

1. 質問

メイン関数をCコードにて作成する場合、どのようにしてSTMを組み込んで動作させるのでしょうか？

2. 回答

メイン関数をCコードにて作成した場合は、STM初期化関数とSTMコール関数をコールして、STMを動作させます。

3 . STM コール手順

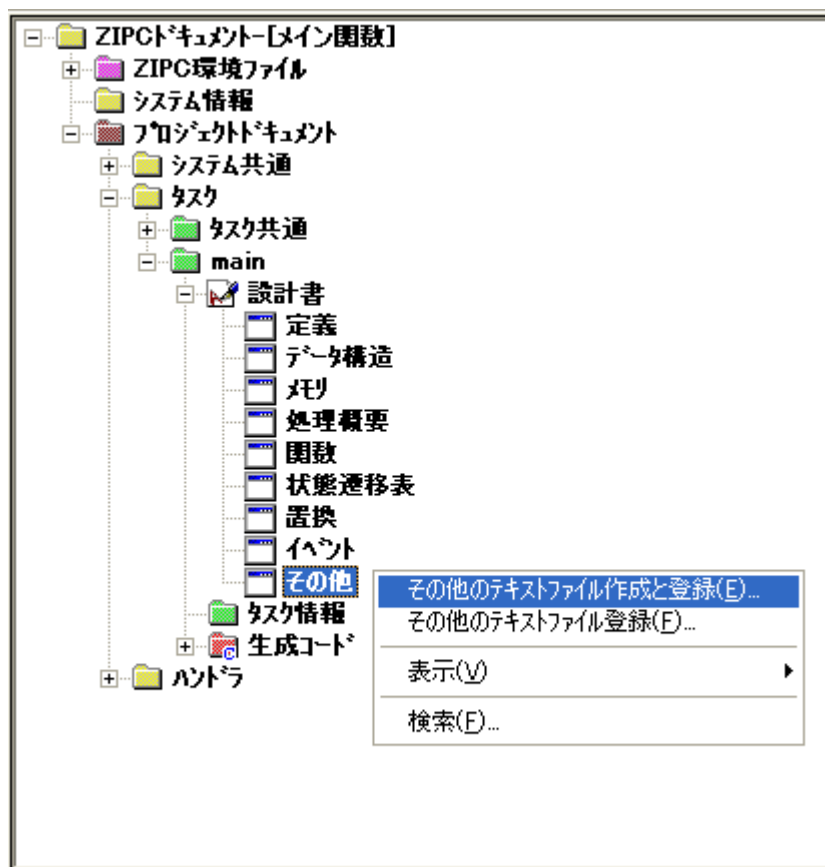
サンプルを基にして、STM 初期化関数 および STM コール関数の設定方法を説明します。

3.1 設計書の登録と作成

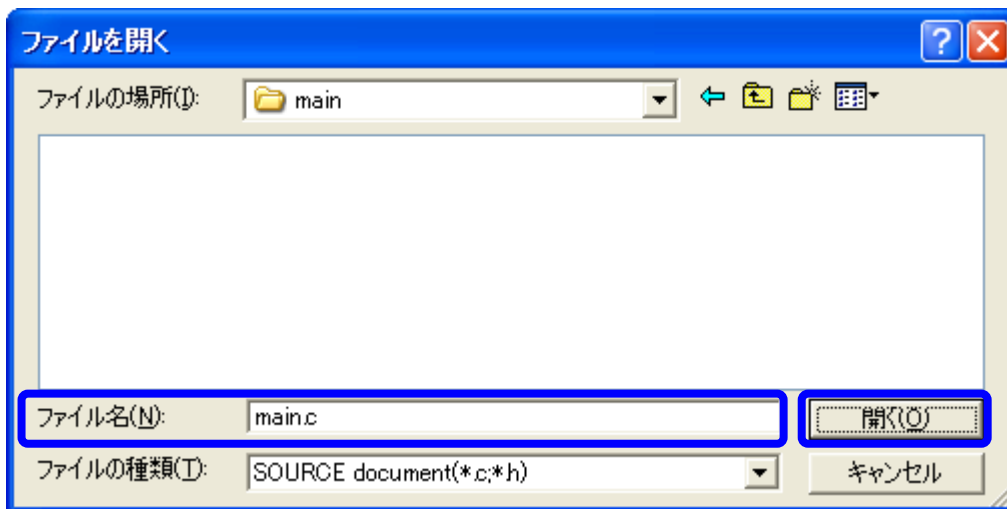
最初に、各設計書を登録し、作成します。

手順

1. ZIPC プロジェクトファイル (メイン関数.zpf) を新規作成します。
2. "main" タスクを登録します。
3. ドキュメントツリーにて、[タスク] - [main] - [設計書] - [その他] 項目を選択します。
4. ドキュメントツリーにて、ポップアップメニューを表示します。
5. ポップアップメニューにて、[その他のテキストファイル作成と登録] を選択します。



