

RX210 グループ

IRQ 割り込みを使用したパルス出力

要旨

本サンプルコードでは、IRQ 割り込みが発生すると、一定期間タイマでパルスを出力する方法について説明します。

対象デバイス

- RX210

内容

1.	仕様	3
2.	動作確認条件	3
3.	ハードウェア説明.....	3
3.1	使用端子一覧.....	3
4.	ソフトウェア説明.....	4
4.1	動作概要	4
4.2	ファイル構成.....	5
4.3	オプション設定メモリ.....	6
4.4	定数一覧	6
4.5	変数一覧	6
4.6	関数一覧	7
4.7	関数仕様	7
4.8	作成する関数のフローチャート.....	8
4.8.1	初期設定.....	8
4.8.2	メイン処理.....	8
4.8.3	IRQ0 割り込み処理	9
4.8.4	TGIA1 割り込み処理	10
5.	PDG の設定	11
5.1	SYSTEM 設定	13
5.2	ICUb 設定	14
5.3	MTU2a 設定	14
5.4	SYSTEM の端子設定	16
5.5	ソースの生成.....	17
5.6	CS+への登録.....	18
6.	CS+のプロジェクトに PDG のソースファイルを登録する際の設定.....	22
7.	参考ドキュメント.....	25

1. 仕様

IRQ の立ち下がりエッジを割り込み要因として、周波数 2kHz のデューティ比 20%のパルスを 1 秒間出力します。

2. 動作確認条件

本サンプルコードは、表 2.1 の条件で動作を確認しています。

表 2.1 動作確認条件

項目	内容
使用マイコン	R5F5210BBDFP (RX210 グループ)
動作周波数	・メインクロック : 20MHz ・動作周波数 : 20MHz
ボード電源電圧	5V
マイコン動作電圧	5V
エンディアン	リトルエンディアン
動作モード	シングルチップモード
プロセッサモード	スーパバイザモード
統合開発環境	ルネサスエレクトロニクス製品 CS+ for CC-RL V5.00.00
エミュレータ	ルネサスエレクトロニクス製 E1 エミュレータ
使用ボード	北斗電子製評価ボード HSBRX210-100B (R5F5210BBDFP)

3. ハードウェア説明

3.1 使用端子一覧

表 3.1 に使用端子と機能を示します。

表 3.1 使用端子と機能

端子名	入出力	内容
PH1	入力	SW2 (IRQ0)
P20	出力	MTIOC1A

4. ソフトウェア説明

4.1 動作概要

IRQ とマルチファンクションタイマパルスユニット (MTU2) を使用して、SW2 を押す(立ち下がりエッジを入力する) と、MTIOC1A 端子から 1 秒間設定したパルスを出力し、停止します。パルス出力を停止中、SW2 を押す(立ち下がりエッジを入力する) と、MTIOC1A 端子から 1 秒間設定したパルスを出力し、停止します。

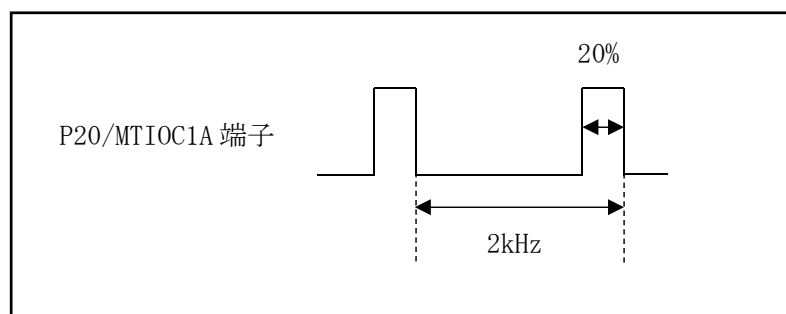
<IRQ>

使用ボードでは SW2 に接続している PH1 端子がプルアップされており、SW2 を押していない状態で High レベルが入力されます。SW2 を押すと GND に接続され、SW2 に接続されている PH1 端子に Low レベルが入力されます。PH1 端子の状態が High レベルのときに SW2 が押されている、Low レベルのときに SW2 が押されていないことが判定できます。

PH1 端子は SW2 を押していない状態 (High レベル) から SW2 を押すことにより (Low レベル)、立ち下がりエッジを入力することができます。PH1 端子を IRQ0 端子機能に設定し、立ち下がりエッジを検出すると、割り込みを発生させることができます。

<MTU2 チャンネル 1>

MTU2 チャンネル 1 で P20 端子を MTIOC1A 端子機能に設定し、周波数 2kHz のデューティ比 20% のパルスを出力させます。



4.2 ファイル構成

本アプリケーションを作成するにあたり、編集したファイルを表 4.1 に示します。(統合開発環境で自動生成され、編集していないファイルについては割愛します)

表 4.1 ファイル名一覧

ファイル名	概要	備考
IRQ_PulseOut_RX210.c	メインファイル <ul style="list-style-type: none"> IRQ0 割り込み処理 TGIA1 割り込み処理 オプション設定メモリ 	
hwsetup.c	初期設定 <ul style="list-style-type: none"> 存在しない端子の処理 クロックの設定 ポートの設定 IRQ0 の設定 MTU1 の設定 	
resetprg.c	リセット例外処理	HardwareSetup(); のコメントアウトを解除しました

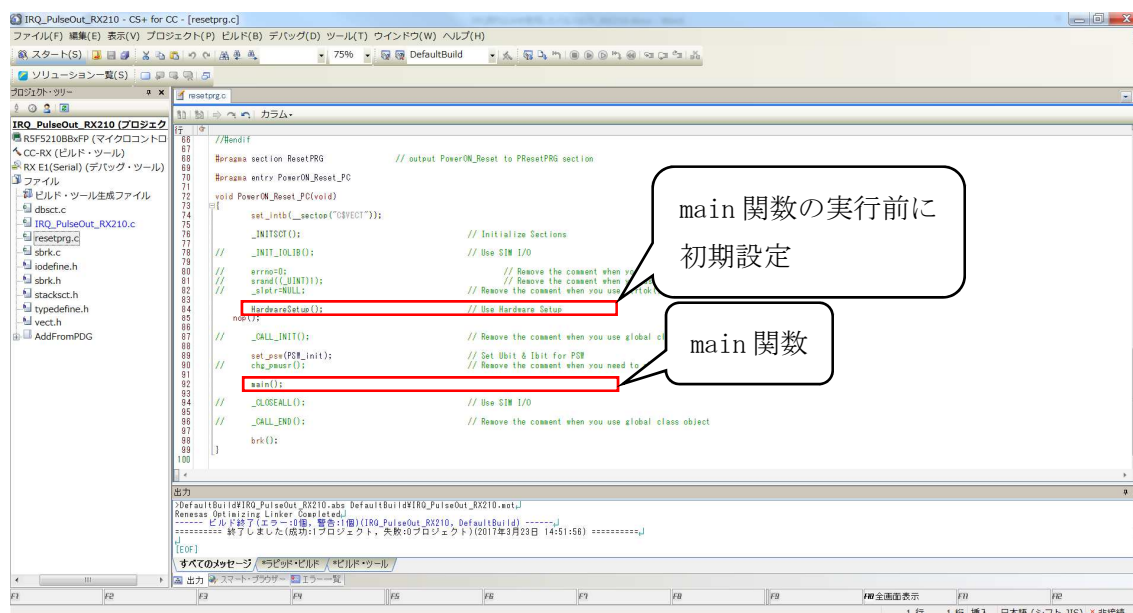


図 4.1 resetprg.c

4.3 オプション設定メモリ

表 4.2 に本サンプルコードで使用するオプション設定メモリの状態を示します。

表 4.2 オプション設定メモリー一覧

シンボル	アドレス	設定値	内容
OFS0	FFFF FF8Fh～FFFF FF8Ch	FFFF FFFFh	リセット後、IWDT は停止 リセット後、WDT は停止
OFS1	FFFF FF8Bh～FFFF FF88h	FFFF FFFFh	リセット後、 電圧監視 0 リセット無効 H0C0(高速オンチップオシレー タ)発振が無効
MDES	FFFF FF83h～FFFF FF80h	FFFF FFFFh	リトルエンディアン

