## Linux コーナ

内部動作がわかればこわくない! 高性能 ARM ボードをハード制御に活用するために

## マイコン・プログラマのための Linux超入門

第12回

ちゃんとできたらプロ!····デバイス・ツリー対応 最新Linuxでオンチップ/オンボード回路を動かす方法

宗像 尚郎

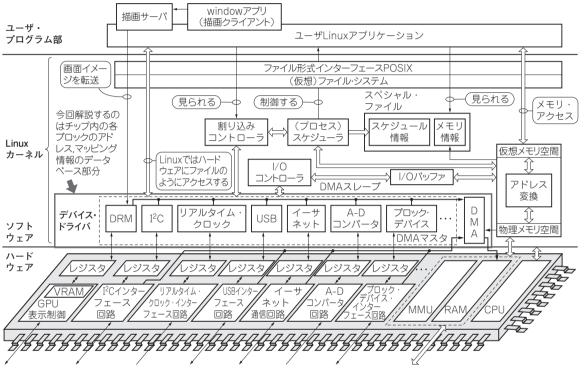


図1 今回解説すること…ARM製品をサポートする上で対応必須!デバイス・ツリーの実装方法や対応を解説 早々にデバイス・ツリー対応を行ったBeagleBone Blackのコードを参考にする

Linux は、x86やARMなどいろいろなCPUアーキテクチャやいろいろなCPUボードでも動作するマルチプラットホーム対応OSです。前回は、Linuxがマルチプラットホームを実現するしくみと、ARMプロセッサをサポートする上で対応が必須となったチップ/ボード固有回路を動かすしくみデバイス・ツリーを紹介しました(図1).

今回は、デバイス・ツリー対応済みのボードである BeagleBone Blackを例に、前回触れなかったデバイス・ツリー関連ファイルの記述や動作を説明します. (編集部)

## ● デバイス・ツリー完全対応でお手本にピッタリ! BeagleBone Black

今回は、BeagleBone Blackでデフォルトで採用されている $^{\dot{\mu}1}$ カーネル・バージョン 3.8 の Angstrom のデバイス・ツリー関連ファイルのコードを見ていきます。 さらに、BeagleBone Blackの拡張ボード Cape をデバイス・ツリーでサポートする Cape Manager について解説します。

## ▶ BeagleBone Black はこんなボード

BeagleBone Blackは、AM335xシリーズ(テキサス・インスツルメンツ)を搭載したLinuxボードです。

注1:最新のRevision CではデフォルトのディストリビューションがDebian に変わりました.

第1回 電源投入からカーネルが起動するまでの動作(2013年11月号)

第2回 仮想アドレスのメカニズム (2013年12月号)

第3回 高性能処理のキモ! プログラム実行順序決定のメカニズム (2014年1月号)

Interface 2014年11月号