

# ColdFire用GCCとGDBスタブについて

横田 敬久

## ● GCCによるコンパイル環境を用意

今月号の特集記事では、CインタプリタのSilentC、CコンパイラのCodeWarriorという2種類のプログラミング環境が解説されています。次号(10月号)ではこれらに続く第3のプログラミング環境として、GCCによるコンパイル環境について解説する予定です。

GCCを使うためには、手元にあるWindowsマシンにCygwinをインストールし、Cygwin上で動作するColdFire用GCCをビルドします。

## ● 付属ColdFireマイコン基板にはGDBスタブを書き込み済み

付属ColdFireマイコン基板に搭載のCPU内蔵フラッシュROMには、出荷状態で既にGDBスタブを書き込み済みです。基板上のジャンパJP2をショート状態で電源を投入すると、CPU内蔵フラッシュROMの後半128Kバイトに格納されたGDBスタブが起動します。

本GDBスタブは、Ethernet経由で実行ファイルをダウンロードします。しかも、ダウンロード先のアドレスが内蔵フラッシュROM領域の場合、内蔵フラッシュROMをブロック単位で消去しながら書き込んでいきます。SilentCやGDBスタブの領域と重ならないROM領域に、ダイレクトにダウンロードが可能です。

さらに、CPU内蔵のハードウェア・ブレーク機能を使ってい

るので、ROM上のプログラムに対しても最大4箇所までのブレークポイントを設定可能です。

## ● 各種サンプル・プログラムを用意

ColdFireが内蔵する周辺機能を制御する各種サンプル・プログラムを用意しました。

例として、付属ColdFireマイコン基板のアクティブLEDを点滅させるサンプル・プログラムのファイル内容を表Aに示します。

このサンプル・プログラムのアーカイブ・ファイルを解凍したディレクトリをカレント・ディレクトリとして、

```
$ make
```

を実行すると、各ソースがコンパイルされ、実行形式ファイル(拡張子.elf)が生成されます。

さらに、

```
$ m68k-elf-gdb
```

を実行すると、GDB用イニシャライズ・ファイル.gdbinitが読み込まれ、cf\_led.elfが付属ColdFireマイコン基板にダウンロードされます。ダウンロードが完了したら、

```
(gdb) c
```

と入力すると実行が開始されます。

## ● 本誌ダウンロード・ページからダウンロード可能

残念ながら誌面の都合で、今月号はGCCについての詳しい解説はできません。しかし本誌10月号の発売を待たなくても、これらのサンプル・プログラムを9月号を発売直後にダウンロードできるように準備する予定です。GCCを使い慣れた読者であれば、すぐにサンプル・プログラムを動かせるでしょう。

よこた・たかひさ  
来栖川電工(有)

表A LED点滅制御サンプル・プログラム内容一試

README.TXT	ファイル内容の説明
Makefile	Makeファイル
Makefile.config	Makeファイル・コンフィグレーション
Makefile.RAM	RAM配置用Makeファイル
Makefile.ROM	ROM配置用Makeファイル
m52235.h	CPU内蔵レジスタ定義ファイル
m68kelf.m52235evb.ld	リンカ・スクリプト
main.c	メイン関数Cソース
main.o	メイン関数オブジェクト
mini_lib.c	簡易ライブラリCソース
mini_lib.h	簡易ライブラリ・ヘッダファイル
mini_lib.o	簡易ライブラリ・オブジェクト
start.S	スタートアップ・ルーチン・アセンブラ・ソース
start.o	スタートアップ・ルーチン・オブジェクト
cf_led.elf	実行形式ダウンロード・ファイル
.gdbinit	GDB用イニシャライズ・ファイル