモード	_CreateTimer	タイマモード設定
5-2		_EnableTimer	タイマモード動作開始
5-3		_DisableTimer	タイマモード動作停止
5-4		_DestroyTimer	タイマモード設定破棄
6-1	タイマ/イベントカウンタモード	_CreateEventCounter	イベントカウンタモード設定
6-2		_EnableEventCounter	イベントカウンタモード動作開始
6-3		_DisableEventCounter	イベントカウンタモード動作停止
6-4		_DestroyEventCounter	イベントカウンタモード設定破棄
6-5		_GetEventCounter	イベントカウンタモードカウンタ値取得
7-1	タイマ/パルス幅変調モード	_CreatePulseWidthModulationMode	パルス幅変調モード設定
7-2		_EnablePulseWidthModulationMode	パルス幅変調モード動作開始
7-3		_DisablePulseWidthModulationMode	パルス幅変調モード動作停止
7-4		_DestroyPulseWidthModulationMode	パルス幅変調モード設定破棄
8-1	タイマ/パルス周期測定モード	_CreatePulsePeriodMeasurementMode	パルス周期測定モード設定
8-2		_EnablePulsePeriodMeasurementMode	パルス周期測定モード動作開始

No	機能	生成関数	機能
8-3		_DisablePulsePeriodMeasurementMode	パルス周期測定モード動作停止
8-4		_DestroyPulsePeriodMeasurementMode	パルス周期測定モード設定破棄
8-5		_GetPulsePeriodMeasurementMode	パルス周期測定モード測定値取得
9-1	タイマ/パルス幅測定モード	_CreatePulseWidthMeasurementMode	パルス幅測定モード設定
9-2		_EnablePulseWidthMeasurementMode	パルス幅測定モード動作開始
9-3		_DisablePulseWidthMeasurementMode	パルス幅測定モード動作停止
9-4		_DestroyPulseWidthMeasurementMode	パルス幅測定モード設定破棄
9-5		_GetPulseWidthMeasurementMode	パルス幅測定モード測定値取得
10-1	タイマ/インプットキャプチャモード	_CreateInputCapture	インプットキャプチャモード設定
10-2		_EnableInputCapture	インプットキャプチャモード動作開始
10-3		_DisableInputCapture	インプットキャプチャモード動作停止
10-4		_DestroyInputCapture	インプットキャプチャモード設定破棄
10-5		_GetInputCapture	インプットキャプチャモードカウンタ値取得
11-1	タイマ/アウトプットコンペアモード	_CreateOutputCompare	アウトプットコンペアモード設定設定
11-2		_EnableOutputCompare	アウトプットコンペアモード動作開始
11-3		_DisableOutputCompare	アウトプットコンペアモード動作停止
11-4		_DestroyOutputCompare	アウトプットコンペアモード設定破棄

各生成関数の仕様を以下に示します。

#### ・シリアル

##### 1-1 \_\_OpenSerialDriver

生成関数	Boolean __OpenSerialDriver_[モード]_U[リソース]_p[設定番号] (void)
周辺機能	シリアル
内容	指定されたシリアル I/F を初期化します。
パラメータ	—
戻り値	RAPI_TRUE を返します。

##### 1-2 \_\_CloseSerialDriver

生成関数	Boolean __CloseSerialDriver_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	指定されたシリアル I/F をクローズします。
パラメータ	—
戻り値	RAPI_TRUE を返します。

##### 1-3 \_\_ConfigSerialDriverNotify

生成関数	Boolean __ConfigSerialDriverNotify_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	指定された種別の通知関数をドライバに登録します。
パラメータ	—
戻り値	RAPI_TRUE を返します。

##### 1-4 \_\_SetSerialFormat

生成関数	Boolean __SetSerialFormat_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	シリアル設定を変更します。
パラメータ	—
戻り値	シリアル通信の設定に成功した場合は RAPI_TRUE を返し、設定に失敗した場合は RAPI_FALSE を返します。

##### 1-5 \_\_SetSerialInterrupt

生成関数	Boolean __SetSerialInterrupt_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	シリアル割り込みを設定します。
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 1-6 \_\_StartSerialReceiving

生成関数	Boolean __StartSerialReceiving_[モード]_U[資源]_p[設定番号] (unsigned char wordNum, unsigned int *dataBuf)
周辺機能	シリアル
内容	受信を開始します。
パラメータ	wordNum : 受信ワード数 dataBuf : 受信データを格納するバッファポインタ
戻り値	シリアル通信の受信開始が成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## 1-7 \_\_StartSerialSending

生成関数	Boolean __StartSerialSending_[モード]_U[資源]_p[設定番号] (unsigned char wordNum, unsigned int *dataBuf)
周辺機能	シリアル
内容	送信を開始します。
パラメータ	wordNum : 送信バイト数 dataBuf : 送信データが格納されているバッファポインタ
戻り値	シリアル通信の送信開始が成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## 1-8 \_\_StopSerialReceiving

生成関数	Boolean __StopSerialReceiving_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	受信を停止します。
パラメータ	—
戻り値	シリアル通信の受信の停止に成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。
備考	M16C/62P,M16C/28,28B,29 の SI/O3,4 では生成されません

## 1-9 \_\_StopSerialSending

生成関数	Boolean __StopSerialSending_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	送信を停止します。
パラメータ	—
戻り値	シリアル通信の送信の停止に成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。
備考	M16C/62P,M16C/28,28B,29 の SI/O3,4 では生成されません

## 1-10 \_\_PollingSerialReceiving

生成関数	Boolean __PollingSerialReceiving_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	ポーリング受信します。
パラメータ	—
戻り値	未受信データがある場合は RAPI_TRUE を返し、受信完了した場合は RAPI_FALSE を返します。

## 1-11 \_\_PollingSerialSending

生成関数	Boolean __PollingSerialSending_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	ポーリング送信します。
パラメータ	—
戻り値	未送信データがある場合は RAPI_TRUE を返し、送信完了した場合は RAPI_FALSE を返します。

## ・ 割り込み

## 2-1 \_\_SetInterrupt

生成関数	Boolean __SetInterrupt_I[資源]_p[設定番号] (void)
周辺機能	割り込み
内容	外部割り込みの設定
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 2-2 \_\_EnableInterrupt

生成関数	Boolean __EnableInterrupt_I[資源]_p[設定番号] (void)
周辺機能	割り込み
内容	外部割込みの制御
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 2-3 \_\_DisableInterrupt

生成関数	Boolean __DisableInterrupt_I[資源]_p[設定番号] (void)
周辺機能	割り込み
内容	外部割込みの制御
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 2-4 \_\_GetInterruptFlag

生成関数	Boolean __GetInterruptFlag_I[資源]_p[設定番号] (unsigned int *data)
周辺機能	割り込み
内容	外部割込みフラグの状態取得
パラメータ	data : 取得したフラグデータを格納するバッファポインタ
戻り値	RAPI_TRUE を返します。

## 2-5 \_\_ClearInterruptFlag

生成関数	Boolean __ClearInterruptFlag_I[資源]_p[設定番号] (void)
周辺機能	割り込み
内容	外部割込みフラグのクリア
パラメータ	—
戻り値	RAPI_TRUE を返します。

## ・ A/D 変換

## 3-1 \_\_CreateADC

生成関数	Boolean __CreateADC_[モード]_[端子番号]_[端子数]_p[設定番号] (void)
周辺機能	A/D 変換
内容	A/D 変換器の設定
パラメータ	—
戻り値	設定に成功した場合は RAPI_TRUE を返し、設定に失敗した場合は RAPI_FALSE を返します。

## 3-2 \_\_EnableADC

生成関数	Boolean __EnableADC_[モード]_[端子番号]_[端子数]_p[設定番号] (void)
周辺機能	A/D 変換
内容	A/D 変換器の動作制御
パラメータ	—
戻り値	制御に成功した場合は RAPI_TRUE を返し、制御に失敗した場合は RAPI_FALSE を返します。
備考	M16C/28,28B,29 の遅延トリガ 0.1 モードでは生成されません

## 3-3 \_\_DisableADC

生成関数	Boolean __DisableADC_[モード]_[端子番号]_[端子数]_p[設定番号] (void)
周辺機能	A/D 変換
内容	A/D 変換器の動作制御
パラメータ	—
戻り値	制御に成功した場合は RAPI_TRUE を返し、制御に失敗した場合は RAPI_FALSE を返します。
備考	M16C/28,28B,29 の遅延トリガ 0.1 モードでは生成されません