OFS0 と OFS1 はメインファイルの最後尾に記載しています。

MDES については vecttbl.c ファイル(プロジェクト作成時に自動生成されるファイル)に定義されています。

4.4 定数一覧

表 4.3 に本サンプルコードで使用する定数を示します。

表 4.3 サンプルコードで使用する定数

定数名	設定値	内容
SET_PULSE_CNT	1999UL	出力するパルス数-1 の設定

4.5 変数一覧

表 4.4 に本サンプルコードで使用する変数を示します。

表 4.4 サンプルコードで使用する変数

型	変数名	内容	使用関数
static unsigned short	tgial_cnt	出力するパルスカウンタ	main Irq0IntFunc Mtu1IcCmAIntFunc

4.6 関数一覧

表 4.5 に関数一覧を掲載します。本サンプルコードで新規作成、もしくは編集した関数のみ記載しています。PDG の設定は 5. PDG の設定を参照ください。

表 4.5 関数一覧

関数名	概要
main	メイン処理
Irq0IntFunc	IRQ0 割り込み処理
Mt1IcCmAIntFunc	TGIA1 割り込み処理

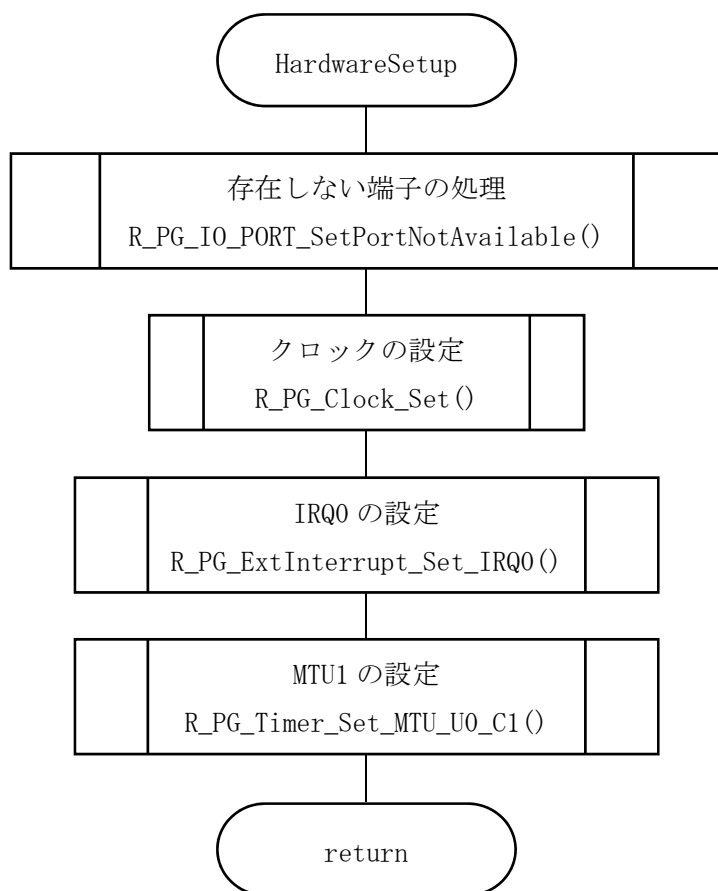
4.7 関数仕様

本サンプルコードで作成、もしくは編集した関数仕様を示します。

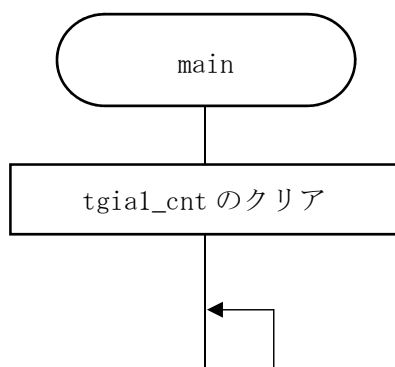
main	
概要	メイン処理
ヘッダ	なし
宣言	void main(void)
説明	割り込みカウンタ
引数	なし
リターン値	なし
Irq0IntFunc	
概要	IRQ0 割り込み処理
ヘッダ	なし
宣言	void Irq0IntFunc (void)
説明	パルス出力(MTU2 チャンネル 1)の開始
引数	なし
リターン値	なし
Mt1IcCmAIntFunc	
概要	TGIA1 割り込み処理
ヘッダ	なし
宣言	void Mt1IcCmAIntFunc (void)
説明	パルス出力の監視、および停止
引数	なし
リターン値	なし