6. [ファイルを開く] ダイアログにて、[ファイル名] 項目に "main.c" を指定し、[開く] ボタンを選択します。



7. "main.c" ファイルを、下記に示すように編集します。
"main.c" ファイル

```
/* ZIPC Include File */
#include "main_inc.h"

void main( void )
{
    /* STM を初期化する */
    StmInit();

    while(1){
        /* STM をコールする */
        StmCall();
    }
}
```

備考：

StmInit 関数は STM 初期化関数になり、StmCall は STM コール関数になります。
STM 初期化関数 および STM コール関数の設定方法は、後程説明します。

8. "main" タスクに、"main" STM 設計書 (main.stm) を登録します。
9. "main" STM 設計書の属性を、下記のように設定します。

"main" STM 設計書属性

No	項目	値
1	STM 種別	E 型通常
2	リターン型	void
3	STM 名称	main
4	レベル番号	1 1
5	引数	void

1 : "main" STM は、C コードからコールするので、ルート STM になるように設定します。

The screenshot shows the 'STM設定' dialog box with the following settings:

- タイトル: STM設定
- 属性: トリガアクティビティ登録, メイン開始アクティビティ, メイン終了アクティビティ, コメント, デイスパッチアクティビティ登録, 開始アクティビティ登録, 終了アクティビティ登録
- タイトル: STM設定
- STM種別: E型通常
- リターン型: void
- STM名称: main
- レベル番号: 1 (highlighted)
- 状態管理: []
- 加減管理: []
- 引数: void
- リターン値: []
- リターン値デフォルト設定: []
- Buttons: OK, キャンセル

