FPGA人工知能のポテンシャルを探る

第3回　FPGA向けニューラル・ネットワークBNNのハードウェア化

ご購入はこちら

鈴木 量三朗

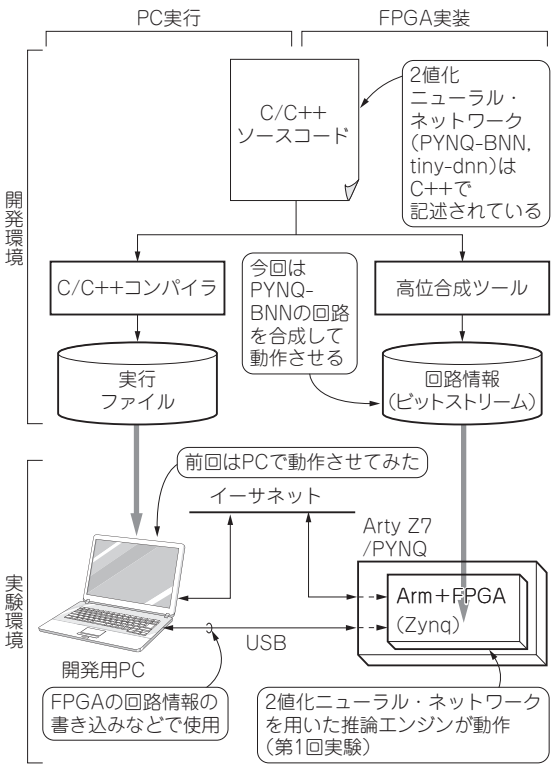

図1　2値化ニューラル・ネットワークBNNの実験環境
今回はPYNQ-BNNを対象に，C＋＋で記述されているコードから回路を合成してボードで動作させてみる

本連載ではFPGAで威力を発揮するといわれている2値化ニューラル・ネットワークBNN（Binarized Neural Networks）のポテンシャルを探ります（**図1**）．

今回は，ザイリンクスが提供しているBNNを詳しく見ていきます．

PYNQ-BNN[1]は，ザイリンクスのFPGA用に提供されているtiny-dnnを元にした，2値化ニューラル・ネットワークです．

Zynqが内蔵するArmプロセッサ（CPU）からは，メモリに張り付いたデバイスのように見えます（**図2**）．事前に学習済みのパラメータを設定し，さらに画像を設定することで，結果を得ることができます．

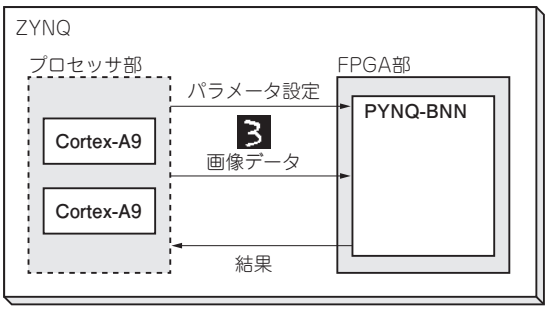

図2　PYNQ-BNNはArmプロセッサ（CPU）からメモリに張り付いたデバイスのように見える

使用するニューラル・ネットワークの構成

● 手書き文字認識MNISTのPYNQ-BNNの構成

lfc-pynqとcnv-pynqの2つのネットワークが用意されています．

MNISTの例で使われているlfc-pynqに注目します（**リスト1**）．`src/network/lfc-pynq/hw/top.cpp`がFPGAで動作するメイン・ルーチンです．

tiny-dnnの例では，C++の`<<`演算子をうまく使ってネットワークを表現していました．PYNQ-BNNではネットワークをストリーム化して並列処理させています．ブロックごとに分けて`hls::stream`というVivado HLSで使う特別な変数でネットワークを構成し，やはりVivado HLSのキーワードであるDATAFLOWを`#pragma`で追加しています．

`Mem2Stream_Batch`と`StreamingFCLayer_Batch`が処理の中心を担っています．`StreamingFCLayer_Batch`がニューラル・ネットワークの各層に対応し，C++の`template`引き数で層の設定をす