C言語で定義済みマクロによるコンパイラ及びOS判定

C言語で様々なOSやコンパイラで共通ソースにしたい時、条件コンパイルで切換えたい場合がある。その時に利用するマクロ名について、備忘録を残す。

定義済みマクロ名

定義済みマクロは、下記コマンドで確認できる。

```
$ echo | gcc -dM -E -
```

```
$ echo | arm-none-eabi-gcc -dM -E -
```

コンパイラ判定

コンパイラ	マクロ名	定義値	備考
GCC	__GNUC__	バージョン番号	
GCC/ARM	__arm__		
IAR EWARM	__ICCARM__		
ARMコンパイラ	__CC_ARM		__ARMCC_VERSIONはバージョン番号
インテルコンパイラ	__INTEL_COMPILER		
Borland C++	__BORLANDC__		
Appleコンパイラ	__APPLE__		

実行環境判定

実行環境	マクロ名	定義値	備考
UNIX	__unix__		__unixでもよい
Linux	__linux__		__linuxでもよい
Windows(32bit)	_WIN32		WIN32でもよい
Windows(64bit)	_WIN64		
msys2(32bit)	__MINGW32__		
msys2(64bit)	__MINGW64__		
Cygwin	__CYGWIN__		
macOS	__MACH__		(Apple silliconでも可)

64bit環境判定

実行環境	マクロ名	定義値	備考
Linux/macOS	__x86_64		
macOS(Apple sillicon)	__arm64__		

参考

1. C言語 マクロ コンパイル 判定 OS コンパイラ
2. cross platform development
3. 条件コンパイル（インテルコンパイラ icc の判別）
4. gcc のプリプロセッサの定義済みマクロ(Predefined Macros)の確認方法
5. コンパイラの種類とバージョンを知るためのpre-defineされたマクロは？
6. GCC/定義済みマクロ/ARM

From:
https://yokoweb.net/dokuwiki/ - **新石器Wiki**

Permanent link:
https://yokoweb.net/dokuwiki/programing/c-language/compiler-os-macro-define

Last update: **2026/07/28 13:16**