## 3-4 \_\_DestroyADC

生成関数	Boolean __DestroyADC_[モード]_[端子番号]_[端子数]_p[設定番号] (void)
周辺機能	A/D 変換
内容	A/D 変換器の設定破棄
パラメータ	—
戻り値	設定破棄に成功した場合は RAPI_TRUE を返し、設定破棄に失敗した場合は RAPI_FALSE を返します。

## 3-5 \_\_GetADC

生成関数	Boolean __GetADC_[モード]_[端子番号]_[端子数]_p[設定番号]_ad[レジスタ] (unsigned int *data)
周辺機能	A/D 変換
内容	A/D 変換値の取得
パラメータ	data : A/D 変換値を格納するポインタ
戻り値	A/D 変換値取得に成功した場合は RAPI_TRUE を返し、A/D 変換値取得に失敗した場合は RAPI_FALSE を返します。

## 3-6 \_\_GetADCAI

生成関数	Boolean __GetADCAI_[モード]_[端子番号]_[端子数]_p[設定番号] (unsigned int *data)
周辺機能	A/D 変換
内容	A/D 変換値の取得(全レジスタ)
パラメータ	data : A/D 変換値を格納するポインタ
戻り値	A/D 変換値取得に成功した場合は RAPI_TRUE を返し、A/D 変換値取得に失敗した場合は RAPI_FALSE を返します。

## 3-7 \_\_GetADCStatus

生成関数	Boolean __GetADCStatus_[モード]_[端子番号]_[端子数]_p[設定番号] (unsigned int *data)
周辺機能	A/D 変換
内容	A/D 変換器の状態の取得
パラメータ	data : A/D 変換器の状態を示すレジスタの内容を格納するポインタ。*data の下位 1 ビット目に割り込みビットの状態(M16C または R8C 使用時)または A/D エンドフラグの値(H8/300H 使用時)を格納します。また*data の下位 2 ビット目に A/D 変換開始フラグの状態を格納します。M16C で使用する場合には、*data の上位 8 ビットに A/D 変換ステータスレジスタ 0 の値を格納します。
戻り値	A/D 変換器の状態取得に成功した場合は RAPI_TRUE を返し、A/D 変換器の状態取得取得に失敗した場合は RAPI_FALSE を返します。

## 3-8 \_\_ClearADCStatus

生成関数	Boolean __ClearADCStatus_[モード]_[端子番号]_[端子数]_p[設定番号] (unsigned int data)
周辺機能	A/D 変換
内容	A/D 変換器の状態のクリア
パラメータ	data : 指定した A/D 変換器の状態フラグをクリアします。data の下位 1 ビット目に割り込みビットの状態(M16C または R8C 使用時)または A/D エンドフラグの値(H8/300H 使用時)を指定します。M16C で使用する場合には、data の上位 8 ビットに A/D 変換ステータスレジスタ 0 の値を指定します。クリアするビットには 0 を設定し、クリアしないビットには 1 を設定します。
戻り値	A/D 変換器の状態フラグのクリアに成功した場合は RAPI_TRUE を返し、A/D 変換器の状態フラグのクリア取得に失敗した場合は RAPI_FALSE を返します。

## ・ I/O ポート

## 4-1 \_\_SetIOPort

生成関数	Boolean __SetIOPort_[ポート]_[端子番号]_p[設定番号] (void)
周辺機能	I/O ポート
内容	I/O ポートの設定
パラメータ	—
戻り値	RAPI_TRUE を返します。
備考	H8/3687,36049,36077 の RB 端子、H8/36109 の RF 端子、M16C/62P の P85 端子、R8C/13,2A-2D の P46,47 端子、R8C/22-29 の P42,46,47 端子では出力されません

## 4-2 \_\_ReadIOPort

生成関数	Boolean __ReadIOPort_[ポート]_[端子番号]_p[設定番号] (unsigned int *data)
周辺機能	I/O ポート
内容	I/O ポートからの読み出し
パラメータ	data : I/O ポートから読み出した値を格納する変数のポインタ
戻り値	RAPI_TRUE を返します。

## 4-3 \_\_WriteIOPort

生成関数	Boolean __WriteIOPort_[ポート]_[端子番号]_p[設定番号] (unsigned int data)
周辺機能	I/O ポート
内容	I/O ポートへの書き込み
パラメータ	data : I/O ポートに書き込むデータ
戻り値	RAPI_TRUE を返します。
備考	H8/3687,36049,36077 の RB 端子、H8/36109 の RF 端子、M16C/62P の P85 端子、R8C/13,2A-2D の P46,47 端子、R8C/22-29 の P42,46,47 端子では出力されません

## 4-4 \_\_ReadIOPortRegister

生成関数	Boolean __ReadIOPortRegister_[ポート]_p[設定番号] (unsigned int *data)
周辺機能	I/O ポート
内容	I/O ポートレジスタからの読み出し
パラメータ	data : I/O ポートレジスタから読み出した値を格納する変数のポインタ
戻り値	RAPI_TRUE を返します。

## 4-5 \_\_WriteIOPortRegister

生成関数	Boolean __WriteIOPortRegister_[ポート]_p[設定番号] (unsigned int data)
周辺機能	I/O ポート
内容	I/O ポートレジスタへの書き込み
パラメータ	data : I/O ポートレジスタに書き込むデータ
戻り値	RAPI_TRUE を返します。
備考	H8/3687,36049,36077 の RB 端子、H8/36109 の RF 端子、M16C/62P の P85 端子、R8C/13,2A-2D の P46,47 端子、R8C/22-29 の P42,46,47 端子では出力されません

## ・ タイマ/タイマモード

## 5-1 \_\_CreateTimer

生成関数	Boolean __CreateTimer_T[資源]_p[設定番号] (void)
周辺機能	タイマ/タイマモード
内容	タイマモード設定
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 5-2 \_\_EnableTimer

生成関数	Boolean __EnableTimer_T[資源]_p[設定番号] (void)
周辺機能	タイマ/タイマモード
内容	タイマモード動作制御(動作開始)
パラメータ	—
戻り値	RAPI_TRUE を返します。
備考	H8/3687,36049,36077,36109 のタイマ B1 では出力されません

## 5-3 \_\_DisableTimer

生成関数	Boolean __DisableTimer_T[資源]_p[設定番号] (void)
周辺機能	タイマ/タイマモード
内容	タイマモード動作制御(動作停止)
パラメータ	—
戻り値	RAPI_TRUE を返します。
備考	H8/3687,36049,36077,36109 のタイマ B1 では出力されません

## 5-4 \_\_DestroyTimer

生成関数	Boolean __DestroyTimer_T[資源]_p[設定番号] (void)
周辺機能	タイマ/タイマモード
内容	タイマモード設定破棄
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 5-5 \_\_DestroyTimer

生成関数	Boolean __DestroyTimer_T[資源]_p[設定番号] (void)
周辺機能	タイマ/タイマモード
内容	タイマモード設定破棄
パラメータ	—
戻り値	RAPL_TRUE を返します。

## ・タイマ/イベントカウンタモード

## 6-1 \_\_CreateEventCounter

生成関数	Boolean __CreateEventCounter_T[資源]_p[設定番号] (void)
周辺機能	タイマ/イベントカウンタモード
内容	イベントカウンタモード設定
パラメータ	—
戻り値	RAPL_TRUE を返します。

## 6-2 \_\_EnableEventCounter

生成関数	Boolean __EnableEventCounter_T[資源]_p[設定番号] (void)
周辺機能	タイマ/イベントカウンタモード
内容	イベントカウンタモード動作制御(動作開始)
パラメータ	—
戻り値	RAPL_TRUE を返します。
備考	H8/3687,36049,36077,36109 のタイマ B1 では出力されません

## 6-3 \_\_DisableEventCounter

生成関数	Boolean __DisableEventCounter_T[資源]_p[設定番号] (void)
周辺機能	タイマ/イベントカウンタモード
内容	イベントカウンタモード動作制御(動作停止)
パラメータ	—
戻り値	RAPL_TRUE を返します。
備考	H8/3687,36049,36077,36109 のタイマ B1 では出力されません

## 6-4 \_\_DestroyEventCounter

生成関数	Boolean __DestroyEventCounter_T[資源]_p[設定番号] (void)
周辺機能	タイマ/イベントカウンタモード
内容	イベントカウンタモード設定破棄
パラメータ	—
戻り値	RAPL_TRUE を返します。

## 6-5 \_\_GetEventCounter

生成関数	Boolean __GetEventCounter_T[資源]_p[設定番号] (unsigned int *data)
周辺機能	タイマ/イベントカウンタモード
内容	イベントカウンタモードカウンタ値取得
パラメータ	data : カウンタ値を格納するバッファポインタ
戻り値	RAPL_TRUE を返します。