4.8 作成する関数のフローチャート

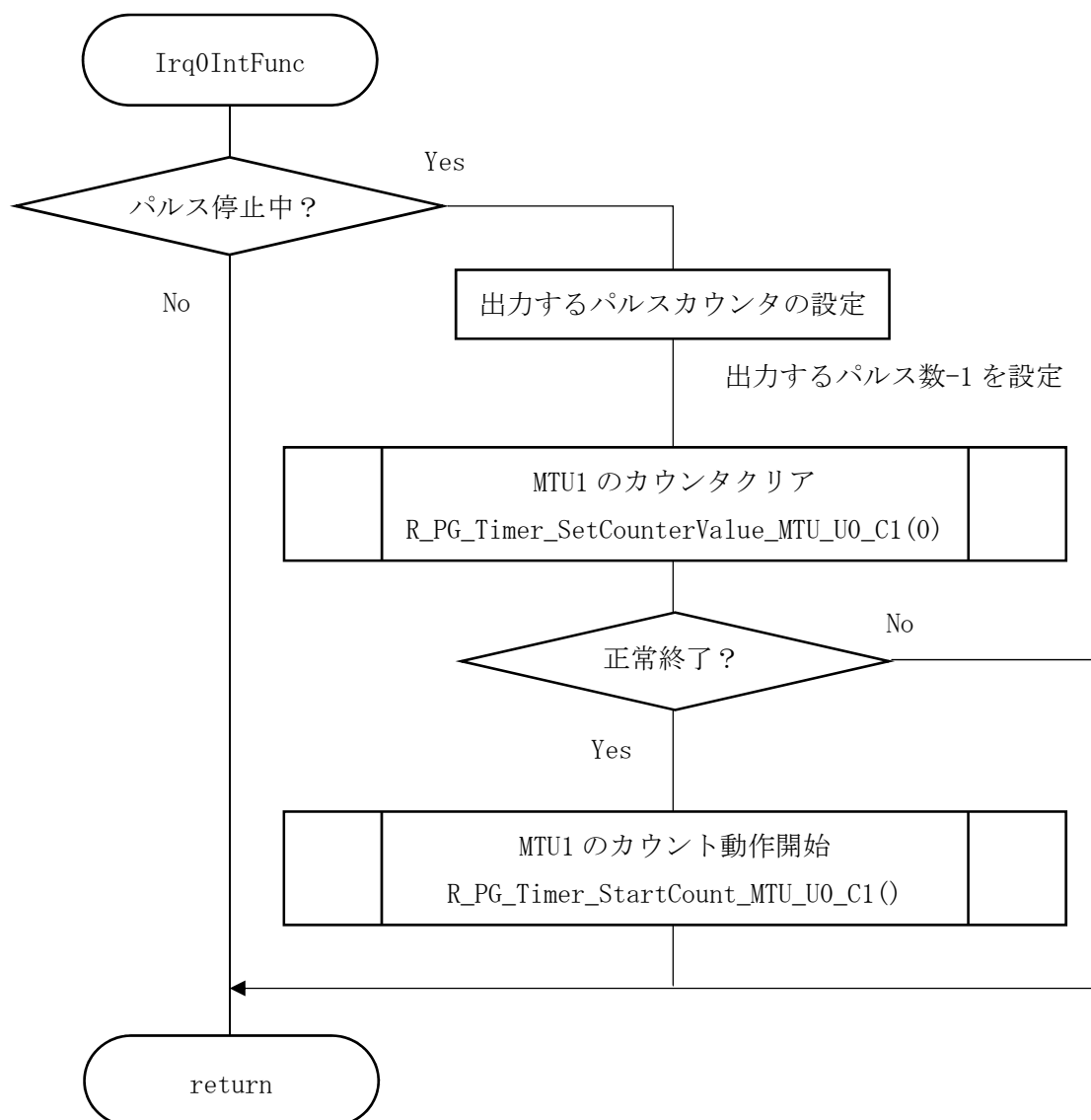
4.8.1 初期設定



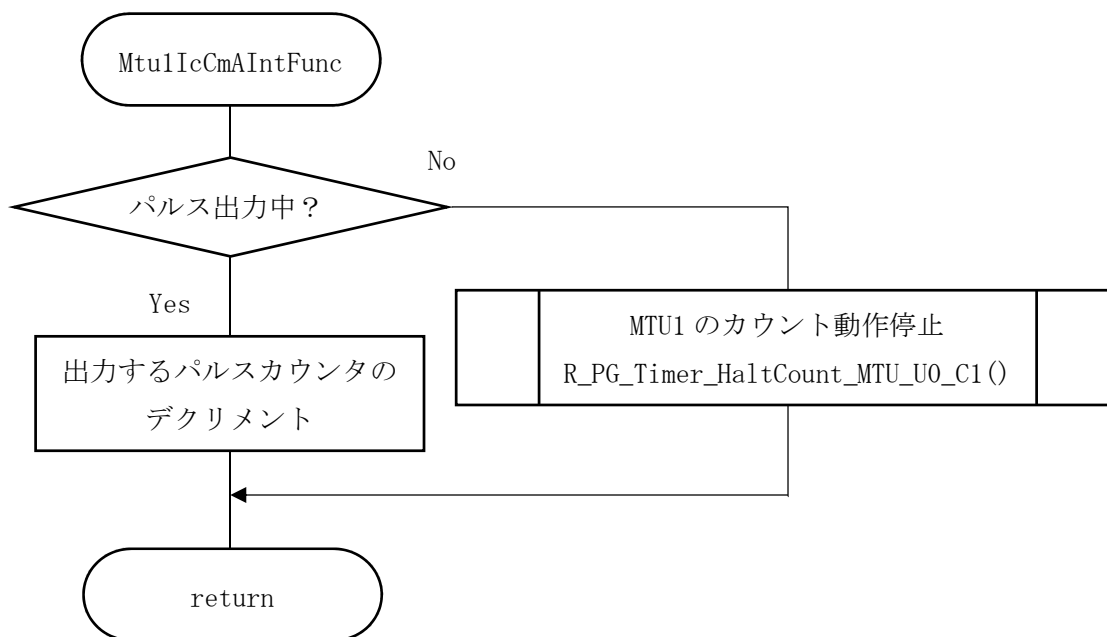
4.8.2 メイン処理



4.8.3 IRQ0 割り込み処理



4.8.4 TGIA1 割り込み処理



5. PDG の設定

本サンプルコードにおける PDG の設定を以下に説明します。本設定において生成されるソースファイルの詳細は”RX210 グループ Peripheral Driver Generator リファレンスマニュアル”を参照ください。

Peripheral Driver Generator 2 を起動します。



メニューバーのファイル→プロジェクトの新規作成 をクリックすると、以下のウィンドウが表示されます。プロジェクト名、マイコンのグループ、型を入力し、「OK」をクリックすると、プロジェクトが作成されます。

新規作成

プロジェクト名:
IRQ_PulseOut_RX210

ディレクト ?
c:\renesas\PDG2_proj 参照...

