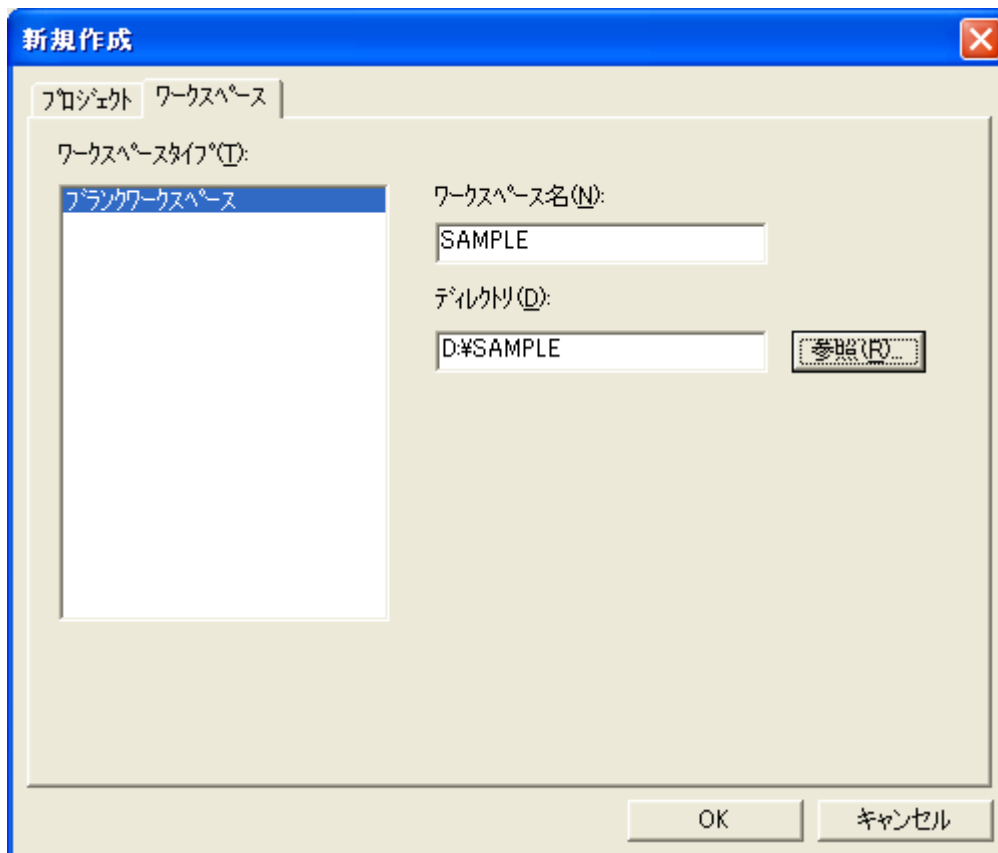


I/Oレジスタファイル使用方法 (例：16bitマイコン用C言語の場合)

