

2017 年 10 月 13 日

株式会社 **インプレスR&D**

<http://nextpublishing.jp/>

技術書典シリーズ第四弾！

「ソフトウェア技術者のための FPGA 入門 機械学習編」発行

FPGA の活用法をプログラマのためにわかりやすく解説！

インプレスグループで電子出版事業を手がける株式会社インプレス R&D は、『ソフトウェア技術者のための FPGA 入門 機械学習編』（著者：石原ひでみ）を発行いたします。

『ソフトウェア技術者のためのFPGA入門 機械学習編』

<https://nextpublishing.jp/isbn/9784844398004>



著者：石原ひでみ

小売希望価格：電子書籍版 1800 円（税別）／印刷書籍版 2400 円（税別）

電子書籍版フォーマット：EPUB3／Kindle Format8

印刷書籍版仕様：B5 判／モノクロ／本文 170 ページ

ISBN：978-4-844398004

発行：インプレス R&D

<< 内容紹介 >>

【ソフトウェアプログラマのための FPGA チュートリアルガイド！】

プログラム可能なハードウェア「FPGA」をソフトウェア技術者が活用するためのチュートリアルガイドです。

FPGA の基礎知識からツールの入手方法、機械学習を例にした実際の動作までを図表やツールの動作、ソースコードの紹介を交えてわかりやすく紹介しています。

（本書は、次世代出版メソッド「NextPublishing」を使用し、出版されています。）

FPGA の仕組みをわかりやすく紹介

図 1-1: FPGA の構成

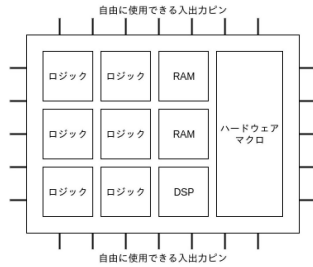
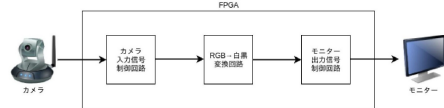


図 1-2: 自動画像フィルタのシステム構成例



FPGA の得意分野

FPGA を使用してハードウェア・プログラミングを行うためには、FPGA が得意とするところ、そして、FPGA をどのように使用するとメリットが発生するのかを理解してプログラミングする必要があります。

独自回路の設計

FPGA は自由に回路を構成し設計できることから、何らかの目的専用の処理を行う独自回路を開発することができます。例えば、「独自開発の汎用的ではない通信方式」や「特殊な映像効果を与える画像処理」などを FPGA 上で独自に回路を開発して実現することが可能です。この時「汎用的な処理を行う回路」とは、CPU の周辺ペリフェラル (UART や SATA、PCIe などのインターフェース) などを指します。多くのコンピュータはこれら汎用の回路をハードウェア・コントローラ (ハードウェア・マクロ) として実装しています。そして、汎用のハードウェア・コントローラが実現できないことを FPGA では実現可能です。

並列処理

FPGA では処理を並列して行うことができます。CPU ではプログラミングされた命令を逐次実行していきますが、FPGA では同時にいくつでも並列して処理する回路を設計することができます。

例えば、リスト 1-1 のように for ループ文で 100 回の乗算をするようなソースコードの場合、一般的に CPU では 100 回の乗算を繰り返し行いますが、FPGA ではリスト 1-2 のように 100 個の乗算器を並べて配置し一度の時間軸に並列に演算することも可能です。

リスト 1-1: for ループでの演算

```
for( I = 0; i < 100; ++i){
    z[i] = x[i] * y[i];
}
```

リスト 1-2: 並列に並べられた演算

```
z[0] = x[0] * y[0];
z[1] = x[1] * y[1];
z[2] = x[2] * y[2];
...
z[98] = x[98] * y[98];
z[99] = x[99] * y[99];
```

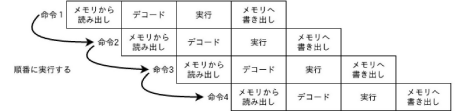
CPU と FPGA の違い

FPGA を用いたハードウェア・プログラミングを行うために CPU と FPGA の処理方法の違いを理解しておきましょう。

CPU でのソフトウェア処理

CPU では、ソフトウェアの実行をプログラムで記述されている順番に、CPU コアに用意されている命令処理回路で処理を行います。図 1-3 のように、プログラミングされた命令はメモリから読み出され、デコード、実行、メモリに書き戻す一連の動作を繰り返します。

