

第4章 ARM Cortex-A9 搭載！全部入り最新FPGAの研究
～アルテラ SoC 編～

Android アプリケーションからハードウェアを制御する方法

伊藤 裕之 Hiroyuki Ito

第2章は Helio ボードで Linux を起動し、アプリケーションを作成して動かすところまでが解説されています。Linux が起動したら、次は Helio ボードで Android を起動してみましょう。ここでは Android を起動した後、Helio ボード上の LED を、Android アプリから ON/OFF 制御する方法について解説します。

1. Android デモ・システム概要

● **ターゲットは Helio ボード**

今回はアルティマから販売されている、Cyclone V SoC の開発キットである Helio ボード (Rev1.2) を使用して、ユーザ LED を Android アプリケーションから制御するシステムの開発手順について解説します。ここで作成するものとしては、

- FPGA 論理回路
 - Linux デバイス・ドライバ
 - Java Native Interface (JNI)
 - Android アプリケーション
- となります。

● **LCD ボードを拡張**

基板の構成は、Macnica Helio SoC Evaluation Kit (アルティマ) と、7 インチの LCD Touch Screen (Terasic 社製) です。基板の詳細についてはアルテラ SoC コミュニティ・サイトの Rocketboards.org

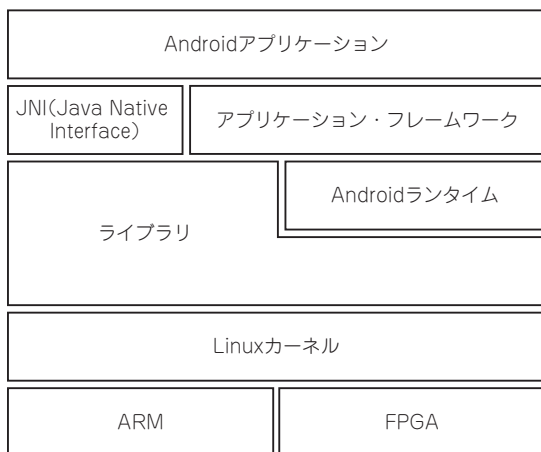


図1 Android デモ・システムの構成図

(<http://www.rocketboards.org/foswiki/Documentation/MacnicaHelioSoCEvaluationKit>), もしくはアルティマのホームページを参照してください。

システムの構成は図1のようになっています。画面から Android アプリケーションのボタンを押すと、Android アプリケーション → JNI → Linux デバイス・ドライバ → FPGA 論理回路という経由で処理を行います。

2. ユーザ・ロジック開発手順

● **FPGA に実装する回路**

まずは Helio ボード上の LED を点灯制御する回路を FPGA 部分に実装します。これまでの FPGA マガジンで解説された内容と重複しますが、開発手順を説明します。

Altera 社製 Quartus II 開発ソフトウェアおよびアルテラ SoC エンベデッド・デザイン・スイート (EDS) をインストールします。今回はバージョン 13.1 を例に挙げて説明します。

Rocketboards.org サイトの Macnica Helio SoC Evaluation Kit から、Helio ボード用のリファレンス・デザインがダウンロードできるので、このプロジェクトを使用して開発を始めていきます。まずこのプロジェクトを Quartus II で立ち上げ、Qsys を立ち上げます (図2)。

● **既に led_pio が実装されている**

LED 処理に関しては図2からも分かるように、既に今回使用するサンプル・プロジェクトで実装されており、Helio ボード上の LED3 ~ 6 を制御できるようになっています。今回はこの led_pio の IP コアを制御します。

この LED IP コアを Android アプリケーションから制御するために、まず Android が動作する必要があるため、LCD モジュールおよびタッチパネル・モジュールの追加を行います。また Android を動作させるため