Ver:2.0 2009年7月15日
富士通マイクロエレクトロニクス株式会社

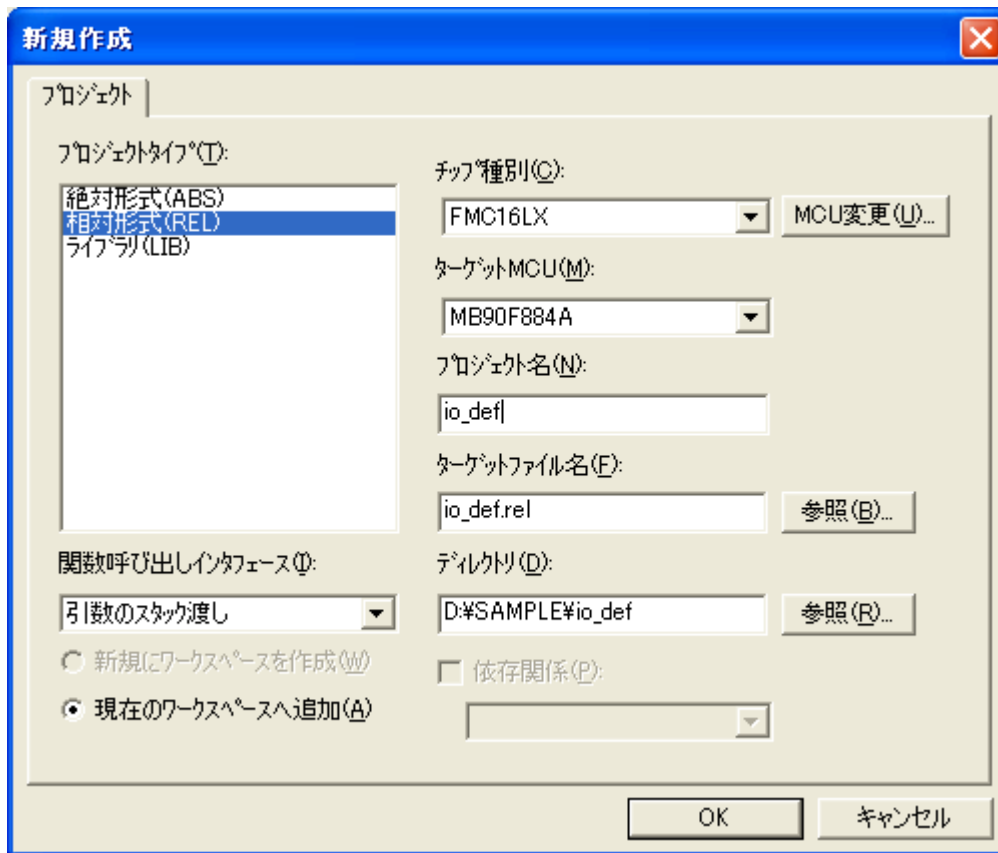
- 【手順1】 使用する I/O レジスタファイルをダウンロードします。
- 【手順2】 ダウンロードした zip ファイルを任意のディレクトリに解凍します。
- 【手順3】 SOFTUNE Workbench を起動し、ワークスペースを作成します。