## ・タイマ/パルス幅変調モード

## 7-1 \_\_CreatePulseWidthModulationMode

生成関数	Boolean __CreatePulseWidthModulationMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅変調モード
内容	パルス幅変調モード設定
パラメータ	—
戻り値	RAPL_TRUE を返します。



## 7-2 \_\_EnablePulseWidthModulationMode

生成関数	Boolean __EnablePulseWidthModulationMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅変調モード
内容	パルス幅変調モード動作制御(動作開始)
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 7-3 \_\_DisablePulseWidthModulationMode

生成関数	Boolean __DisablePulseWidthModulationMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅変調モード
内容	パルス幅変調モード動作制御(動作停止)
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 7-4 \_\_DestroyPulseWidthModulationMode

生成関数	Boolean __DestroyPulseWidthModulationMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅変調モード
内容	パルス幅変調モード設定破棄
パラメータ	—
戻り値	RAPI_TRUE を返します。

## ・タイマ/パルス周期測定モード

## 8-1 \_\_CreatePulsePeriodMeasurementMode

生成関数	Boolean __CreatePulsePeriodMeasurementMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス周期測定モード
内容	パルス周期測定モード設定
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 8-2 \_\_EnablePulsePeriodMeasurementMode

生成関数	Boolean __EnablePulsePeriodMeasurementMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス周期測定モード
内容	パルス周期測定モード動作制御(動作開始)
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 8-3 \_\_DisablePulsePeriodMeasurementMode

生成関数	Boolean __DisablePulsePeriodMeasurementMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス周期測定モード
内容	パルス周期測定モード動作制御(動作停止)
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 8-4 \_\_DestroyPulsePeriodMeasurementMode

生成関数	Boolean __DestroyPulsePeriodMeasurementMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス周期測定モード
内容	パルス周期測定モード設定破棄
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 8-5 \_\_GetPulsePeriodMeasurementMode

生成関数	Boolean __GetPulsePeriodMeasurementMode_T[資源]_p[設定番号] (unsigned int *data)
周辺機能	タイマ/パルス周期測定モード
内容	パルス周期測定モード測定値取得
パラメータ	data : カウンタ値を格納するバッファポインタ
戻り値	RAPI_TRUE を返します。

## ・ タイマ/パルス幅測定モード

## 9-1 \_\_CreatePulseWidthMeasurementMode

生成関数	Boolean __CreatePulseWidthMeasurementMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅測定モード
内容	パルス幅測定モード設定
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 9-2 \_\_EnablePulseWidthMeasurementMode

生成関数	Boolean __EnablePulseWidthMeasurementMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅測定モード
内容	パルス幅測定モード動作制御(動作開始)
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 9-3 \_\_DisablePulseWidthMeasurementMode

生成関数	Boolean __DisablePulseWidthMeasurementMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅測定モード
内容	パルス幅測定モード動作制御(動作停止)
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 9-4 \_\_DestroyPulseWidthMeasurementMode

生成関数	Boolean __DestroyPulseWidthMeasurementMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅測定モード
内容	パルス幅測定モード設定破棄
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 9-5 \_\_GetPulseWidthMeasurementMode

生成関数	Boolean __GetPulseWidthMeasurementMode_T[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅測定モード
内容	パルス幅測定モード測定値取得
パラメータ	—
戻り値	RAPI_TRUE を返します。

## ・ タイマ/インプットキャプチャモード

## 10-1 \_\_CreateInputCapture

生成関数	Boolean __CreateInputCapture_T[資源]_p[設定番号] (void)
周辺機能	タイマ/インプットキャプチャモード
内容	インプットキャプチャモード設定
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 10-2 \_\_EnableInputCapture

生成関数	Boolean __EnableInputCapture_T[資源]_p[設定番号] (void)
周辺機能	タイマ/インプットキャプチャモード
内容	インプットキャプチャモード動作制御(動作開始)
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 10-3 \_\_DisableInputCapture

生成関数	Boolean __DisableInputCapture_T[資源]_p[設定番号] (void)
周辺機能	タイマ/インプットキャプチャモード
内容	インプットキャプチャモード動作制御(動作停止)
パラメータ	—
戻り値	RAPL_TRUE を返します。

## 10-4 \_\_DestroyInputCapture

生成関数	Boolean __DestroyInputCapture_T[資源]_p[設定番号] (void)
周辺機能	タイマ/インプットキャプチャモード
内容	インプットキャプチャモード設定破棄
パラメータ	—
戻り値	RAPL_TRUE を返します。

## 10-5 \_\_GetInputCapture

生成関数	Boolean __GetInputCapture_T[資源]_p[設定番号] (unsigned int *data)
周辺機能	タイマ/インプットキャプチャモード
内容	インプットキャプチャモードカウンタ値取得
パラメータ	<p>data : 取得したカウンタ値を格納する配列のポインタを指定します。</p> <p>(M16C)</p> <ul style="list-style-type: none"> <li>[0]: ベースタイムレジスタの値を格納します。</li> <li>[1]: 時間計測レジスタ0 の値を格納します。</li> <li>[2]: 時間計測レジスタ1 の値を格納します。</li> <li>[3]: 時間計測レジスタ2 の値を格納します。</li> <li>[4]: 時間計測レジスタ3 の値を格納します。</li> <li>[5]: 時間計測レジスタ4 の値を格納します。</li> <li>[6]: 時間計測レジスタ5 の値を格納します。</li> <li>[7]: 時間計測レジスタ6 の値を格納します。</li> <li>[8]: 時間計測レジスタ7 の値を格納します。</li> </ul> <p>(R8C)</p> <ul style="list-style-type: none"> <li>・タイマ C 使用時 (RAPLTIMER_C を指定時) <ul style="list-style-type: none"> <li>[0]: タイマ C レジスタの値を格納します。</li> <li>[1]: キャプチャ、コンペア0 レジスタの値を格納します。</li> </ul> </li> <li>・タイマ RD 使用時 (RAPLTIMER_RD0~RAPLTIMER_RD1 を指定時) <ul style="list-style-type: none"> <li>[0]: タイマカウンタの値を格納します。</li> <li>[1]: ジェネラルレジスタ A の値を格納します。</li> <li>[2]: ジェネラルレジスタ B の値を格納します。</li> <li>[3]: ジェネラルレジスタ C の値を格納します。</li> <li>[4]: ジェネラルレジスタ D の値を格納します。</li> </ul> </li> </ul> <p>(H8/300H)</p> <ul style="list-style-type: none"> <li>[0]: タイマカウンタの値を格納します。</li> <li>[1]: ジェネラルレジスタ A の値を格納します。</li> <li>[2]: ジェネラルレジスタ B の値を格納します。</li> <li>[3]: ジェネラルレジスタ C の値を格納します。</li> <li>[4]: ジェネラルレジスタ D の値を格納します。</li> </ul>
戻り値	RAPL_TRUE を返します。

## ・タイマ/アウトプットコンペアモード

## 11-1 \_\_CreateOutputCompare

生成関数	Boolean __CreateOutputCompare_T[資源]_p[設定番号] (void)
周辺機能	タイマ/アウトプットコンペアモード
内容	アウトプットコンペアモード設定
パラメータ	—
戻り値	RAPL_TRUE を返します。

## 11-2 \_\_EnableOutputCompare

生成関数	Boolean __EnableOutputCompare_T[資源]_p[設定番号] (void)
周辺機能	タイマ/アウトプットコンペアモード
内容	アウトプットコンペアモード動作制御(動作開始)
パラメータ	—
戻り値	RAPL_TRUE を返します。

## 11-3 \_\_DisableOutputCompare

生成関数	Boolean __DisableOutputCompare_T[資源]_p[設定番号] (void)
周辺機能	タイマ/アウトプットコンペアモード
内容	アウトプットコンペアモード動作制御(動作停止)
パラメータ	—
戻り値	RAPL_TRUE を返します。

## 11-4 \_\_DestroyOutputCompare

生成関数	Boolean __DestroyOutputCompare_T[資源]_p[設定番号] (void)
周辺機能	タイマ/アウトプットコンペアモード
内容	アウトプットコンペアモード設定破棄
パラメータ	—
戻り値	RAPL_TRUE を返します。