デバイス選択

シリーズ: RX200

グループ: RX210

型: R5F5210BBxFP

パッケージ: PLQP0100KB-A

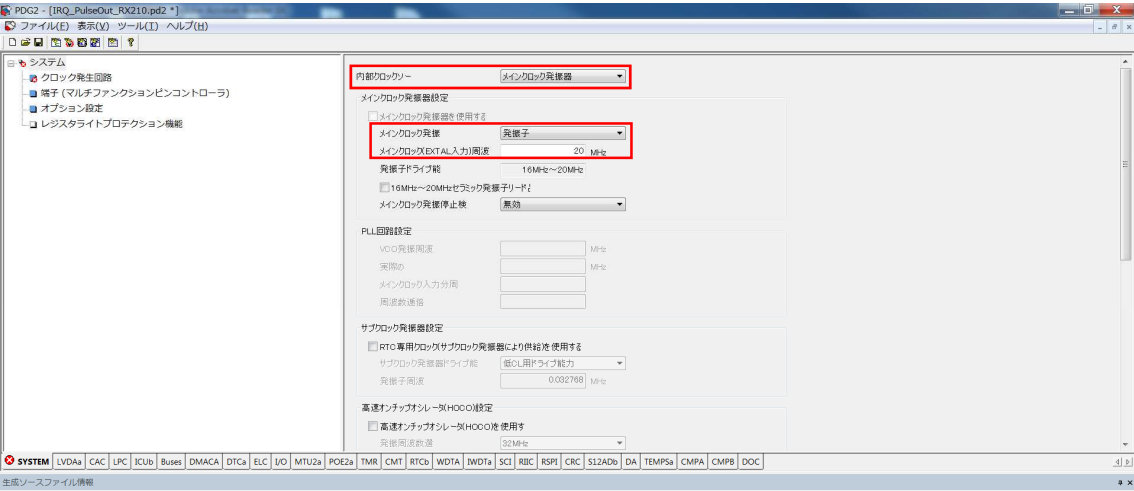
ROM容 1M バイト

RAM容 96K バイト

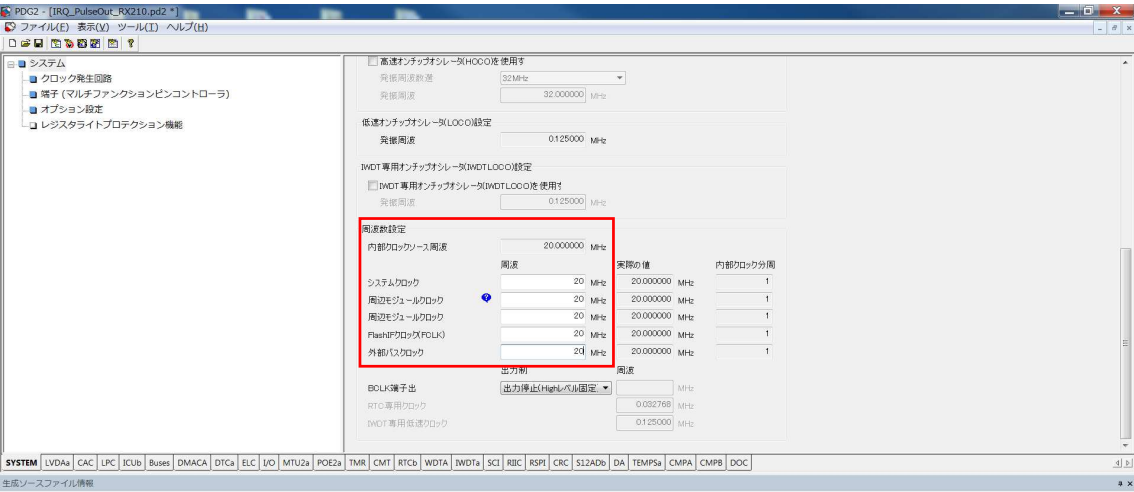
クリック OK キャンセル

5.1 SYSTEM 設定

システムタブのクロック発生回路の設定を以下に示します。



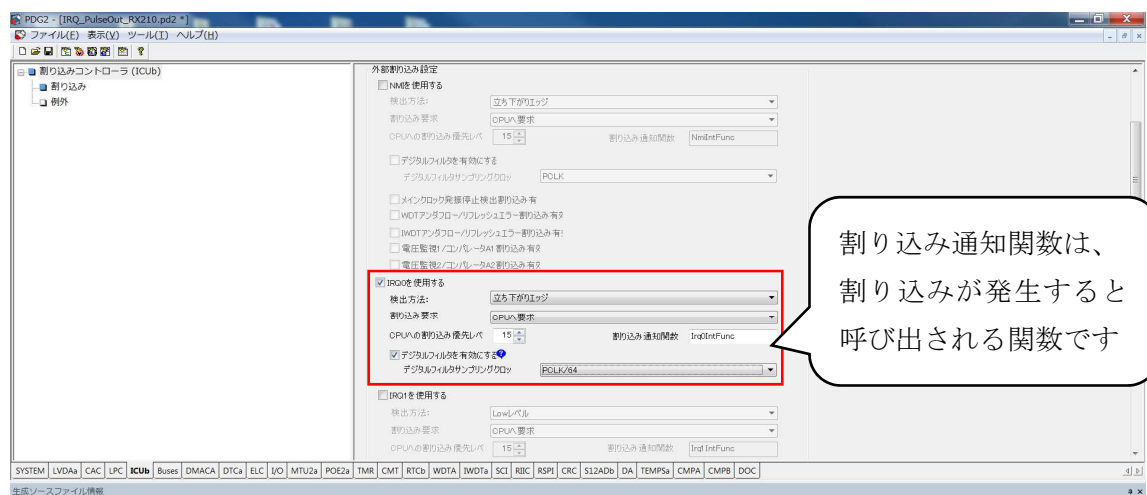
レディ CAP NUM SCRL



レディ CAP NUM SCRL

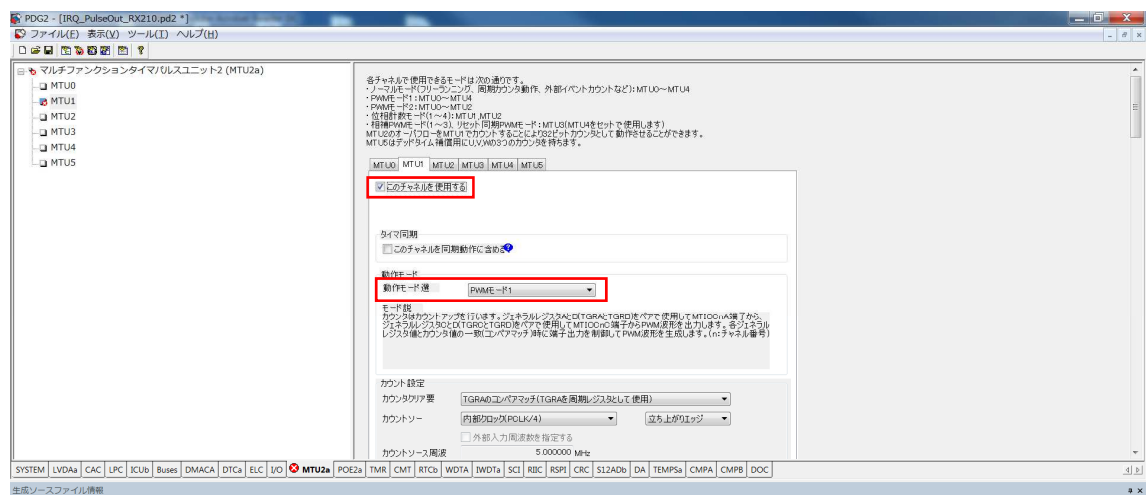
5.2 ICub 設定

ICub の設定を以下に示します。



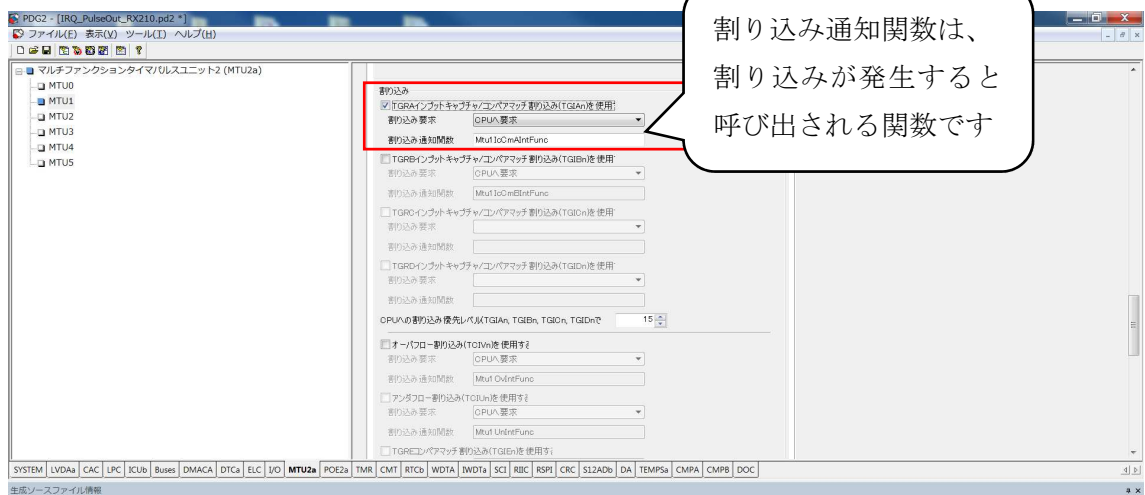
