

Cにちょっとアセンブラ… 全ARMで動く共通プログラム

村井 和夫

本稿では、32ビット系ARMを例にして、全ARMで動かす共通プログラムの基本構造について解説します(図1)。ターゲットARMボードを以下に示します。

- ラズベリー・パイ (ARM11 コア)
- ラズベリー・パイ2 (ARM Cortex-A7 コア)
- BeagleBone Black 基板 (ARM Cortex-A8 コア)
- CEV RZ/A1L 基板 (ARM Cortex-A9 コア)
- LPC2388 基板 (ARM7TDMI コア)

本章では、32ビットARM共通のプログラム記述方法を解説していきます。今回はその後、ラズベリー・パイやラズベリー・パイ2について、CPU固有機能のプログラムを作成する方法を紹介します。

他の32ビット系ARMボードや、アーキテクチャが大きく異なる64ビットARM系ラズベリー・パイ3(64

ビット対応ARM Cortex-A53コア)、Cortex-M系FM3マイコン基板(ARM Cortex-M3コア)も基本は同様です。対応の詳細については、誌面の都合で全てを本特集で紹介することは難しいため、今後連載などでフォローアップしていきます。(編集部)

プログラミング体得! ARM共通プログラムの仕様

ARM共通プログラムの要求条件として、以下の項目を満たすものとします。

- (1) I/O処理を行う
 - 代表的なI/O処理をひと通り行います
 - 汎用デジタルI/O (GPIO)
 - シリアル通信 (UART)

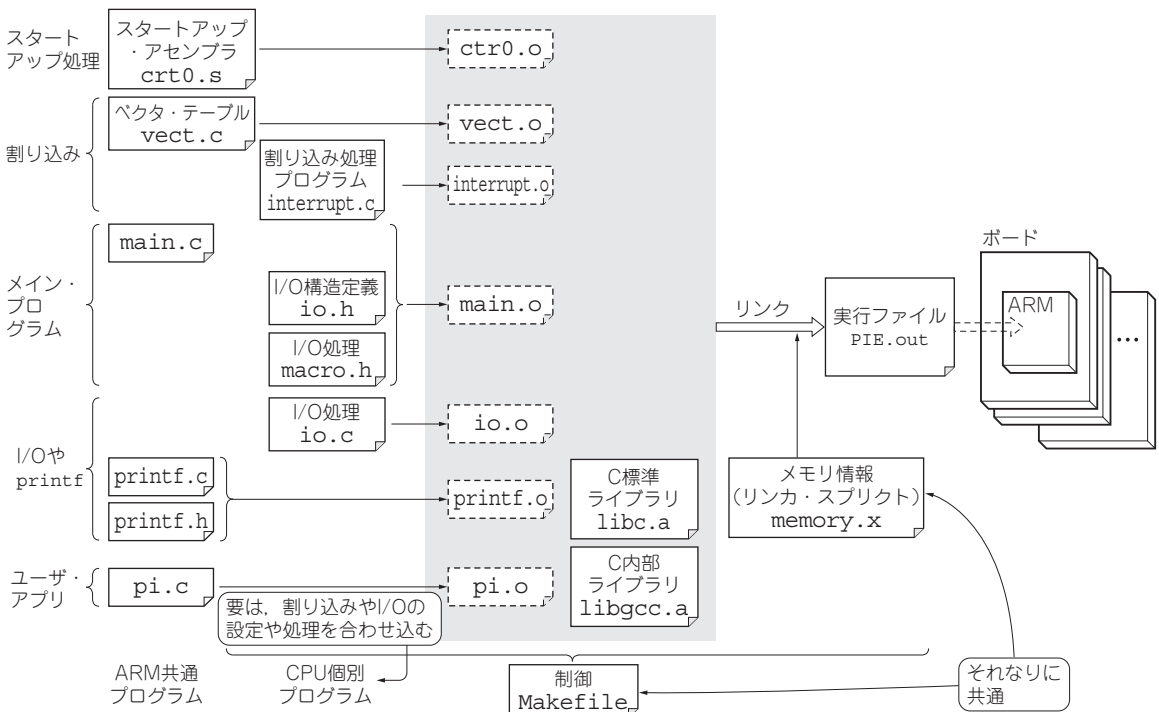


図1 ARM共通プログラムの構成…割り込みやI/O関連の設定や処理を合わせこめばどのARMで動かすことも可能になる