ファイル→新規作成→ワークスペース/プロジェクトファイル→ワークスペース



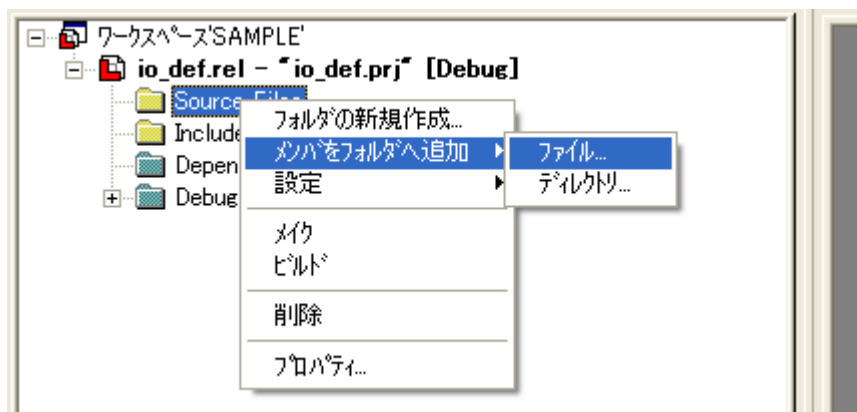
【手順4】 相対形式(REL)プロジェクトを作成します。

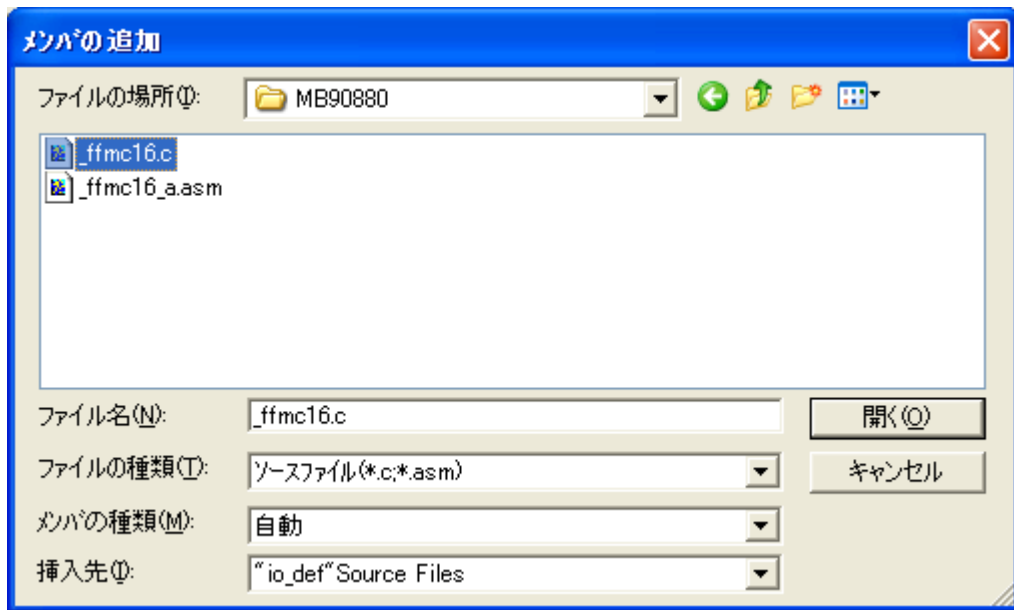
ファイル→新規作成→ワークスペース/プロジェクトファイル→プロジェクト(相対形式(REL))



