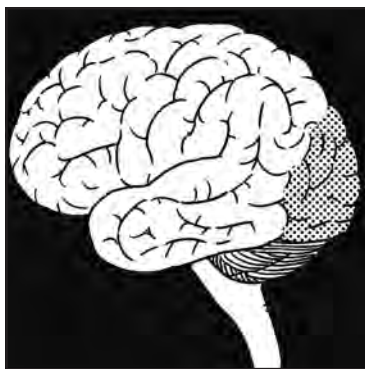


短期連載



クロック周波数1/26で演算時間1/19! 3600並列ニューロン回路が欠陥を自動検出 自力で学習する 大脳視覚野AIチップの製作

第3回 C++言語でAIチップ製作
～ソフトウェア・ワンストップ開発環境 SDSoCの使い方～

安永 守利 Moritoshi Yasunaga

● ワンストップ環境でAIチップをソフトウェア開発
従来、ZynqなどのSoC型FPGAは、次のような流
れでハードウェアとソフトウェアを別々の環境で開発
していました。

- (1) VivadoHLS(ザイリンクス)を使って、高位記
述(CやC++)からHDLに変換したIPコア、
つまりハードウェアを作る
- (2) Vivadoを利用してIPコアを取り込み、HDL
によるハードウェアの全体記述に統合した後、
論理合成、配置配線、コンパイルを行う
- (3) SDKでソフトウェアを作る

最新の統合開発環境であるSDSoCを利用すれば、
C言語やC++言語で、ソフトウェアとハードウェア
で構成されたシステム全体を記述できます。記述後ビルド・ボタンを押すと、FPGA部分であるPL部のビット・ファイルが生成され、同時にソフトウェア部分であるPS部がArm用にコンパイルされます。

今回は、ワンストップ開発環境SDSoC上でC++プログラムを行い、打音検査用SOM AIチップのソフトウェアとハードウェアを開発します。ハードウェアはSoC型FPGA“Zynq”を搭載したスタータキットZedBoardです。



図15 本誌ダウンロード・サイトで提供するサンプル・プログラムのファイル構成

SDSoCで記述したサンプル・プログラムの全ファイル(図15)は、次のサイトで公開しています。

<https://toragi.cqpub.co.jp/tabid/927/Default.aspx>

srcフォルダにはサンプル・プログラム一式、dataフォルダにはSOMの学習データが入っています。ハードウェア化する関数を含む主な処理はすべて、main.cppにあるメイン関数から呼び出されます。

SDSoCを使った ワンストップ開発の流れ

● パソコンでC/C++を記述してZedBoardにSDから読み込ませる

Zynq向け統合開発環境SDSoC(ザイリンクス)をWindowsにインストールし、C/C++言語で、連載第1回と第2回で説明したSOMの構成をソフトウェアで記述します(図16)。

ハードウェア化する部分をC/C++の関数単位で指定します。ビルド・ボタンを押すと、ハードウェアとソフトウェアがコンパイルされ、必要なファイルが生成されます。

生成されたいくつかのファイルをSDカードにコピーし、Zynqの評価ボードZedBoardのSDカード・スロットに挿入して起動します。

今回使用したSDSoCのバージョンは“2016.4”です。ザイリンクス社は「2019年2月から、SDSoCを含めた同社の開発環境群をVitisプラットフォームに統合する」とアナウンスしています(本稿執筆中の2019年11月時点)。

SDSoC、Vitisの下にアーカイブされています。最終版のバージョンは“2019.1”ですが、これだと公開しているサンプル・プログラムの結果表示に1～2分かかるようです(理由は不明)。

● ハードウェアを最適化するときにはプラグマを使う
ハードウェアの動作を細かく指定したいときは、C/C++で記述した全体のソースコードに、プラグマと呼ばれる指示子を挿入します。今回使った主なプラ