図 1-3: CPU の処理



FPGA の開発環境の整備についてハードの紹介も含め詳しく解説

第二章 開発環境の整備

FPGA でハードウェア・アクセラレーションするための開発環境は各社から様々なツールが提供されていますが本書では Xilinx 社の SDSoC を使用して開発を行っています。本章では SDSoC の導入方法を解説します。

SDSoC とは

SDSoC とは Xilinx 社の FPGA を使用した高位合成のアプリケーション開発環境です。FPGA の知識がなくてもソフトウェアをシームレスに FPGA へアクセラレーションする開発ツールです。SDSoC は次の Xilinx 社のツールも含んだ高位合成の統合開発環境になっています。

- ・ Vivado Design Suite
- ・ Vivado HLS (High Level Synthesis)

SDSoC の開発環境は Eclipse ベースの IDE が提供されており、FPGA へのアクセラレーションなどの設定を一括で管理し、開発が行えるようになっています。

SDSoC は C/C++ で開発したソースコードのうち関数単位でハードウェア・アクセラレーションが可能であり、ソフトウェアを開発したソフトウェア・エンジニアが FPGA や HDL を知らなくても指定した関数をシームレスに FPGA にハードウェア・アクセラレーションする環境を整えてくれます。

開発環境 (Linux 環境)

本書では Linux Ubuntu 16.04 LTS をベース PC として Xilinx 社の SDSoC をインストールしてハードウェア・プログラミングを行う開発環境を解説しています。SDSoC は次の環境にインストールすることが可能です。

- ・ Ubuntu 16.04 LTS (64bit)
- ・ Red Hat Enterprise Workstation/Server 7.2 and 7.3 (64-bit).
- ・ Red Hat Enterprise Workstation 6.7 and 6.8 (64-bit).
- ・ Windows 10 Professional (64-bit)
- ・ Windows 7 and 7 SP1 Professional (64-bit) (SDSoC only)

SDSoC のダウンロード

本書では SDSoC 2017.2 で解説を進めています。SDSoC のバージョンが違えば適宜、説

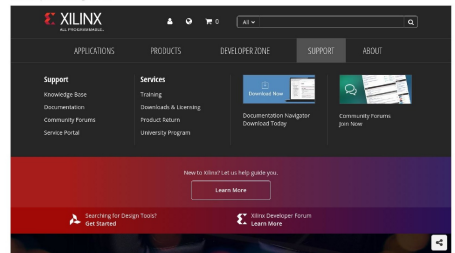
み替えて進めていってください。

Xilinx 社 Web サイト (図 2-1) の画面上位にあるメニューから図 2-2 のように「SUPPORT」を選択し、「Download & Licensing」を選択してダウンロードページへ進みます。

図 2-1: Xilinx Web サイト (https://www.xilinx.com/)



図 2-2: 「SUPPORT」メニュー



ハードウェア・プログラミングについて基礎からチューニングによる性能向上までを説明



本例でのConvolution関数はCNNレイヤが3レイヤになっており、レイヤ0が2回、レイヤ1が4回、レイヤ2が8回実行されます。図6-6はレイヤ0の1回分のトレース情報を拡大しています。図6-6の両端で緑のFPGA化した関数の実行タイミングが少し途切れていますがこれがレイヤ0の1回分の実行時間になります。トレース情報の色が付いている部分にマウスカーソルを乗せるとその配列（レジスタ）が連続で読み書きまたはデータアクセスを行っている時間情報を表示することができます（単位は秒）。図6-6の場合、レイヤ0のデータアクセス時間は65.4msと確認することができました。