## 5.2 SH/Tinyの生成関数

SH/Tiny シリーズ(SH7125) の生成関数を表 5.2 に示します。

表5.2 SH/Tinyシリーズの生成関数

No	機能	生成関数	内容
1-1	シリアル	_CreateSCI	シリアルの初期設定
1-2		_DestroySCI	シリアルポートのクローズ
1-3		_StartSCIReceiving	シリアル受信の開始
1-4		_StartSCISending	シリアル送信の開始
1-5		_StopSCIReceiving	シリアル受信の停止
1-6		_StopSCISending	シリアル送信の停止
1-7		_PollingSCIReceiving	ポーリングを用いたシリアル受信
1-8		_PollingSCISending	ポーリングを用いたシリアル送信
1-9		_GetSCIStatus	シリアル通信の各種送受信情報を取得
1-10		_ClearSCIStatus	シリアル通信の各種送受信情報をクリア
1-11		_OutputSCISck	調歩同期式モード時にSCK出力を制御
1-12		_OutputSCITxd	調歩同期式モード時にTXD出力を制御
2-1	割り込み	_CreateInterrupt	外部割り込みの初期設定
2-2		_EnableInterrupt	外部割り込みの有効・無効切替
2-3		_GetInterruptAndPinInfo_ALL	割込要求のステータスを取得
2-4		_ClearInterruptFlag_ALL	割込要求のステータスをクリア
3-1	A/D変換	_CreateADC	A/D変換器の初期設定
3-2		_EnableADC_RAPI	A/D変換器の動作開始、動作停止を制御
3-3		_DestroyADC_RAPI	A/D変換器の設定をクリア
3-4		_GetADC	A/D変換結果を取得
3-5		_GetADCFlag_RAPI	A/D変換器のステータスフラグを取得
3-6		_ClearADCFlag_RAPI	A/D変換器のステータスフラグをクリア
4-1	I/Oポート	_SetIOPort	I/Oポートの動作条件設定
4-2		_ReadIOPort_ALL	指定したI/Oポートのデータ取得
4-3		_WriteIOPort_ALL	指定したI/Oポートへデータ書き込み
5-1	タイマ/タイマモード	_CreateTimer	タイマの初期設定
5-2		_EnableTimer	タイマの動作開始、動作停止を制御
5-3		_DestroyTimer	タイマモードの設定をクリア
5-4		_DestroyTimer_ALL	タイマモードの設定をクリア
5-5		_GetTimerCounter	タイマのカウント値を取得
6-1	タイマ/イベントカウンタモード	_CreateEventCounter	イベントカウンタモードの初期設定
6-2		_EnableEventCounter	タイマの動作開始、動作停止を制御
6-3		_DestroyEventCounter	イベントカウンタモードの設定をクリア
6-4		_DestroyEventCounter_ALL	イベントカウンタモードの設定をクリア
6-5		_GetTimerCounter	タイマのカウント値を取得
7-1	タイマ/パルス幅変調モード	_CreatePWM	パルス幅変調モードの初期設定
7-2		_EnablePWM	タイマの動作開始、動作停止を制御

No	生成関数	生成関数	内容
7-3		_DestroyPWM	パルス幅変調モードの設定をクリア
7-4		_DestroyPWM_ALL	パルス幅変調モードの設定をクリア
8-1	タイマ/パルス周期測定モード	_CreatePulsePeriodMeasurementMode	パルス周期測定モードの初期設定
8-2		_EnablePulsePeriodMeasurementMode	タイマの動作開始、動作停止を制御
8-3		_DestroyPulsePeriodMeasurementMode	パルス周期測定モードの設定をクリア
8-4		_DestroyPulsePeriodMeasurementMode_ALL	パルス周期測定モードの設定をクリア
8-5		_GetPulsePeriodMeasurementMode	パルス周期測定モードのカウント値を取得
9-1	タイマ/パルス幅測定モード	_CreatePulseWidthMeasurementMode	パルス幅測定モードの初期設定
9-2		_EnablePulseWidthMeasurementMode	タイマの動作制御
9-3		_DestroyPulseWidthMeasurementMode	パルス幅測定モードの設定をクリア
9-4		_DestroyPulseWidthMeasurementMode_ALL	パルス幅測定モードの設定をクリア
9-5		_GetPulseWidthMeasurementMode	パルス幅測定モードカウント値を取得
10-1	タイマ/インプットキャプチャモード	_CreateInputCapture	インプットキャプチャモードの初期設定
10-2		_EnableInputCapture	タイマの動作開始、動作停止を制御
10-3		_DestroyInputCapture	インプットキャプチャモード設定をクリア
10-4		_DestroyInputCapture_ALL	インプットキャプチャモード設定をクリア
10-5		_GetCaptureValue	インプットキャプチャモードのカウント値を取得
11-1	タイマ/アウトプットコンペアモード	_CreateOutputCompare	アウトプットコンペアモードの初期設定
11-2		_EnableOutputCompare	タイマの動作開始、動作停止を制御
11-3		_DestroyOutputCompare	アウトプットコンペアモードの設定をクリア
11-4		_DestroyOutputCompare_ALL	アウトプットコンペアモードの設定をクリア
11-5		_GetTimerFlag	タイマのステータスフラグを取得
11-6		_ClearTimerFlag	タイマのステータスフラグをクリア

各生成関数の仕様を以下に示します。

## ・シリアル

### 1-1 \_\_CreateSCI

生成関数	Boolean __CreateSCI[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	シリアルの初期設定
パラメータ	—
戻り値	シリアル通信の設定に成功した場合は RAPI_TRUE を返し、設定に失敗した場合は RAPI_FALSE を返します。

### 1-2 \_\_DestroySCI

生成関数	Boolean __DestroySCI[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	シリアルポートのクローズ
パラメータ	—
戻り値	RAPI_TRUE を返します。

### 1-3 \_\_StartSCIReceiving

生成関数	Boolean __StartSCIReceiving[モード]_U[資源]_p[設定番号] (unsigned char *data1, unsigned short data2, unsigned short *data3)
周辺機能	シリアル
内容	シリアル受信の開始
パラメータ	data1 : 受信データを格納するバッファへのポインタ data2 : 受信バイト数 data3 : 受信したデータ数を格納するアドレスへのポインタ
戻り値	シリアル受信開始に成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## 1-4 \_\_StartSCISending

生成関数	Boolean __StartSCISending_[モード]_U[資源]_p[設定番号] (unsigned char *data1, unsigned short data2, unsigned short *data3)
周辺機能	シリアル
内容	シリアル送信の開始
パラメータ	data1 : 送信データへのポインタ data2 : 送信バイト数 data3 : 送信したデータ数を格納するアドレスへのポインタ
戻り値	シリアル送信開始に成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## 1-5 \_\_StopSCIReceiving

生成関数	Boolean __StopSCIReceiving_[モード]_U[資源]_p[設定番号] (unsigned short data)
周辺機能	シリアル
内容	シリアル受信の停止
パラメータ	data : シリアル受信停止までのウェイト時間
戻り値	シリアル受信停止に成功し、なおかつ受信エラーが発生していない場合は RAPI_TRUE を返します。それ以外の場合は RAPI_FALSE を返します。

## 1-6 \_\_StopSCISending

生成関数	Boolean __StopSCISending_[モード]_U[資源]_p[設定番号] (unsigned short data)
周辺機能	シリアル
内容	シリアル送信の停止
パラメータ	data : シリアル送信停止までのウェイト時間
戻り値	シリアル送信停止に成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## 1-7 \_\_PollingSCIReceiving

生成関数	Boolean __PollingSCIReceiving_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	ポーリングを用いたシリアル受信
パラメータ	—
戻り値	受信データが間違っている場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 1-8 \_\_PollingSCISending

生成関数	Boolean __PollingSCISending_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	ポーリングを用いたシリアル送信
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 1-9 \_\_GetSCIStatus

生成関数	Boolean __GetSCIStatus_[モード]_U[資源]_p[設定番号] (unsigned long data, unsigned char *status)
周辺機能	シリアル
内容	シリアル通信の各種送受信情報を取得
パラメータ	data : 取得するフラグ (以下のパラメータを指定して下さい。複数のパラメータを選択する場合は、パラメータ間に“ ”記号を入れて下さい。) RAPI_ALL_FLAG : SCI のすべてのフラグを取得 RAPI_ERROR_FLAG : SCI のすべてのエラーフラグを取得 RAPI_TDRE : 送信データエンティフラグを取得 RAPI_RDRF : 受信データフルフラグを取得 RAPI_ORER : オーバランエラーフラグを取得 RAPI_FER : フレーミングエラーフラグを取得 RAPI_PER : パリティエラーフラグを取得 RAPI_TEND : 送信終了フラグを取得 RAPI_MPB : マルチプロセスビットフラグを取得 status : 受信エラーフラグを格納するのに使用するバイトのアドレス
戻り値	RAPI_TRUE を返します。