5.3 MTU2a 設定

MTU2a の設定を以下に示します。





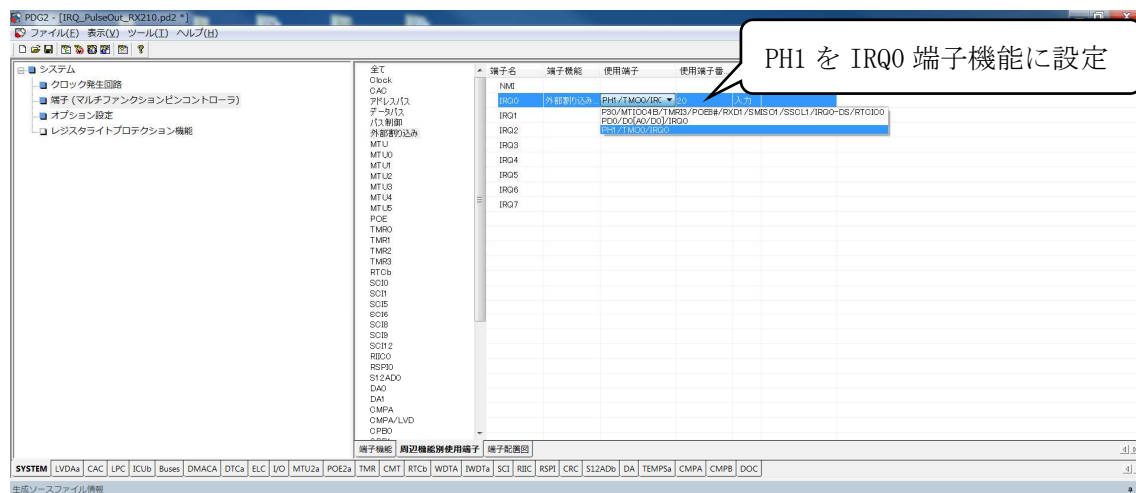
レディ CAP [NUM] SCL



レディ CAP [NUM] SCL

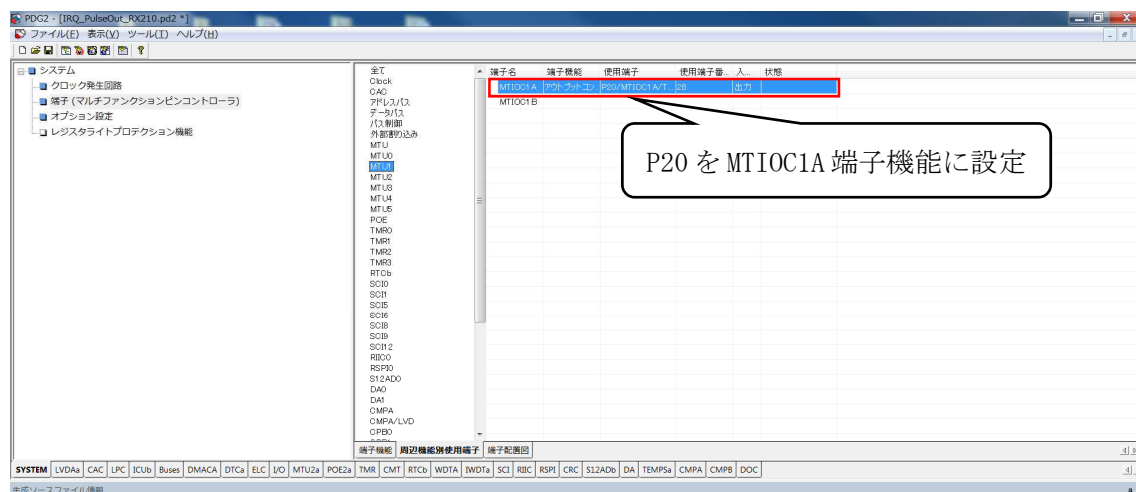
5.4 SYSTEM の端子設定

SYSTEM の端子設定を以下に示します。周辺機能別使用端子タブの外部割込みで設定します。



レディ CAP NUM SCRL

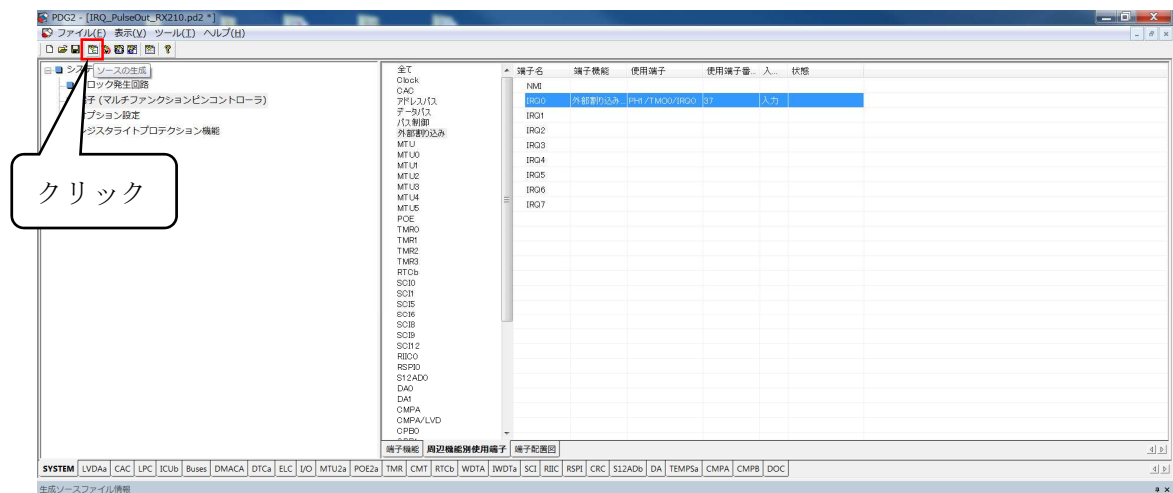
周辺機能別使用端子タブの MTU1 で設定します。



レディ CAP NUM SCRL

5.5 ソースの生成

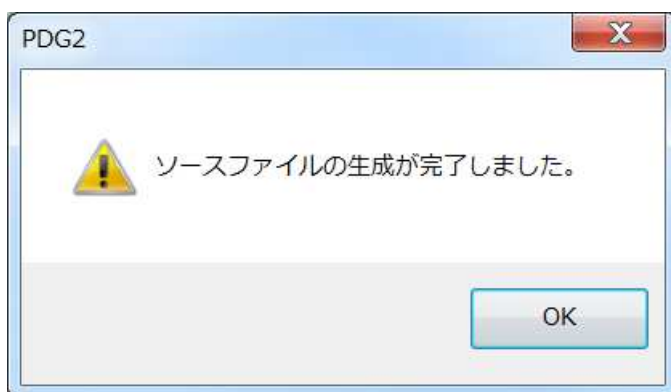
以下の GUI をクリックすると、



ソース生成を実行します