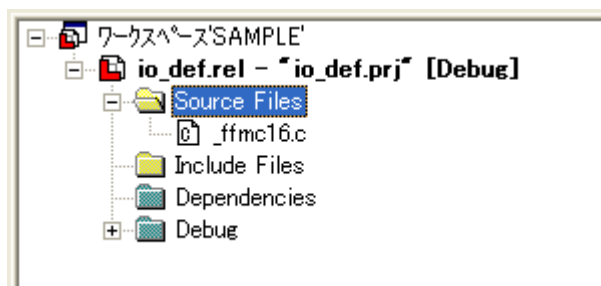
【手順5】 I/O レジスタファイルをメンバに追加します。

Source Files(右クリック)→メンバをフォルダへ追加→ファイル

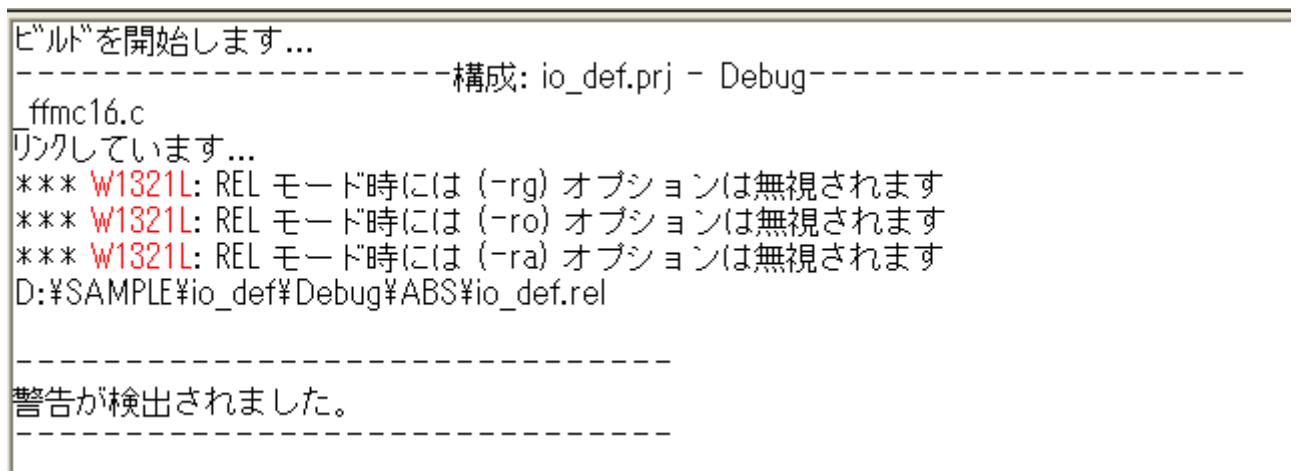




【補足説明】 *.c ファイルが複数ある場合は、*.c ファイル全てを追加して下さい。

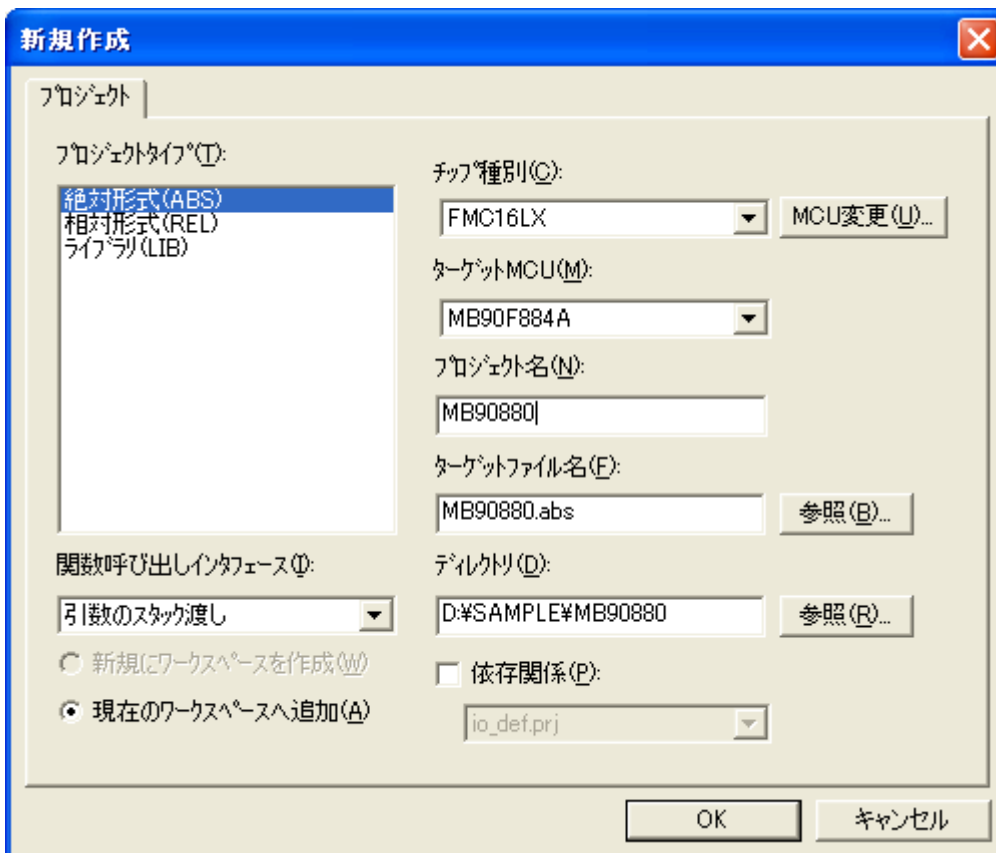
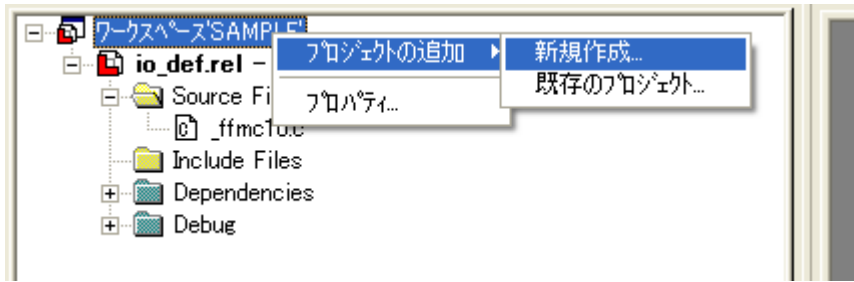