## 1-10 \_\_ClearSCISStatus

生成関数	Boolean __ClearSCISStatus_[モード]_U[資源]_p[設定番号] (unsigned long data)
周辺機能	シリアル
内容	シリアル通信の各種送受信情報をクリア
パラメータ	data : クリアするフラグ (以下のパラメータを指定して下さい。複数のパラメータを選択する場合は、パラメータ間に“ ”記号を入れて下さい。) RAPI_ALL_FLAG : SCI のすべてのフラグをクリア RAPI_RECV_ERROR : SCI のすべてのエラーフラグおよび受信完了フラグをクリア RAPI_TDRE : 送信データエンプティフラグをクリア RAPI_RDRF : 受信データフルフラグをクリア RAPI_ORER : オーバランエラーフラグをクリア RAPI_FER : フレーミングエラーフラグをクリア RAPI_PER : パリティエラーフラグをクリア RAPI_TEND : 送信終了フラグをクリア RAPI_MPB : マルチプロセッサビットフラグをクリア
戻り値	RAPI_TRUE を返します。

## 1-11 \_\_OutputSCISck

生成関数	Boolean __OutputSCISck_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	調歩同期式モード時に SCK 出力を制御
パラメータ	—
戻り値	RAPI_TRUE を返します。
備考	非同期の場合のみ出力されます。

## 1-12 \_\_OutputSCITxd

生成関数	Boolean __OutputSCITxd_[モード]_U[資源]_p[設定番号] (void)
周辺機能	シリアル
内容	調歩同期式モード時に TXD 出力を制御
パラメータ	なし
戻り値	RAPI_TRUE を返します。
備考	非同期の場合のみ出力されます。

## ・ 割込み

## 2-1 \_\_CreateInterrupt

生成関数	Boolean __CreateInterrupt_I[資源]_p[設定番号] (void)
周辺機能	割り込み
内容	外部割込みの初期設定
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 2-2 \_\_EnableInterrupt

生成関数	Boolean __EnableInterrupt_I[資源]_p[設定番号] (unsigned long data)
周辺機能	割り込み
内容	外部割込みの有効・無効切替
パラメータ	data : 割り込みの動作設定 (以下のパラメータを指定して下さい。複数のパラメータを選択する場合は、パラメータ間に“ ”記号を入れて下さい。) ステータスフラグの設定 RAPI_INT_REQUEST_CLEAR : IRQi 割り込み要求ステータスフラグ (i = 0~3) をクリア (ローレベル検出に設定している場合は無効) RAPI_INT_REQUEST_REMAIN : IRQi 割り込み要求ステータスフラグ (i = 0~3) を保持 (ローレベル検出に設定している場合は無効) 割り込み許可/禁止の設定 RAPI_IRQ_DIS : 割り込みを禁止 RAPI_IRQ_ENA : 割り込みを許可
戻り値	RAPI_TRUE を返します。

## 2-3 \_\_GetInterruptAndPinInfo\_ALL

生成関数	Boolean __GetInterruptAndPinInfo_ALL (unsigned long data1, unsigned char *data2)
周辺機能	割り込み
内容	外部割り込みの入力端子状態と割込要求のステータスを取得
パラメータ	data1 : 取得する外部割り込み入力端子のステータスおよび割り込み要求フラグ (以下のパラメータを指定して下さい。複数のパラメータを選択する場合は、パラメータ間に“ ”記号を入れて下さい。) RAPI_NMI_PIN : NMI の端子レベルを取得 RAPI_IRQ0_FLAG : IRQ0 割り込み要求フラグの値を取得 RAPI_IRQ0_PIN : IRQ0 の端子レベルを取得 RAPI_IRQ1_FLAG : IRQ1 割り込み要求フラグの値を取得 RAPI_IRQ1_PIN : IRQ1 の端子レベルを取得 RAPI_IRQ2_FLAG : IRQ2 割り込み要求フラグの値を取得 RAPI_IRQ2_PIN : IRQ2 の端子レベルを取得 RAPI_IRQ3_FLAG : IRQ3 割り込み要求フラグの値を取得 RAPI_IRQ3_PIN : IRQ3 の端子レベルを取得 data2 : 端子の値および取得したフラグを格納するバッファへのポインタ
戻り値	RAPI_TRUE を返します。

## 2-4 \_\_ClearInterruptFlag

生成関数	Boolean __ClearInterruptFlag_ALL (unsigned long data)
周辺機能	割り込み
内容	外部割り込みの入力端子状態と割込要求のステータスをクリア
パラメータ	data1 : クリアする割り込み要求フラグ (以下のパラメータを指定して下さい。複数のパラメータを選択する場合は、パラメータ間に“ ”記号を入れて下さい。) RAPI_IRQ0_FLAG : IRQ0 割り込み要求フラグ RAPI_IRQ1_FLAG : IRQ1 割り込み要求フラグ RAPI_IRQ2_FLAG : IRQ2 割り込み要求フラグ RAPI_IRQ3_FLAG : IRQ3 割り込み要求フラグ
戻り値	RAPI_TRUE を返します。

## ・ A/D 変換

## 3-1 \_\_CreateADC

生成関数	Boolean __CreateADC_[モード]_RAPI_[資源]_p[設定番号] (void)
周辺機能	A/D 変換
内容	A/D 変換器の初期設定
パラメータ	—
戻り値	A/D コンバータの設定に成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## 3-2 \_\_EnableADC

生成関数	Boolean __EnableADC_RAPI_[モジュール]_ALL (unsigned long data)
周辺機能	A/D 変換
内容	A/D 変換器の動作開始、動作停止を制御
パラメータ	data : 変換器の動作開始、動作停止の設定 (以下のパラメータを指定して下さい。) RAPI_AD_ON : A/D コンバータ動作開始 RAPI_AD_OFF : A/D コンバータ動作停止
戻り値	A/D コンバータの制御に成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## 3-3 \_\_DestroyADC

生成関数	Boolean __DestroyADC_RAPI_[モジュール]_ALL (void)
周辺機能	A/D 変換
内容	A/D 変換器の設定をクリア
パラメータ	—
戻り値	A/D コンバータの設定クリアに成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## 3-4 \_\_GetADC

生成関数	Boolean __GetADC_[モード]_RAPI_[資源]_p[設定番号] (unsigned long data1, unsigned short* data2)
周辺機能	A/D 変換
内容	A/D 変換結果を取得
パラメータ	data1 : 変換結果を取得する A/D データレジスタ (以下のパラメータを指定して下さい。複数のパラメータを選択する場合は、パラメータ間に“ ”記号を入れて下さい。) RAPI_ADDR0 : A/D データレジスタ 0 RAPI_ADDR1 : A/D データレジスタ 1 RAPI_ADDR2 : A/D データレジスタ 2 RAPI_ADDR3 : A/D データレジスタ 3 RAPI_ADDR4 : A/D データレジスタ 4 RAPI_ADDR5 : A/D データレジスタ 5 RAPI_ADDR6 : A/D データレジスタ 6 RAPI_ADDR7 : A/D データレジスタ 7 RAPI_ADDR_ALL : A/D データレジスタ 0~7 のすべての値を取得 data2 : A/D 変換値を格納するバッファへのポインタ (A/D 変換後、A/D データレジスタには変換値が左詰めで格納されますが、バッファには右詰めで格納されます。)
戻り値	A/D 変換結果の取得に成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## 3-5 \_\_GetADCFlag

生成関数	Boolean __GetADCFlag_RAPI_[モジュール]_ALL (unsigned char* status)
周辺機能	A/D 変換
内容	A/D 変換器のステータスフラグを取得
パラメータ	status : A/D コンバータのステータスを示す A/D エンドフラグを格納するバッファへのポインタ
戻り値	A/D コンバータのステータス取得に成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## 3-6 \_\_ClearADCFlag

生成関数	Boolean __ClearADCFlag_RAPI_[モジュール]_ALL (void)
周辺機能	A/D 変換
内容	A/D 変換器のステータスフラグをクリア
パラメータ	—
戻り値	A/D コンバータのステータスクリアに成功した場合は RAPI_TRUE を返し、失敗した場合は RAPI_FALSE を返します。

## ・I/O ポート

## 4-1 \_\_SetIOPort

生成関数	Boolean __SetIOPort_p[設定番号]_[ポート] (void)
周辺機能	I/O ポート
内容	I/O ポートの動作条件設定
パラメータ	—
戻り値	RAPI_TRUE を返します。

