微分可能ニューラルコンピュータ向き ハードウェアマルチコアアクセラレータの検討

1215082　齋藤 あかね　【 コンピュータ構成学研究室 】

A Study on Hardware Multicore Accelerator for Differentiable Neural Computer

1215082　Akane SAITO　【 Advanced Computer Engineering Lab. 】

1　はじめに

近年の DNN(Deep Neural Network) 技術の進展に伴って，IoT 機器へも AI 技術を導入する研究が活発になっている．中でも自動運転や AI アシスタント等への需要から，IoT エッジデバイスで DNN を低消費電力および高速に実現するためのアクセラレータの研究開発が活発化している．

現状では，画像処理向きの CNN に特化したアクセラレータが多く開発されている ([1], [2] など)．そこで本研究では Google Deep Mind 社より提案された微分可能ニューラルコンピュータ DNC(Differentiable Neural Computer) [3] に着目した．DNC は文章やグラフなど複雑なデータ構造の学習が可能な DNN の 1 つであるが，このアルゴリズムに対応したアクセラレータは現時点では見当たらない．

本研究では，パイプライン実行可能な積和演算や複数種類の非線形関数を含む命令セットに加え，時系列データを考慮したメモリアドレッシング方式を導入したシングルコアアーキテクチャを採用している．さらに，このコアを複数接続した DNC 向きハードウェアマルチコアアクセラレータの提案を行う．

2　微分可能ニューラルコンピュータ (DNC)

DNC の構成は図 1 のように表される．長期短期記憶 LSTM(Long Short-term Memory) ニューラルネットワークと構造体データを記憶する外部メモリから構成される．外部メモリは Read/Write ヘッドを介しての読み書きされる．LSTM ネットワークの出力は，メモリ操作のためのパラメータ生成にも使われ，コントローラネットワークとしての役割を担う．

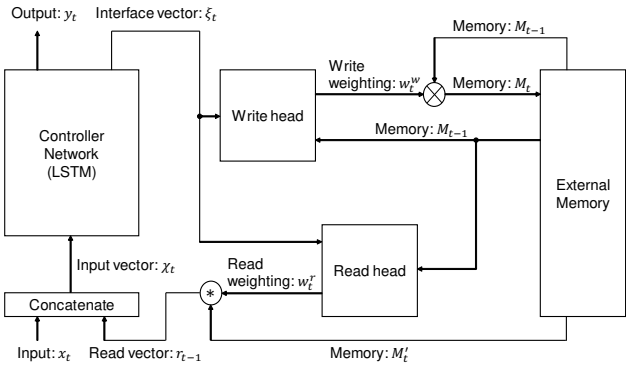

図 1　DNC 構成概略図

DNC 向きアクセラレータを設計するためには，LSTM 演算機能に加えて，図 1 中に示すような読み取り重み ($\mathbf{w}_t^r$)，書き込み重み ($\mathbf{w}_t^w$) を生成する演算に対応する必要がある．

3　アクセラレータアーキテクチャ

3.1　シングルコア

コアは，図 2 のような 4 段のパイプライン構成をとる．試作コアの仕様を表 1 に示す．

表 1　コアの設計仕様

データ形式	16bit fixed-point (Q6.10)
命令メモリ	35 bit × 256 words
データメモリ	16 bit × 4,096 words × 2
LUT	16 bit × 256 words × 2

積和演算命令の実行時には，演算対象ベクトルを DM-X と DM-W から連続的に読み出し，乗算結果をアキュムレータ (acc) に順次積算する．よって，演算対象のベクトルサイズ n に対して，$n+3$ クロックサイクルで実行が可能である．このとき，LSTM および DNC で扱うデータは時系列データであるため，時系列レジスタ (seq) を導入し，命令実行時にこの値を適宜更新して DM-X および DM-W 内のベクトルへアドレッシングする．シグモイド関数や $tanh$ 等の非線形関数の適用には，LUT を用いる．LUT サイズ削減のために，零近辺のみ ($|x| < 7$) を参照する方式とした．

3.2　マルチコア構成法

マルチコア構成では，演算対象ベクトルを分割し各コアに分配して演算を行う．よってコア間通信コストを考慮しない場合，演算に要するクロックサイクル数は $n+3$ から $(n+3)/C$ へ短縮される (C:コア数)．実際にはコア間通信コストや送受信同期待ちが発生するため，

スケーラビリティは低下する．

本研究では，マルチコア構成法として図 3 に示すように，数コアで 1 つのクラスタを形成し，これらのクラスタを相互に接続した階層型構成を検討する．

コア間のデータ送受信を行うために，SEND 命令および RECEIVE 命令を追加した．SEND 命令を実行すると，データメモリから該当するデータが読み出され，MA ステージを通過した後に送信先コアへ送信される．送信先コアで RECEIVE 命令を実行すると，相互結合網経由で送信されてきたデータが自コア内のデータメモリへ書き込まれる．これらの送受信命令がなるべく遅滞なく同期できるように，命令スケジューリングを実施することによって，スケーラビリティを維持する方法を採用した．