【手順6】 メイク/ビルドでエラーがないことを確認します。

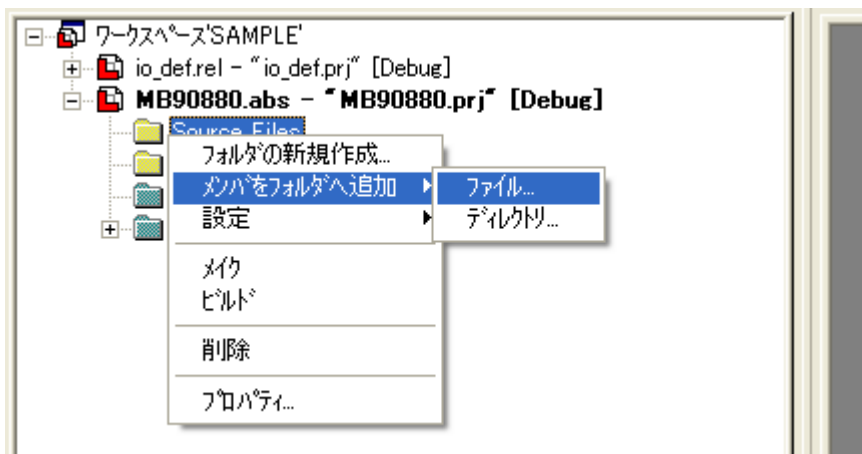


【補足説明】 相対形式(REL)を作成したため、W1321L ワーニングは、無視して下さい。

【手順7】絶対形式(ABS)プロジェクトを作成します。



【手順8】ユーザプログラムをメンバに追加します。



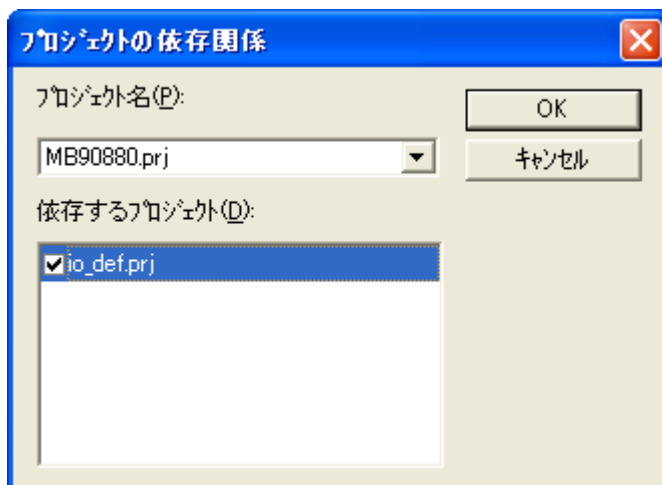
I/O レジスタ変数を参照するユーザプログラムを記述する場合は、以下のように記述して下さい。

例：

```
#include "_ffmc16.h"
void func0{
.....
        IO_PDR0.bit.P00 = 1;
.....
}
```

