

## コンパイルオプションとオブジェクトファイルの設定

# Contents

コンパイルオプションとオブジェクトファイルの設定.....	3
-------------------------------	---

## コンパイルオプションとオブジェクトファイルの設定

Makefileのターゲット依存部で以下で説明する変数を定義する時には、`:=`を用いて、それまでの定義に追加する形で行う。例えば、コンパイラに対するその他のオプションとして「`-Wall -g -O2`」を追加したい場合には、「`COPTS := $(COPTS) -Wall -g -O2`」という記述をMakefileのターゲット依存部に含める。

(2-4-1) COPTS コンパイラに対するその他のオプション

(2-4-2) CDEFS マクロ定義オプション (-Dオプション)

(2-4-3) INCLUDES ヘッダファイルの置かれたディレクトリ指定オプション (-Iオプション)

(2-4-4) LDFLAGS リンカに対するその他のオプション

(2-4-5) LIBS ライブラリリンク指定のためのオプション

ターゲットに依存して、すべてのソースファイルに共通するコンパイルオプションの追加が必要な場合には、オプションの種類毎に上に示した変数に定義する。

ほとんどの場合に、ターゲット依存部ディレクトリのMakefile.targetには、以下の記述を含める必要がある。

```
INCLUDES := $(INCLUDES) -I$(TARGETDIR)
```

ここでTARGETDIRは、ターゲット依存部ディレクトリに定義されている。またSRCDIRは、Makefileにおいて、ソースプログラムのルートディレクトリ（configureが置かれているディレクトリ）に定義されている。

コンパイラの問題等で、警告メッセージが出ることを防げない状況以外では、COPTSに`-Werror`を追加することを推奨する。

```
COPTS := $(COPTS) -Werror
```

カーネルのコンパイル時に、`dereferencing type-punned pointer will break strict-aliasing rules`という警告が出る場合には、以下を追加するとよい。この警告メッセージに関する詳細は、「TOPPERS/ASPカーネル 設計メモ」の「型キャストに伴う警告メッセージ」の節を参照すること。

```
KERNEL_CFLAGS := $(KERNEL_CFLAGS) -fno-strict-aliasing
```

また、アセンブリ言語レベルの識別名が、C言語レベルの識別名の先頭に`_`が付いたものになる場合には、CDEFSに`-DTOPPERS_LABEL_ASM`を追加する。

(2-4-6) SYSSVC\_DIR システムサービスのソースが置かれたディレクトリ

(2-4-7) SYSSVC\_ASMOBS アセンブリ言語で記述されたシステムサービスのオブジェクト

(2-4-8) SYSSVC\_COBS C言語で記述されたシステムサービスのオブジェクト

(2-4-9) SYSSVC\_CFLAGS システムサービスに対するコンパイルオプション

(2-4-10) SYSSVC\_LIBS システムサービスに対するライブラリリンク指定

ターゲットに依存して、システムサービス（システムログタスクやデバイスドライバなど）のソースが置かれたディレクトリ、システムサービスを構成するオブジェクトファイルのリスト、それらをコンパイル

する際に適用するコンパイルオプション、その構成に必要なライブラリリンク指定を追加する場合には、上に示した変数に定義する。

(2-4-11) `KERNEL_DIR` カーネルのソースが置かれたディレクトリ

(2-4-12) `KERNEL_ASMOBS` アセンブリ言語で記述されたカーネルのオブジェクト

(2-4-13) `KERNEL_COBS` C言語で記述されたカーネルのオブジェクト

(2-4-14) `KERNEL_CFLAGS` カーネルに対するコンパイルオプション

ターゲットに依存して、カーネルのソースが置かれたディレクトリ、カーネルを構成するオブジェクトファイルのリスト、それらをコンパイルする際に適用するコンパイルオプションを追加する場合には、上に示した変数に定義する。

ほとんどの場合に、ターゲット依存部ディレクトリの`Makefile.target`には、以下の記述を含める必要がある。

```
KERNEL_DIR := $(KERNEL_DIR) $(TARGETDIR)
KERNEL_ASMOBS := $(KERNEL_ASMOBS) target_support.o
KERNEL_COBS := $(KERNEL_COBS) target_config.o target_timer.o
```

(2-4-15) `CFG_TABS` コンフィギュレータに対するオプション

ターゲットに依存して、コンフィギュレータに対するオプションを追加する場合には、上に示した変数に定義する。具体的には、コンフィギュレータの値取得シンボルテーブルのターゲット依存部 (`target_def.csv`) がある場合には、それを指定するオプションを追加する必要がある。

ほとんどの場合に、ターゲット依存部ディレクトリの`Makefile.target`には、以下の記述を含める必要がある。

```
CFG_TABS := $(CFG_TABS) --cfg1-def-table $(TARGETDIR)/target_def.csv
```

(2-4-16) `CFG1_OUT_LDFLAGS` `cfg1_out.c`に対するリンクオプション

ターゲットに依存して、`cfg1_out.c`をリンクする際に適用するオプションを追加する場合には、上に示した変数に定義する。

(2-4-17) `CFG_OBS` システムコンフィギュレーションのためのオブジェクト

ターゲットに依存して、システムコンフィギュレーションのためのオブジェクトファイル (`kernel_cfg.o`以外に) 追加する場合には、上に示した変数に定義する。この場合、追加したオブジェクトファイルの作成ルール (コンパイル/アセンブルルールと依存関係作成ルール) を、`Makefile`のターゲット依存部に記述する必要がある。

(2-4-18) `CFG2_OUT`

ターゲットに依存して、コンフィギュレータのパス2で`kernel_cfg.c`と`kernel_cfg.h`以外のファイルを生成する場合には、上に示した変数に定義する。

(2-4-19) `OMIT_WARNING_ALL`

(2-4-20) `OMIT_OPTIMIZATION`

サンプルの`Makefile`では、コンパイラに対するオプションに「`-Wall -g -O2`」を追加する。`-Wall`を追加したくない場合には、`Makefile`のターゲット依存部で`OMIT_WARNING_ALL`を`"true"`に定義する。`-O2`を追加したくない場合には、`Makefile`のターゲット依存部で`OMIT_OPTIMIZATION`を`"true"`に定義する。