10. "main" STM 設計書を、編集します。
11. "main" タスクに、"main" RAM 設計書を登録し、編集します。

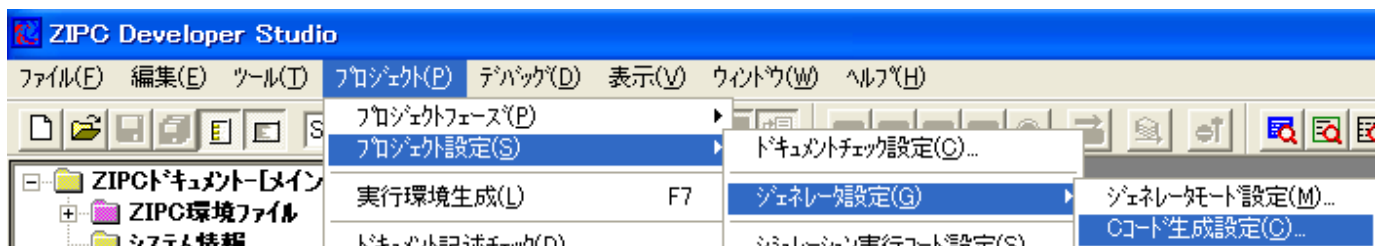
これで、必要な設計書は登録・作成しました。

3.2 STM 関数設定

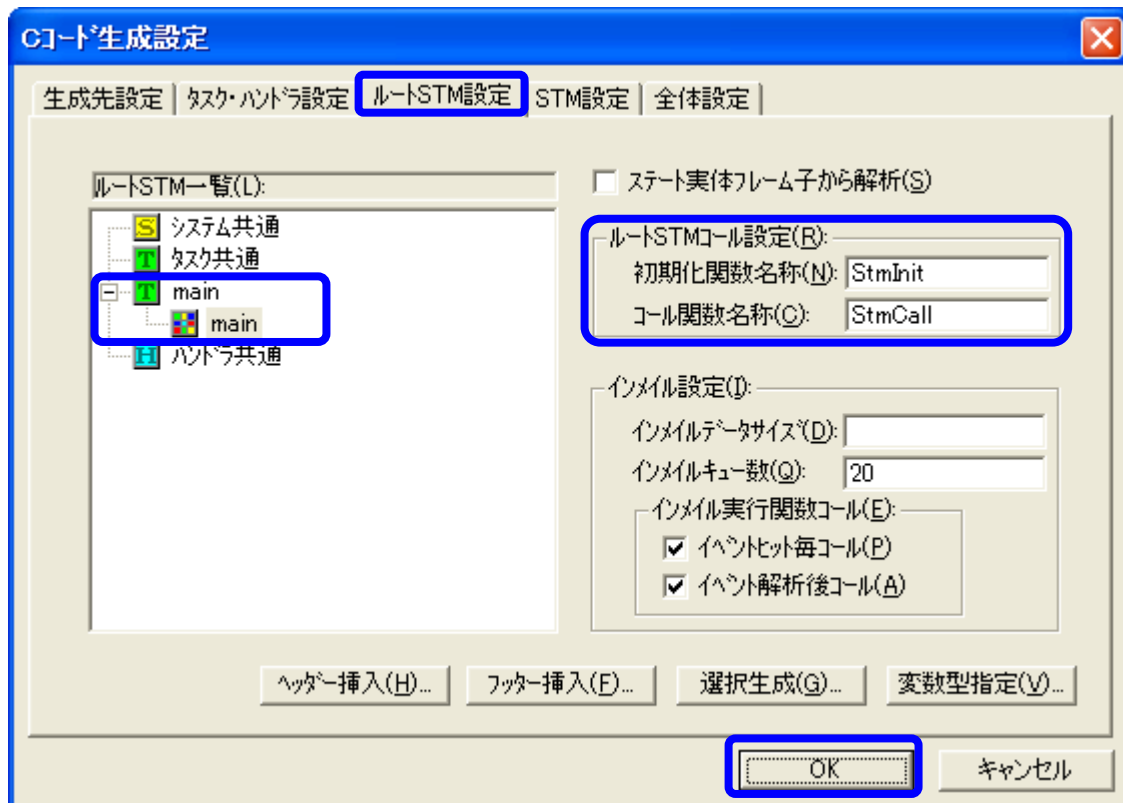
本章では、STM 初期化関数名称、STM コール関数名称を設定します。

手順

1. シミュレーションフェーズに移行します。
2. [プロジェクト] - [プロジェクト設定] - [ジェネレータ設定] - [Cコード生成設定]メニューを選択します。



3. [Cコード生成設定] ダイアログにて、[ルート STM 設定] タブを選択します。
4. [Cコード生成設定] ダイアログ - [ルート STM 設定] タブにて、[ルート STM 一覧] 項目にて、[メイン関数] タスクの [メイン関数] STM 設計書を選択します。
5. [Cコード生成設定] ダイアログ - [ルート STM 設定] タブにて、[ルート STM コール設定] - [初期化関数名称] 項目を "StmInit" に設定します。
6. [Cコード生成設定] ダイアログ - [ルート STM 設定] タブにて、[ルート STM コール設定] - [コール関数名称] 項目を "StmCall" に設定します。
7. [Cコード生成設定] ダイアログにて、[OK] ボタンを選択します。



これで、STM 初期化関数名称、STM コール関数名称の設定は完了です。

3.3 シミュレーション

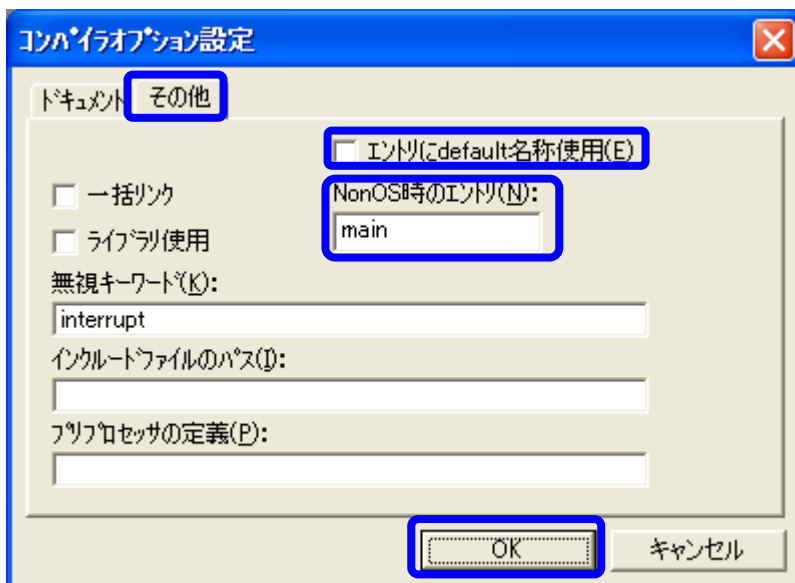
前章までで作成したプロジェクトでは、シミュレーションデバッグすることができません。
本章では、本プロジェクトにてシミュレーションデバッグを行う為に必要な手順を説明します。

手順

1. [プロジェクト] - [プロジェクト設定] - [シミュレーション実行コード設定]メニューを選択します。



2. [コンパイラオプション設定] ダイアログにて、[その他] タブを選択します。
3. [コンパイラオプション設定] ダイアログ - [その他] タブにて、[エントリに default 名称使用] 項目のチェックを外します。
4. [コンパイラオプション設定] ダイアログ - [その他] タブにて、[NonOS 時のエントリ] 項目を "main" に設定します。
5. [コンパイラオプション設定] ダイアログにて、[OK] ボタンを選択します。



これで、シミュレーション環境を構築し、シミュレーションデバッグすることができるようになります。