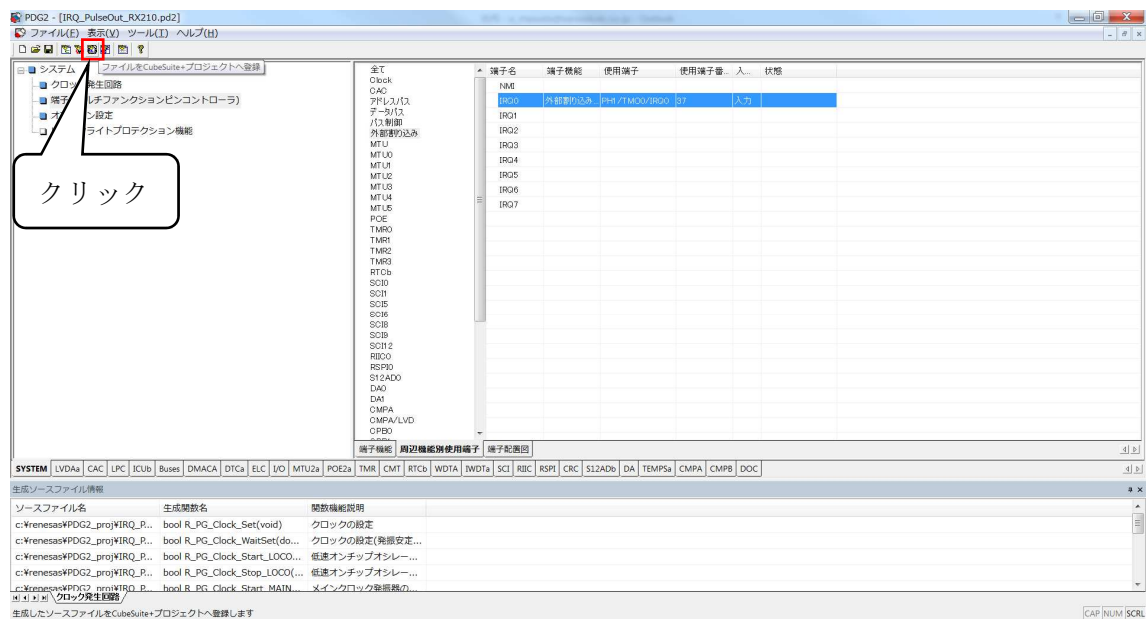
CAP NUM SCRL

ソースファイルが生成されます。



5.6 CS+への登録

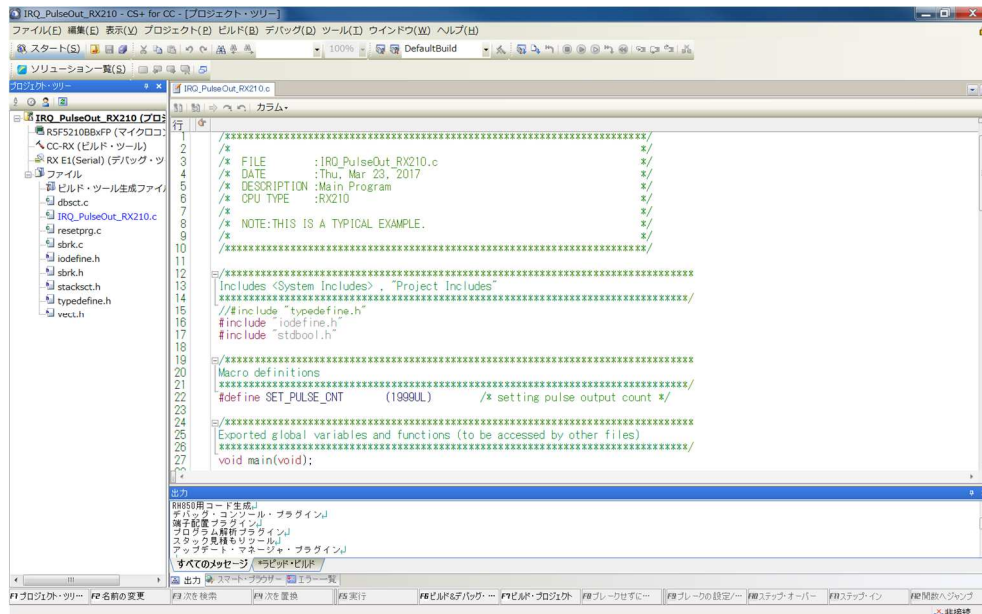
以下の GUI をクリックすると、



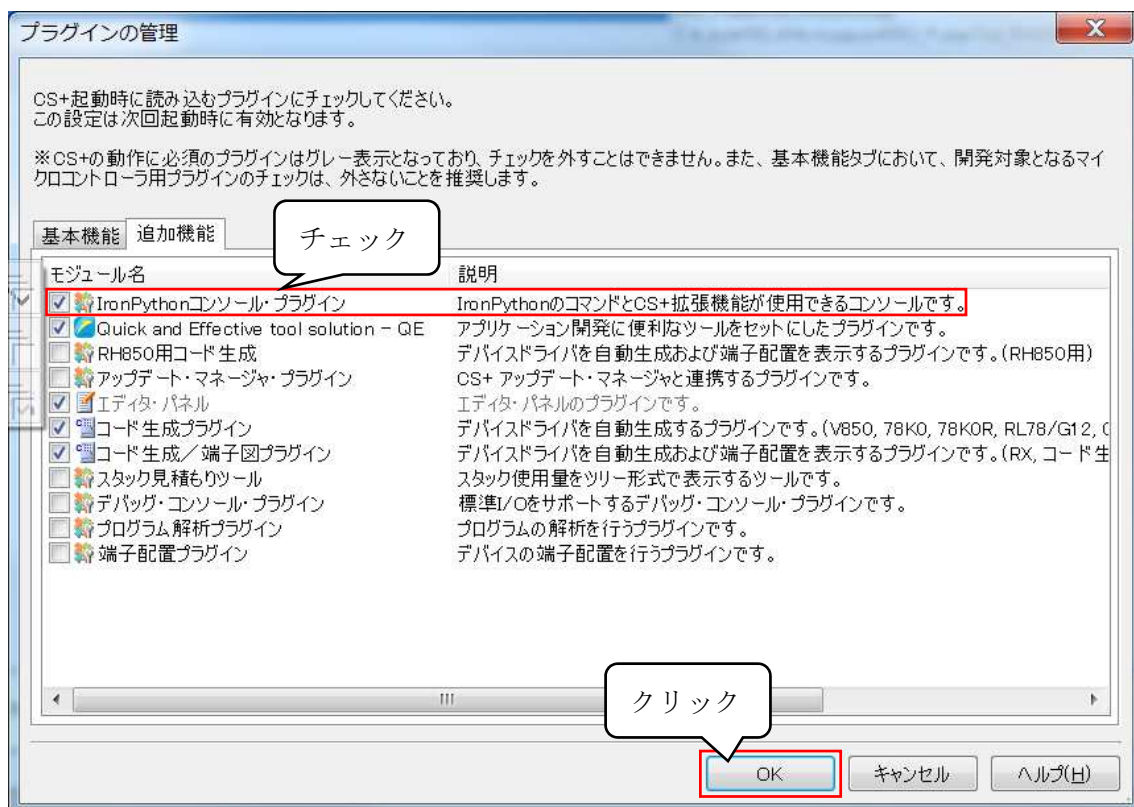
以下のウィンドウが表示されます。



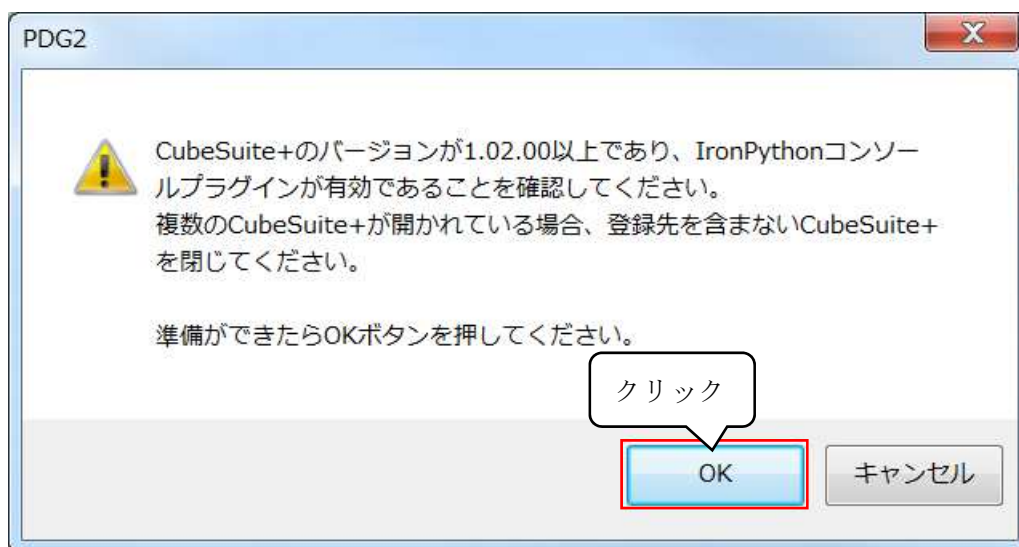
対象のCS+プロジェクトを開きます。



IronPython コンソールプラグインが有効かどうかを確認するため、メニューバーのツール -> プラグイン管理 -> ビルド・ツール -> プロパティをクリックすると、以下のウィンドウが表示され、「IronPython コンソール・プラグイン」をチェック後に「OK」をクリックします。



