

# オープンソースCPU 「RISC-V」の研究

第8回 定番実装Rocket ChipをFPGAで動かして「hello!」

ご購入はこちら

@msyksphinz

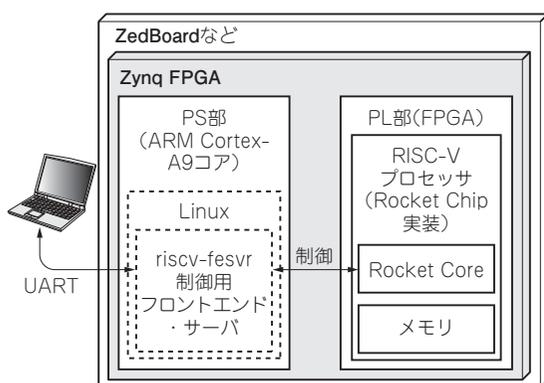


図1 RISC-Vの定番実装Rocket CoreをFPGAで動かしてみる  
Zynqを使って内蔵ARMコアから制御する

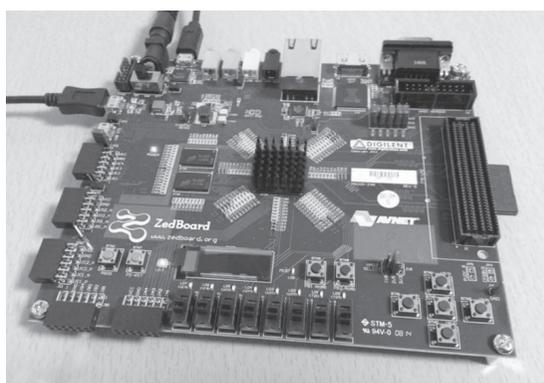


写真1 インターフェースをいろいろ備えて実験がしやすい元祖  
Zynqボード「ZedBoard」で試してみる  
秋月電子通商などで6万円程度(論理合成ツールVivado付き)

## 実験の構成

### ● 使用するFPGA

今回は、Rocket ChipをRTLシミュレーションではなく、FPGAに書き込んで、動作させてみましょう。Rocket ChipをFPGAで動作させるための構成を図1に示します。

Rocket ChipはARMコア内蔵FPGAであるZynq(ザイリンクス)のPL(Programmable Logic)部に実装して動かします。Rocket Chipの制御は、ZynqのPS(Processing System)部に搭載されたARMコア上でLinuxを動作させ、そこからコマンドを出して制御します。

このため、Rocket Chipを動作させるためには、PL部へRocket Chipのインプリメントと、PS部へのLinux構築、そして制御プログラムの実装が必要になります。これらの構築は、全てfpga-zynqリポジトリ(後述)で構築することができます。

### ● 論理合成ツールVivado

FPGAの論理合成にはザイリンクス製のVivado論理合成ツールが必要です。論理合成および配置配線にはVivado 2016.2を使用することが前提としてスクリプトが組まれています。しかしある程度バージョンがずれていても、スクリプトのバージョン指定を変更すれば特に問題なさそうです。

### ● 使用したZynqボード

筆者は今回の作業をZedBoard上で確認を行いました(写真1)。使用したVivadoのバージョンは、2016.2です。

## Rocket ChipをFPGAに構築する方法

### ● 最もシンプルな1コアのコンフィグで動かす

RISC-VのRocket Chipにはさまざまなコンフィグレーションが存在します。一般的な構成から、カスタム設計されたアクセラレータが搭載されている構成、バスの幅やCPUのコア数を変更した構成などがあり、

注：本稿の内容は執筆時点のもので、随時更新されていく可能性があります。

編集注：本誌2018年2月号特集2「新時代プロセッサ システム作り」ではバークレイ以外のさまざまなコアも紹介しています。