

オープンソースCPU 「RISC-V」の研究

第6回 RISC-V マイコン基板でAI手書き文字認識 [ご購入はこちら](#)

@msyksphinz

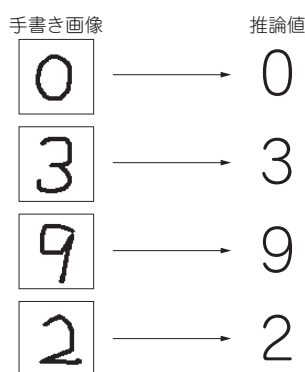


図1 MNIST (数字文字認識) は数字が書かれている画像ファイルを入力としてその数字が何であるかを推論する

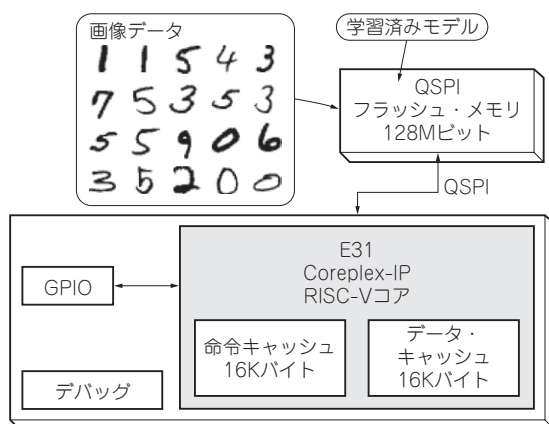


図2 MNIST をHiFive1で実行する仕組み
HiFive1が持つSPIフラッシュ・メモリに、数字の入った画像データと学習済みのパラメータを格納して、文字認識を実行する

HiFive1は手軽に試せるボードですが、Arduino互換ボードとしては非常に高速な動作周波数で動かせることが最大の特徴です。より高度なアプリケーションや、たくさんのデータを処理させるアプリケーションを実行できます。ここでは、HiFive1を使って、少し複雑なプログラムを動かしてみます。

機械学習の最も基本的なプログラムである手書き文字認識 (MNIST) を動作させてみます (図1)。

実験すること… AI手書き文字認識 MNIST

HiFive1は現状ではオペレーティング・システム (OS) を使用できず、外部の容量の大きなSRAMも接続できないため、できることは限られます (Arduinoボードとして発売されているボードなので、当たり前かもしれないが…)。

MNISTは学習データ、測定データが巨大ですが、HiFive1に搭載されているフラッシュ・メモリは128Mビットと比較的大きいので、何とかかなりそうで

す。またMNISTは浮動小数点を使わず、整数演算だけで実行できます。ハードウェア浮動小数点演算回路を持っていないHiFive1でも何とかできるのではないかと考えました (図2)。

今回構築した、HiFive1でニューラル・ネットワークを動作させるシステムの概要を図3に示します。MNISTをはじめとするニューラル・ネットワークのアプリケーションは、ネットワーク内の最適なパラメータを学習するための「学習」フェーズ、最適化されたパラメータを使い実際のデータを入力して学習結果をテストする「推論」フェーズに分けられます。学習のフェーズと最適化のパラメータの計算はx86上で実行し、推論の部分のみHiFive1で実行しました。

HiFive1のSPIフラッシュ・メモリに、学習済みのパラメータ、推論フェーズのテストに使用する画像、さらに画像の認識結果の答え合わせをするためのラベル・データを格納します。これらのデータを利用してニューラル・ネットワークを動作させ、答え合わせを行って、正しく動作していることを確認します。結果やデバッ

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」ではパークレイ以外のさまざまなコアも紹介しています。