## 4-2 \_\_ReadIOPort\_ALL

生成関数	Boolean __ReadIOPort_ALL (unsigned long data1, void *data2)
周辺機能	I/O ポート
内容	指定した I/O ポートのデータ取得
パラメータ	SH/Tiny 版 Renesas Embedded Application Programming Interface ユーザーズマニュアル の __ReadIOPort の項を参照してください。
戻り値	I/O ポートの指定が正しくない場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 4-3 \_\_WriteIOPort\_ALL

生成関数	Boolean __WriteIOPort_ALL (unsigned long data1, unsigned short data2)
周辺機能	I/O ポート
内容	指定した I/O ポートへデータ書き込み
パラメータ	SH/Tiny 版 Renesas Embedded Application Programming Interface ユーザーズマニュアル の __WriteIOPort の項を参照してください。
戻り値	I/O ポートの指定が正しくない場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## ・タイマ/タイマモード

## 5-1 \_\_CreateTimer

生成関数	Boolean __CreateTimer_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/タイマモード
内容	タイマの初期設定
パラメータ	—
戻り値	タイマの初期設定に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 5-2 \_\_EnableTimer

生成関数	Boolean __EnableTimer_Tch[資源]_p[設定番号] (unsigned long data)
周辺機能	タイマ/タイマモード
内容	タイマの動作開始、動作停止を制御
パラメータ	data : タイマの動作開始、動作停止の設定 (以下のパラメータを指定して下さい) RAPI_TIMER_ON : タイマモードに設定されているタイマを動作開始に設定 RAPI_TIMER_OFF : タイマモードに設定されているタイマを動作停止に設定
戻り値	タイマの制御に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 5-3 \_\_DestroyTimer

生成関数	Boolean __DestroyTimer_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/タイマモード
内容	タイマモードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 5-4 \_\_DestroyTimer\_ALL

生成関数	Boolean __DestroyTimer_ALL (void)
周辺機能	タイマ/タイマモード
内容	タイマモードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 5-5 \_\_GetTimerCounter

生成関数	Boolean __GetTimerCounter_Tch[資源]_p[設定番号] (unsigned short *data)
周辺機能	タイマ/タイマモード
内容	タイマのカウント値を取得
パラメータ	data : カウント値を格納するバッファへのポインタ
戻り値	タイマのカウント値取得に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## ・タイマ/イベントカウンタモード

## 6-1 \_\_CreateEventCounter

生成関数	Boolean __CreateEventCounter_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/イベントカウンタモード
内容	イベントカウンタモードの初期設定
パラメータ	—
戻り値	タイマの初期設定に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 6-2 \_\_EnableEventCounter

生成関数	Boolean __EnableEventCounter_Tch[資源]_p[設定番号] (unsigned long data)
周辺機能	タイマ/イベントカウンタモード
内容	タイマの動作開始、動作停止を制御
パラメータ	data : タイマの動作開始、動作停止の設定 (以下のパラメータを指定して下さい) RAPI_TIMER_ON : イベントカウンタモードに設定されているタイマを動作開始に設定 RAPI_TIMER_OFF : イベントカウンタモードに設定されているタイマを動作停止に設定
戻り値	タイマの制御に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 6-3 \_\_DestroyEventCounter

生成関数	Boolean __DestroyEventCounter_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/イベントカウンタモード
内容	イベントカウンタモードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 6-4 \_\_DestroyEventCounter\_ALL

生成関数	Boolean __DestroyEventCounter_ALL (void)
周辺機能	タイマ/イベントカウンタモード
内容	イベントカウンタモードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 6-5 \_\_GetTimerCounter

生成関数	Boolean __GetTimerCounter_Tch[資源]_p[設定番号] (unsigned short *data)
周辺機能	タイマ/イベントカウンタモード
内容	タイマのカウンタ値を取得
パラメータ	data : カウンタ値を格納するバッファへのポインタ
戻り値	タイマのカウンタ値取得に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## ・タイマ / パルス幅変調モード

## 7-1 \_\_CreatePWM

生成関数	Boolean __CreatePWM_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅変調モード
内容	パルス幅変調モードの初期設定
パラメータ	—
戻り値	タイマの初期設定に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 7-2 \_\_EnablePWM

生成関数	Boolean __EnablePWM_Tch[資源]_p[設定番号] (unsigned long data)
周辺機能	タイマ/パルス幅変調モード
内容	タイマの動作開始、動作停止を制御
パラメータ	data : タイマの動作開始、動作停止の設定 (以下のパラメータを指定して下さい) RAPI_TIMER_ON : タイマを動作開始に設定 RAPI_TIMER_OFF : タイマを動作停止に設定
戻り値	タイマの制御に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 7-3 \_\_DestroyPWM

生成関数	Boolean __DestroyPWM_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅変調モード
内容	パルス幅変調モードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 7-4 \_\_DestroyPWM\_ALL

生成関数	Boolean __DestroyPWM_ALL (void)
周辺機能	タイマ/パルス幅変調モード
内容	パルス幅変調モードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## ・ タイマ / パルス周期測定モード

## 8-1 \_\_CreatePulsePeriodMeasurementMode

生成関数	Boolean __CreatePulsePeriodMeasurementMode_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス周期測定モード
内容	パルス周期測定モードの初期設定
パラメータ	—
戻り値	タイマの初期設定に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 8-2 \_\_EnablePulsePeriodMeasurementMode

生成関数	Boolean __EnablePulsePeriodMeasurementMode_Tch[資源]_p[設定番号] (unsigned long data)
周辺機能	タイマ/パルス周期測定モード
内容	タイマの動作開始、動作停止を制御
パラメータ	data : タイマの動作開始、動作停止の設定 (以下のパラメータを指定して下さい。) RAPTIMER_ON : タイマを動作開始に設定 RAPTIMER_OFF : タイマを動作停止に設定
戻り値	タイマの制御に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 8-3 \_\_DestroyPulsePeriodMeasurementMode

生成関数	Boolean __DestroyPulsePeriodMeasurementMode_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス周期測定モード
内容	パルス周期測定モードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 8-4 \_\_DestroyPulsePeriodMeasurementMode\_ALL

生成関数	Boolean __DestroyPulsePeriodMeasurementMode_ALL (void)
周辺機能	タイマ/パルス周期測定モード
内容	パルス周期測定モードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 8-5 \_\_GetPulsePeriodMeasurementMode

生成関数	Boolean __GetPulsePeriodMeasurementMode_Tch[資源]_p[設定番号] (unsigned short *data)
周辺機能	タイマ/パルス周期測定モード
内容	パルス周期測定モードに設定されているタイマのカウンタ値を取得
パラメータ	data : カウンタ値を格納するバッファへのポインタ
戻り値	タイマのカウンタ値取得に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## ・ タイマ / パルス幅測定モード

## 9-1 \_\_CreatePulseWidthMeasurementMode

生成関数	Boolean __CreatePulseWidthMeasurementMode_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅測定モード
内容	パルス幅測定モードの初期設定
パラメータ	—
戻り値	タイマの初期設定に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 9-2 \_\_EnablePulseWidthMeasurementMode

生成関数	Boolean __EnablePulseWidthMeasurementMode_Tch[資源]_p[設定番号] (unsigned long data)
周辺機能	タイマ/パルス幅測定モード
内容	タイマの動作制御
パラメータ	data : タイマの動作開始、動作停止の設定 (以下のパラメータを指定して下さい。) RAPTIMER_ON : タイマを動作開始に設定 RAPTIMER_OFF : タイマを動作停止に設定
戻り値	タイマの制御に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 9-3 \_\_DestroyPulseWidthMeasurementMode

生成関数	Boolean __DestroyPulseWidthMeasurementMode_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/パルス幅測定モード
内容	パルス幅測定モードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 9-4 \_\_DestroyPulseWidthMeasurementMode\_ALL

生成関数	Boolean __DestroyPulseWidthMeasurementMode_ALL (void)
周辺機能	タイマ/パルス幅測定モード
内容	パルス幅測定モードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 9-5 \_\_GetPulseWidthMeasurementMode

生成関数	Boolean __GetPulseWidthMeasurementMode_Tch[資源]_p[設定番号] (unsigned short *data)
周辺機能	タイマ/パルス幅測定モード
内容	パルス幅測定モードに設定されているタイマのカウント値を取得
パラメータ	data : カウンタ値を格納するバッファへのポインタ
戻り値	タイマのカウント値取得に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## ・タイマ / インพุットキャプチャモード

## 10-1 \_\_CreateInputCapture

生成関数	Boolean __CreateInputCapture_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/インพุットキャプチャモード
内容	端子をインพุットキャプチャに設定
パラメータ	—
戻り値	タイマの初期設定に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 10-2 \_\_EnableInputCapture