【手順9】 I/O レジスタファイルと、ユーザプログラムを、プロジェクトの依存関係で関連付けます。

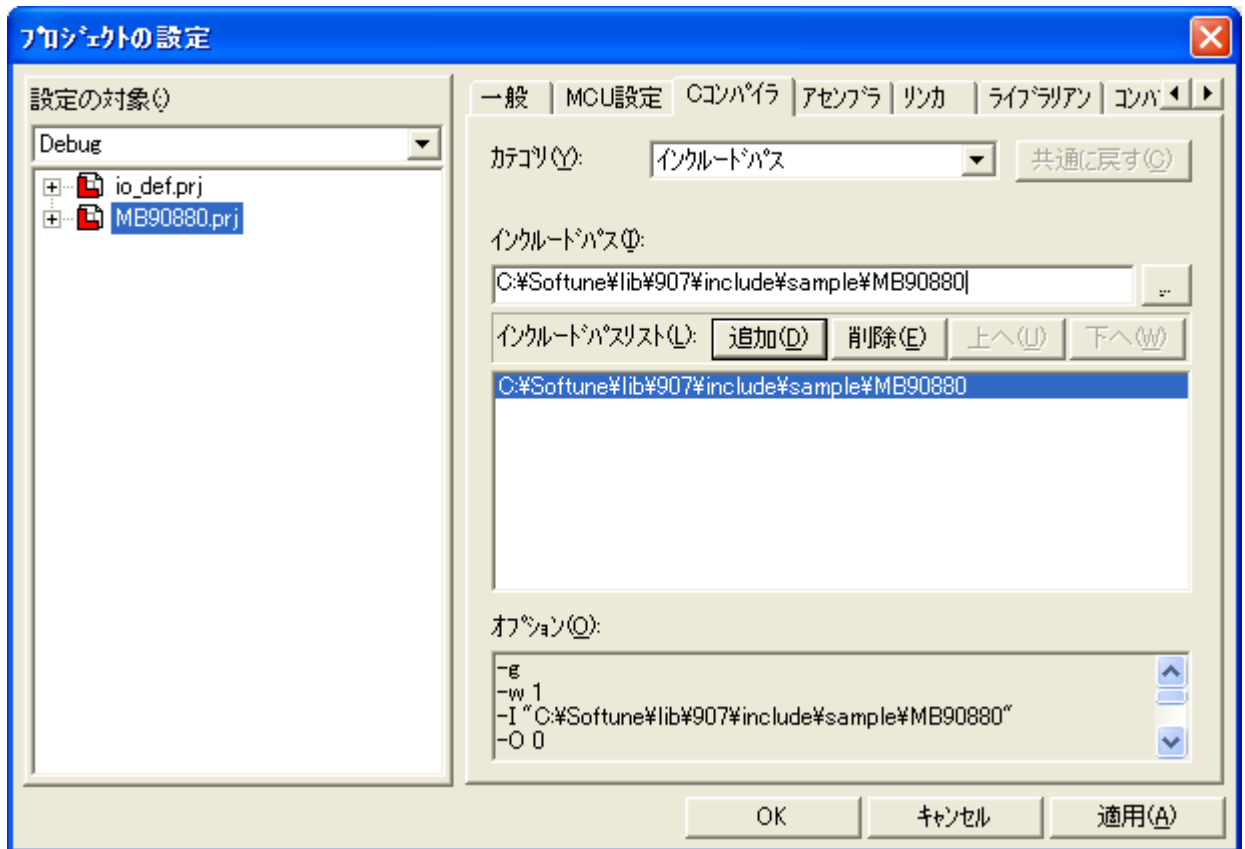
プロジェクト→プロジェクトの依存関係



【手順10】 インクルードパスを設定します。

プロジェクト→プロジェクトの設定→C コンパイラ→カテゴリ(インクルードパス)

【補足説明】 インクルードパスを選択した後、追加ボタンを押すことにより、有効になります。



【手順11】 メイク/ビルドでエラーがないことを確認します。

```
ビルドを開始します...
-----構成: io_def.prj - Debug-----
ffmc16.c
リンクしています...
*** W1321L: REL モード時には (-rg) オプションは無視されます
*** W1321L: REL モード時には (-ro) オプションは無視されます
*** W1321L: REL モード時には (-ra) オプションは無視されます
D:\SAMPLE\io_def\Debug\ABS\io_def.rel

-----構成: MB90880.prj - Debug-----
main.c
リンクしています...
*** I0312L: S.C.Fチェックで0個のワーニングが出力されました
D:\SAMPLE\MB90880\Debug\ABS\MB90880.abs

-----
警告が検出されました。
-----
```

【補足説明】 I/O 定義ファイルは、相対形式(REL)プロジェクトのため、W1321L ワーニングは無視して下さい。

以上