図6-6 CNNレイヤ0の実行状況

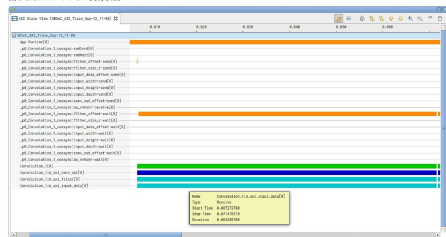


図6-9はレイヤ1の1回分のアクセスタイミングを拡大したのになります。レイヤ0と同じ

ようにトレース情報のところにマウスカーソルをあわせてデータアクセス時間を見ると、7.1msであることが分かります。

図6-9 CNNレイヤ1の実行状況

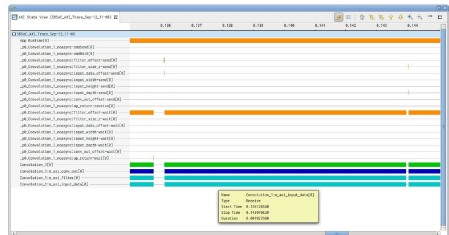


図6-10はレイヤ2の1回分のアクセスタイミングを拡大しています。レイヤ0やレイヤ1と同じようにデータアクセス時間を見ると1.7msとなっています。

図6-10 CNNレイヤ2の実行状況

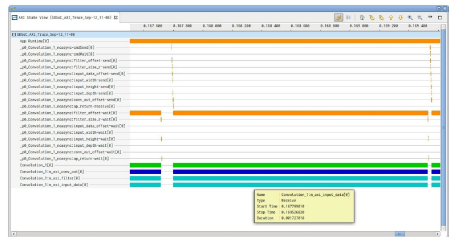


図6-11は1枚の画像をCNN処理する最後の部分を拡大しています。ここでは最後のデータアクセス後、トレース時間の空白の時間が発生しているように見えますが、空白の部分は最後の

<< 目次 >>

第一章 プログラムできるハードウェア

基礎知識:FPGA とは

ソフトウェアのFPGA 化にあたって

FPGA 開発言語

FPGA の処理構成

FPGA の開発フロー

第二章 開発環境の整備

SDSoC とは

開発環境(Linux 環境)

SDSoC のダウンロード

インストール

ライセンスの取得

起動とライセンスの設定

評価ボード

ZedBoard

第三章 ハードウェア・プログラミング（スタートアップ編）

プロジェクトの作成

プロジェクト

ソースコードの作成

コンパイル

実機で動作確認

FPGA 化する関数の指定

第四章 機械学習ソフトウェア

アルゴリズムを確立する

ソースコード

関数の構成

ソフトウェアの動作確認

第五章 ハードウェア・プログラミング(組み込み編)

SDSoC に適用

関数の FPGA 化

SDSoC の pragma で転送方式の指定

FPGA 化する階層を1つ上げる

2つの関数を FPGA 化

上位関数 CNNLayer を対象

第六章 ハードウェア・プログラミング(チューニング編)

アルゴリズムの把握

FPGA 化関数のトレース

データアクセスの修正

上位関数のトレース

メモリアクセス

ソースコードのリファクタリング

メモリアクセスとリファクタリング例

生成される回路規模

HLS プラグマの適用

エミュレータ

<< 著者紹介 >>

石原ひでみ

大手メーカーで LSI 設計、放送・通信機器、ハードウェア OS のベンチャー参画のあと、設計コンサルタントとして活動中。FPGA マガジン(CQ 出版)にて、執筆活動中。「ひでみのアイデア帳」(<http://sweetcafe.jp/>)にて、FPGA 関係の情報を発信中。

<< 販売ストア >>

電子書籍:

Amazon Kindle ストア、楽天 kobo イーブックストア、Apple iBookstore、紀伊國屋書店 Kinoppy、Google Play Store、honto 電子書籍ストア、Sony Reader Store、BookLive!、BOOK☆WALKER

印刷書籍:

Amazon.co.jp、三省堂書店オンデマンド、honto ネットストア、楽天ブックス

※ 各ストアでの販売は準備が整いしだい開始されます。

※ 全国の一般書店からもご注文いただけます。

【株式会社インプレス R&D】 <http://nextpublishing.jp/>

株式会社インプレス R&D (本社：東京都千代田区、代表取締役社長：井芹昌信) は、デジタルファーストの次世代型電子出版プラットフォーム「NextPublishing」を運営する企業です。また自らも、NextPublishing を使った「インターネット白書」の出版など IT 関連メディア事業を展開しています。

※NextPublishing は、インプレス R&D が開発した電子出版プラットフォーム(またはメソッド)の名称です。電子書籍と印刷書籍の同時制作、プリント・オンデマンド(POD)による品切れ解消などの伝統的出版の課題を解決しています。

これにより、伝統的出版では経済的に困難な多品種少部数の出版を可能にし、優秀な個人や組織が持つ多様な知の流通を目指しています。

【インプレスグループ】 <http://www.impressholdings.com/>



株式会社インプレスホールディングス(本社:東京都千代田区、代表取締役:唐島夏生、証券コード:東証1部9479)を持株会社とするメディアグループ。「IT」「音楽」「デザイン」「山岳・自然」「モバイルサービス」を主要テーマに専門性の高いコンテンツ+サービスを提供するメディア事業を展開しています。2017年4月1日に創設25周年を迎えました。

【お問い合わせ先】

株式会社インプレス R&D NextPublishing センター

〒101-0051 東京都千代田区神田神保町 1-105

TEL 03-6837-4820

電子メール: np-info@impress.co.jp