生成関数	Boolean __EnableInputCapture_Tch[資源]_p[設定番号] (unsigned long data)
周辺機能	タイマ/インพุットキャプチャモード
内容	タイマの動作開始、動作停止を制御
パラメータ	data : タイマの動作開始、動作停止の設定 (以下のパラメータを指定して下さい) RAPI_TIMER_ON : タイマを動作開始に設定 RAPI_TIMER_OFF : タイマを動作停止に設定
戻り値	タイマの制御に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 10-3 \_\_DestroyInputCapture

生成関数	Boolean __DestroyInputCapture_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/インพุットキャプチャモード
内容	インพุットキャプチャモードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 10-4 \_\_DestroyInputCapture\_ALL

生成関数	Boolean __DestroyInputCapture_ALL (void)
周辺機能	タイマ/インพุットキャプチャモード
内容	インพุットキャプチャモードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 10-5 \_\_GetCaptureValue

生成関数	Boolean __GetCaptureValue_Tch[資源]_p[設定番号] (unsigned short *data)
周辺機能	タイマ/インพุットキャプチャモード
内容	インพุットキャプチャモードに設定されているタイマのカウント値を取得
パラメータ	data : カウンタ値を格納するバッファへのポインタ
戻り値	タイマのカウント値取得に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。



## ・タイマ / アウトプットコンペアモード

## 11-1 \_\_CreateOutputCompare

生成関数	Boolean __CreateOutputCompare_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/アウトプットコンペアモード
内容	アウトプットコンペアモードの初期設定
パラメータ	—
戻り値	タイマの初期設定に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 11-2 \_\_EnableOutputCompare

生成関数	Boolean __EnableOutputCompare_Tch[資源]_p[設定番号] (unsigned long data)
周辺機能	タイマ/アウトプットコンペアモード
内容	タイマの動作開始、動作停止を制御
パラメータ	data : タイマの動作開始、動作停止の設定 (以下のパラメータを指定して下さい。) RAPTIMER_ON : タイマを動作開始に設定 RAPTIMER_OFF : タイマを動作停止に設定
戻り値	タイマの制御に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 11-3 \_\_DestroyOutputCompare

生成関数	Boolean __DestroyOutputCompare_Tch[資源]_p[設定番号] (void)
周辺機能	タイマ/アウトプットコンペアモード
内容	アウトプットコンペアモードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 11-4 \_\_DestroyOutputCompare\_ALL

生成関数	Boolean __DestroyOutputCompare_ALL (void)
周辺機能	タイマ/アウトプットコンペアモード
内容	アウトプットコンペアモードに設定されているタイマの設定をクリア
パラメータ	—
戻り値	タイマ設定のクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 11-5 \_\_GetTimerFlag

生成関数	Boolean __GetTimerFlag_Tch[資源]_p[設定番号] (nsigned long data1, unsigned char *data2)
周辺機能	タイマ/アウトプットコンペアモード
内容	タイマのステータスフラグを取得
パラメータ	data1 : 取得するステータスフラグ (以下のパラメータを指定して下さい。複数のパラメータを選択する場合は、パラメータ間に“ ”記号を入れて下さい。) RAPTGF A : MTU2 インพุットキャプチャ/アウトプットコンペアフラグ A (TGFA) RAPTGF B : MTU2 インพุットキャプチャ/アウトプットコンペアフラグ B (TGFB) RAPTGF C : MTU2 インพุットキャプチャ/アウトプットコンペアフラグ C (TGFC) RAPTGF D : MTU2 インพุットキャプチャ/アウトプットコンペアフラグ D (TGFD) RAPTCF D : MTU2 カウント方向フラグ (TCFD) RAPTCF V : MTU2 オーバフローフラグ (TCFV) RAPTCF U : MTU2 アンダフローフラグ (TCFU) RAPTGF E : MTU2 コンペアマッチフラグ E (TGFE) RAPTGF F : MTU2 コンペアマッチフラグ F (TGFF) RAPICMF : CMT コンペアマッチフラグ (CMF) data2 : タイマステータスフラグを格納するバッファへのポインタ
戻り値	ステータス取得に失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 11-6 \_\_ClearTimerFlag

生成関数	Boolean __ClearTimerFlag_Tch[資源]_p[設定番号] (unsigned long data)
周辺機能	タイマ/アウトプットコンペアモード
内容	タイマのステータスフラグをクリア
パラメータ	<p>data1 : クリアするステータスフラグ</p> <p>(以下のパラメータを指定して下さい。複数のパラメータを選択する場合は、パラメータ間に“ ”記号を入れて下さい。)</p> <p>RAPI_TGFA : MTU2 インットキャプチャ/アウトプットコンペアフラグ A(TGFA)</p> <p>RAPI_TGFB : MTU2 インットキャプチャ/アウトプットコンペアフラグ B(TGFB)</p> <p>RAPI_TGFC : MTU2 インットキャプチャ/アウトプットコンペアフラグ C(TGFC)</p> <p>RAPI_TGFD : MTU2 インットキャプチャ/アウトプットコンペアフラグ D(TGFD)</p> <p>RAPI_TCFD : MTU2 カウント方向フラグ(TCFD)</p> <p>RAPI_TCFV : MTU2 オーバフローフラグ(TCFV)</p> <p>RAPI_TCFU : MTU2 アンダフローフラグ(TCFU)</p> <p>RAPI_TGFE : MTU2 コンペアマッチフラグ E(TGFE)</p> <p>RAPI_TGFF : MTU2 コンペアマッチフラグ F(TGFF)</p> <p>RAPI_CMF : CMT コンペアマッチフラグ(CMF)</p>
戻り値	ステータスクリアに失敗した場合は RAPI_FALSE を返し、それ以外の場合は RAPI_TRUE を返します。

## 6. プロジェクトコンバート

### 6.1 コンバート機能

ある機種で作成したプロジェクト(設定内容)を、別の機種で使用できるように変換します。

コンバート元プロジェクトの設定値がコンバート先プロジェクトで使用できない場合は、コンバート先の機種に応じて設定内容を変更します。変更方法については、次節以降を参照してください。

SH7125 から他の機種へのコンバートおよびSH7125 以外の機種からSH7125 へのコンバートはできません。

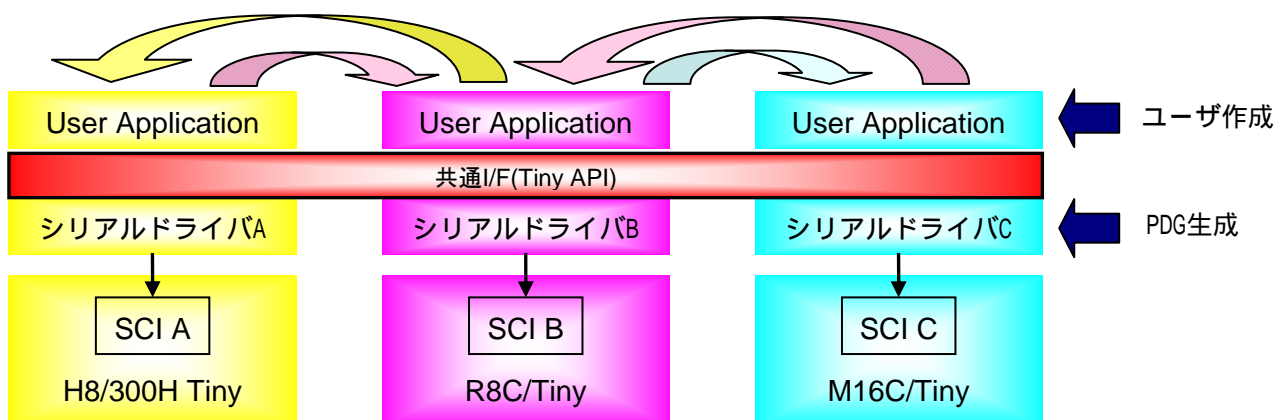


図 6.1 コンバートダイアログボックス

### 6.2 コンバートによる設定内容の変更と表示

設定内容の変更方法には、次の2通りがあります。

- i. 設定値を変更する、又は、新しい値を設定する
  - コンバート元での設定値がコンバート先では使用できない場合
  - コンバート元では項目が無効であり、コンバート先では新しい値を設定する必要がある場合
- ii. 設定項目自体を無効にする
  - コンバート先の機種が、項目をサポートしていない場合