4　評価

提案マルチコアアクセラータの性能評価のために，多様なエッジデバイス用途に適用可能な FPGA 実装を想定した．本評価では，Intel 社製 FPGA Cyclne IV を想定し，開発ツール Quartus Prime を用いた論理合成により，回路規模（LE 数）を評価した．表 2 に，コア数に対する回路規模を示す．

表 2　マルチコア構成の回路規模 (型番 EP4CE6U14l7)

コア数	1	2	16
LE 数	779	1,549	12,224
メモリ量 (bit)	148K	296K	2,371K

マルチコアにおけるメモリ bit 数は，シングルコアにおける使用メモリ bit 数 × コア数となった．対して，LE 数は，シングルコアにおける使用 LE 数 × コア数よりも大きい値となった．これは，コア数が増えるに従い，ユニット内のマルチプレクサ (図 3 中の data_selector) の回路規模が大きくなるためである．

表 3　マルチコア構成の予備評価

コア数	1	2
clock cycle 数	69	51

本評価では，提案マルチコア構成用命令スケジューラ，および，アセンブラを開発し，任意のコア数で任意の規模の LSTM ネットワークが実行可能な開発環境を試作している．この環境で使用する各コア用プログラムのテンプレートを作成するために，入力層 1，LSTM-Block×2，出力層 1 の LSTM 計算を行うプログラムを作成して動作検証を行った．

実行に要した clock cycle(cc) 数を表 3 に示す．動作検証用プログラムでは，コア数が 1 から 2 になると，実行 cc は 18 短縮されたが，これは，積和演算命令を並列に実行することで短縮された cc と，各コアで行う送受信命令を実行することで発生する cc の差分となる．この差分は命令スケジューリングを最適化して並列度を上げることで，改善される可能性がある．

5　まとめ

本研究では，DNC の高速化を目的とした，ハードウェアマルチコアアクセラレータの検討を行った．

DNC の計算に必要な LSTM 演算を行うための命令セットを検討し，パイプライン実行可能な回路構成をとった．更に，送受信命令および相互結合網を実装し，マルチコア構成による高速化を検討した．

今後は DNC 演算に必要となる除算やソートの高速化方法を検討し，提案回路のみで DNC 演算を完了できるように機能を拡張していく必要がある．

参考文献

[1] A. Yazdanbakhsh, et al., “GANAX: A Unified MIMD-SIMD Acceleration for Generative Adversarial Networks, ” ISCA ’18, pp.650–661, 2018.

[2] L. Bai, et al., “A CNN Accelerator on FPGA Using Depthwise Separable Convolution, ” IEEE Transactions on Circuits and Systems II: Express Briefs, Vol. 65 , Issue 10, pp.1415–1419, 2018.

[3] A. Graves, et al., “Hybrid computing using a neural network with dynamic external memory,” NATURE, Vol. 538, Issue 7626, pp.471–476, 2016.

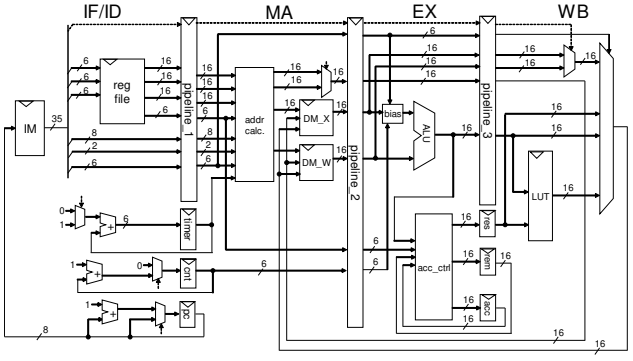

図 2　アクセラレータのシングルコア構成

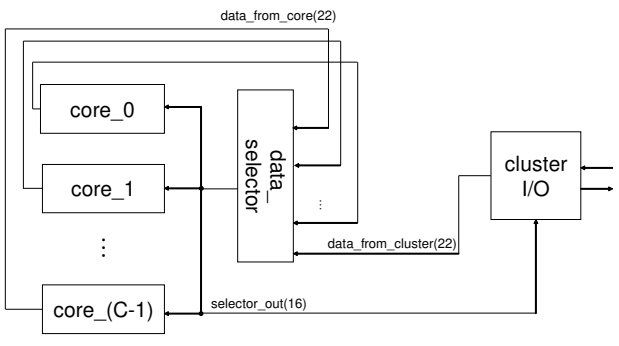

図 3　クラスタ構成 (コア数 C の場合)