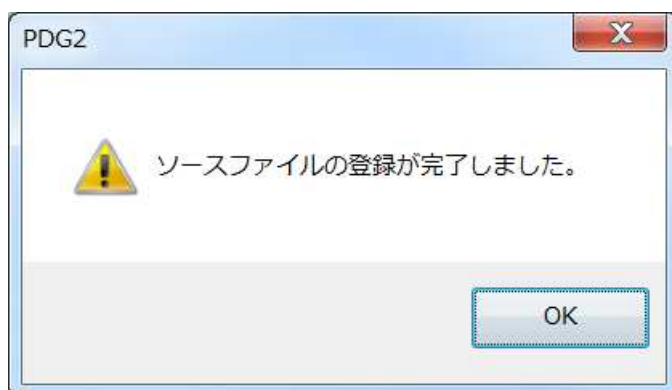
PDG2に戻り、「OK」をクリックします。



ライブラリリンク優先順位設定のウィンドウが表示されます。1つのみであるため、「OK」をクリックします。

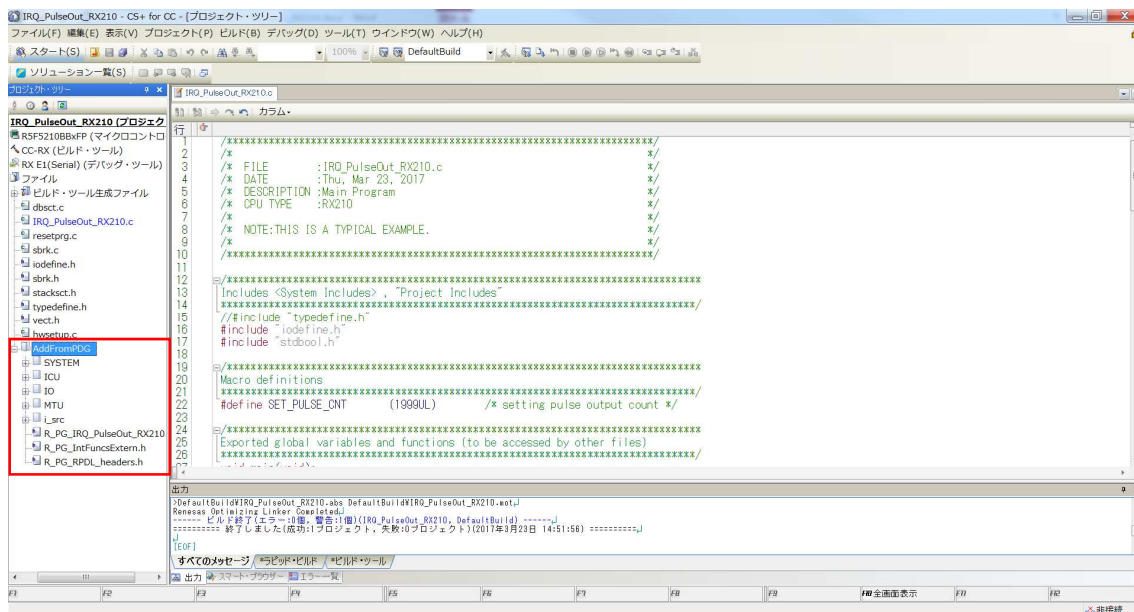


ソースファイルの登録が完了しました。



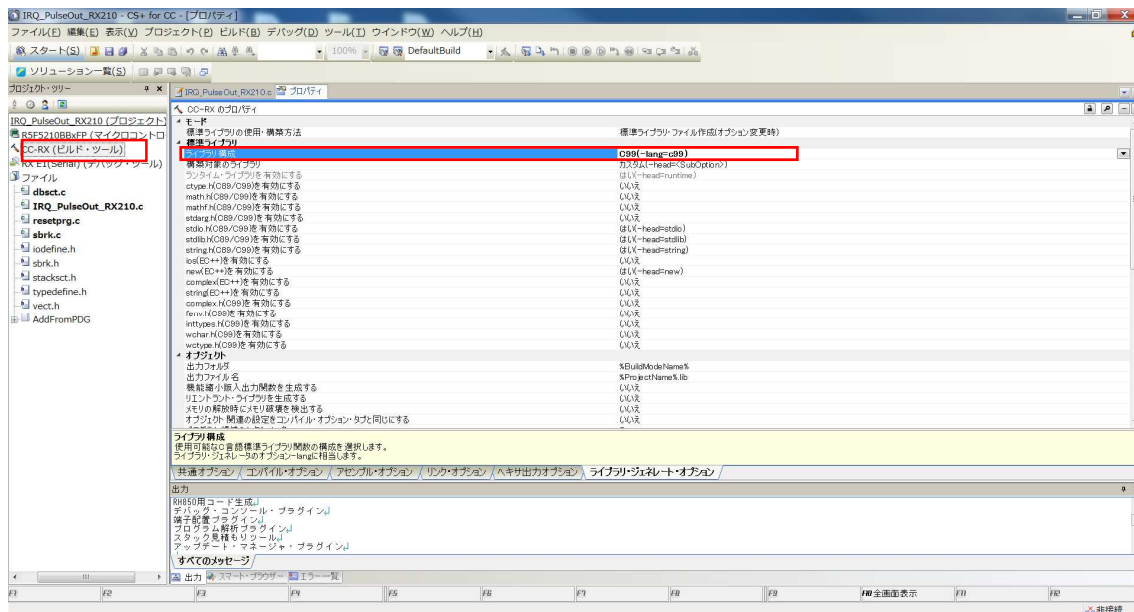
6. CS+のプロジェクトに PDG のソースファイルを登録する際の設定

CS+のプロジェクトに PDG で生成されたソースファイルを登録すると、プロジェクトのファイルに AddFromPDG フォルダが追加されます。

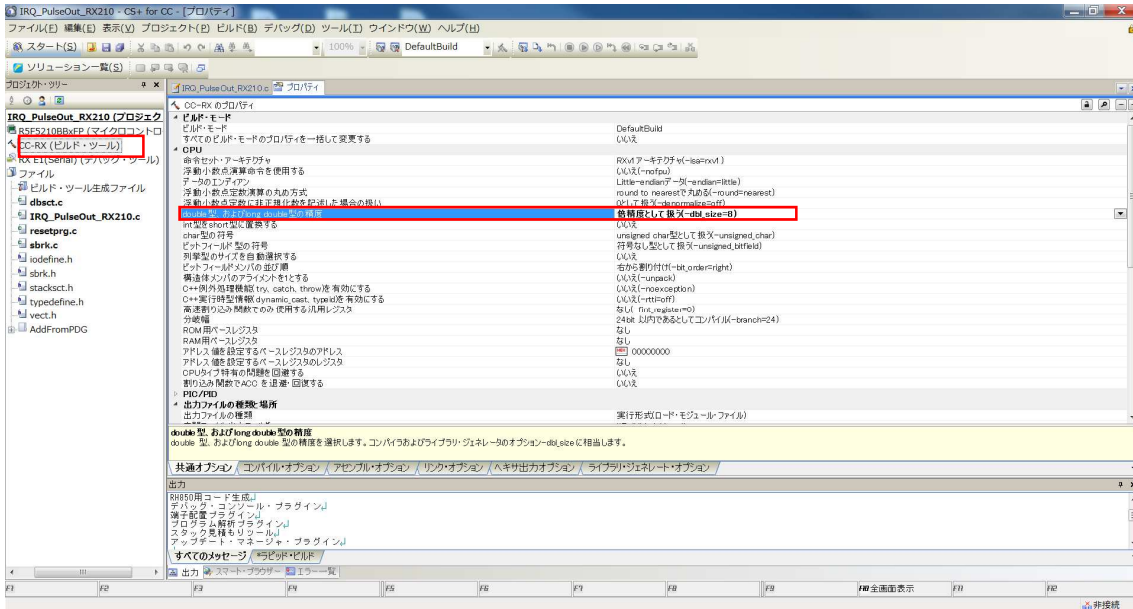


そのままビルドをすると、エラーおよび警告が発生します。解消する設定を以下に示します。

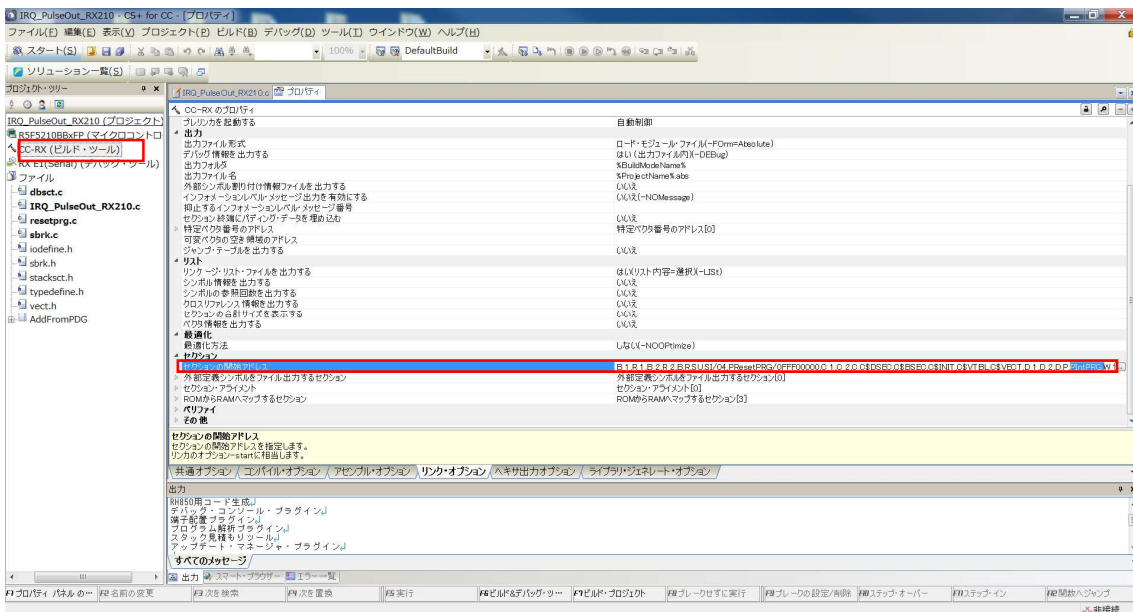
PDG で生成されるソースファイルは `bool` 変数を使用しています。対応させるため、ビルド・ツールを右クリック→プロパティを表示し、ライブラリ・ジェネレート・オプションタブにある「ライブラリ構成」を”C99(-lang=c99)”に設定します。