資源の設定

資源の設定は、すべて削除します。

設定値の表示

変換の結果を表 4-1 に示すアイコンで示します。

表 6.1 変換結果の表示

アイコン	説明	項目番号
	コンバート元の設定値をそのまま使用した	-
	プログラムが設定値を変更した	.
	コンバートによって項目自体が無効になった	.
	コンバート元の設定値をそのまま使用した (コンバート元でもコンバート先でも項目自体が無効)	-

### 6.3 プロジェクトコンバートの操作方法

「ファイル」 - 「プロジェクトコンバート」メニューを選択して、「コンバート」ダイアログボックスを開きます。  
コンバート元プロジェクト名、コンバートして新規に作成するプロジェクト名とその保存ディレクトリを入力します。  
コンバート(変換)したいCPU種別のシリーズ・グループ・型名をそれぞれ選択肢から指定し、「OK」ボタンを選択します。

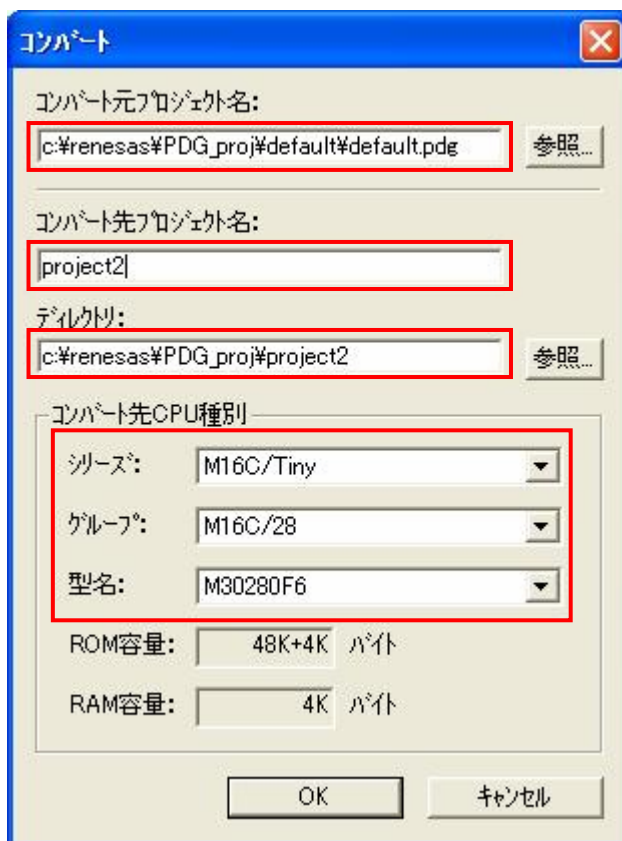


図 6.2 コンバートダイアログボックス

指定したディレクトリに新しくプロジェクトファイルが作られます。コンバートが完了すると、メッセージを表示して知らせます。

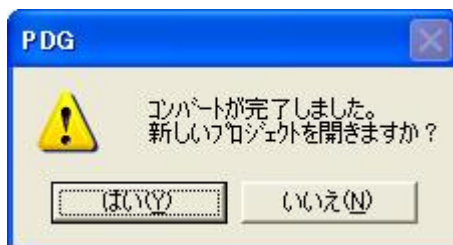


図 6.3 コンバート完了メッセージ(PDG)

「はい」を選択すると、作成したプロジェクトファイルを開きます。  
コンバート前の CPU 及び設定内容により、コンバート後に設定が無効になる項目、または再設定が

必要になる項目が発生する場合があります。使用している周辺 I/O モジュールの各設定パターン表示ウィンドウを順次開き、内容を確認してください。

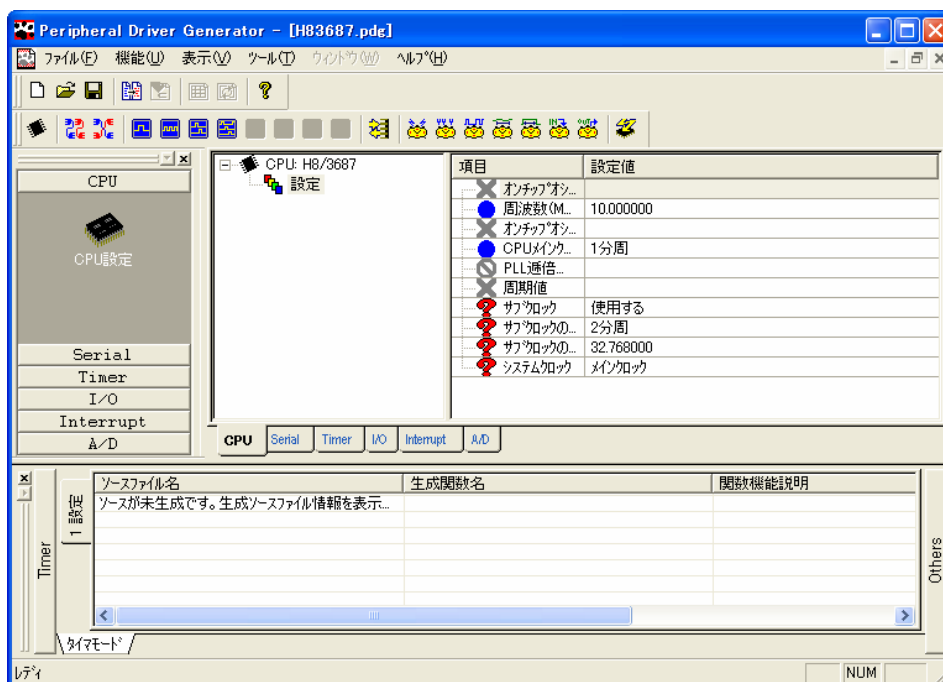




図 6.4 コンバート後のプロジェクト表示例

❓は、コンバート後の CPU の仕様が異なる等により再度設定が必要もしくは確認が必要な項目を示しますので、必要に応じて、設定パターンを変更してください。

必要な変更を行うと、マークが、マークに変わります。

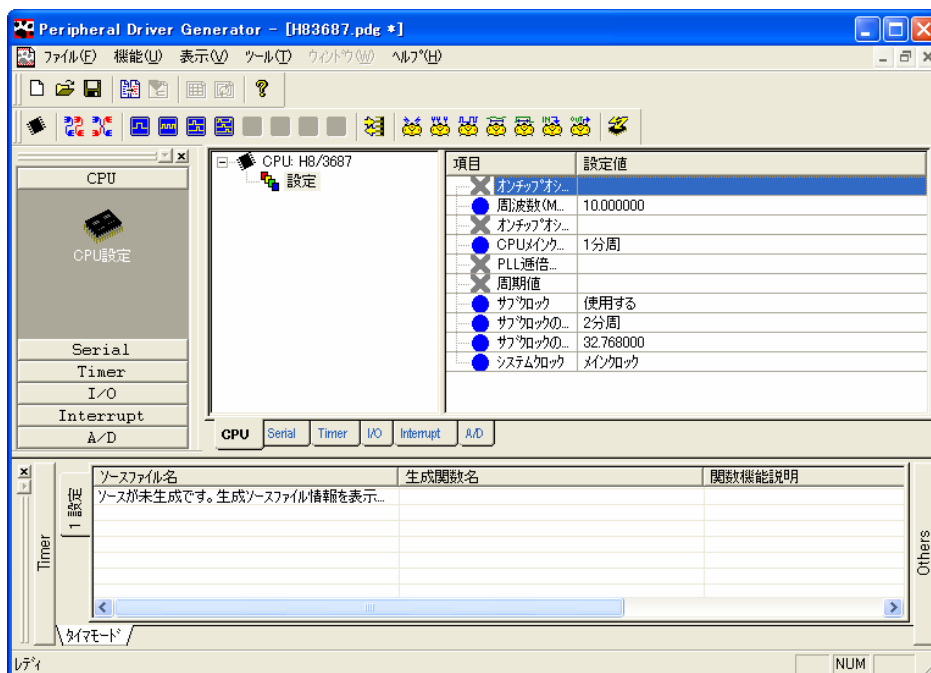


図 6.5 コンバート後のプロジェクト表示例

---

Peripheral Driver Generator V.1.03  
ユーザーズマニュアル

発行年月日 2008年9月19日 Rev.1.00

発行 株式会社 ルネサス テクノロジ 営業統括部  
〒100-0004 東京都千代田区大手町2-6-2

編集 株式会社 ルネサス ソリューションズ ツール開発部



# Peripheral Driver Generator V.1.03 ユーザーズマニュアル



ルネサスエレクトロニクス株式会社  
神奈川県川崎市中原区下沼部1753 〒211-8668

RJJ10J2348-0100