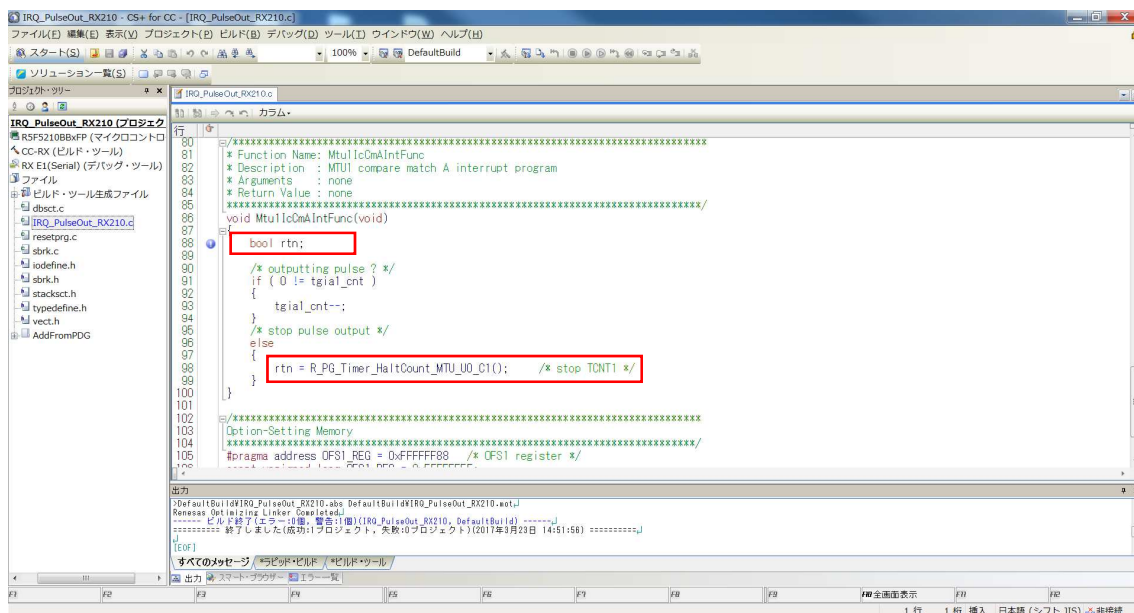
PDG で生成されるソースファイルは double 型、および long double 型の精度を倍精度として扱っているため、ビルド・ツールを右クリック→プロパティを表示し、共通オプションタブにある「double 型、および long double 型の精度」を” 倍精度として扱う(-dbl_size=8)” に設定します。



PDG で生成されるソースファイルを登録すると PIntPRG セクションを使用しないため、CS+プロジェクトを生成した際にデフォルトで設定されている PIntPRG セクションを削除します。ビルド・ツールを右クリック→プロパティを表示し、リンクオプションタブにある「セクションの開始アドレス」から” PIntPRG” を削除します。



本サンプルコードでは、ビルド時に警告が1つ発生します。こちらはPDGで生成された関数の戻り値を使用していない旨の警告です。PDGで生成された関数の実行に失敗することを想定していないため、そのままにしています。御了承ください。



7. 参考ドキュメント

RX63N グループ ユーザーズマニュアルハードウェア編

RX210 グループ Peripheral Driver Generator リファレンスマニュアル

以上