MN-Core™ 2 White Paper

2023-11-12

Preferred Networks

はじめに

AI, HPCのワークロードが要求する計算資源量は増加の一途を辿っています。さらに、Large-language model (LLM) に代表されるような、従来のモデルよりはるかに複雑かつ大規模であり、莫大な計算資源量を必要とするモデルが、実際のプロダクトに組み込まれ、実世界で使用されることを鑑みると、必要となる計算資源量を実現するためには、非常に高速なアクセラレータの存在が鍵になると言えます。Preferred Networksでは、これらの計算需要に応えるために、MN-Coreという独自のアクセラレータを開発しています。

第一世代のアクセラレータであるMN-Coreでは、このアーキテクチャの電力効率が極めて高いことを実証しました。MN-Coreは、スーパーコンピュータの演算性能を競うランキングであるTop500に付随するGreen500において、非常に高い成績を収めています。以下はMN-Coreにおける消費電力あたりの性能 (GFlops/W) と、その時の順位の表です。また、さまざまな実際のAIワークロードにおいても、既存のGPUを大幅に超える性能が実証されています。

	消費電力性能	順位
Jun. 2020	21.11 GFlops/W	#1
Nov. 2020	26.04 GFlops/W	#2
Jun. 2021	29.70 GFlops/W	#1
Nov. 2021	39.38 GFlops/W	#1

MN-Core 2は、Preferred Networksが開発した第二世代のアクセラレータであり、本文書で解説するように、第一世代のMN-Coreと比較すると、メモリ帯域の増大と大幅なコストダウンを実現しています。また、第二世代MN-Coreでは既存のAIワークロード向けのソフトウェアに加え、汎用開発環境を用意し、AI以外のHPCワークロードにも対応を開始しました。これにより、MN-Core 2の持つ高い演算性能を、既存の多様なHPCアプリケーションに活用することが可能になります。

MN-Core 2 概要

本章では、MN-Core 2の概要について解説します。

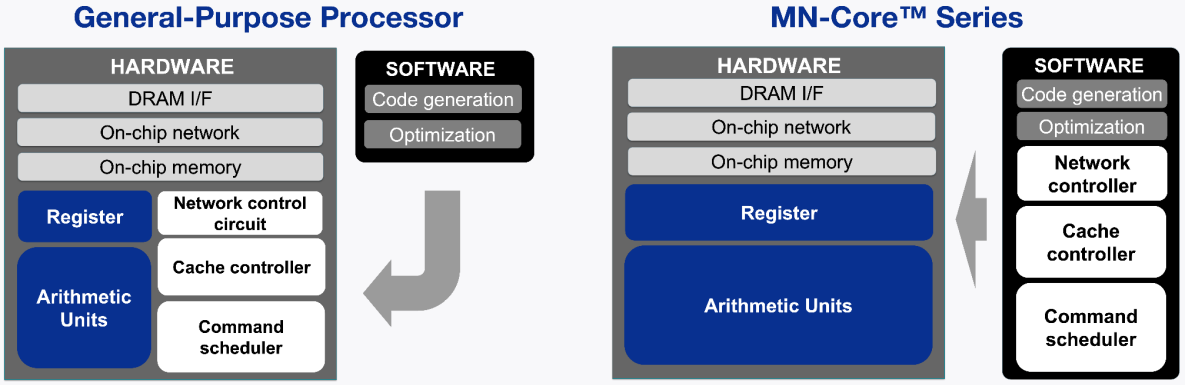
MN-Core 2について

MN-Core 2は、Preferred Networksが開発した第二世代アクセラレータです。MN-Coreシリーズは以下のような特徴を持ちます。

- 高いシリコン利用効率を実現する
- ソフトウェアによる最適化を前提とすることで、高い実行効率を実現する

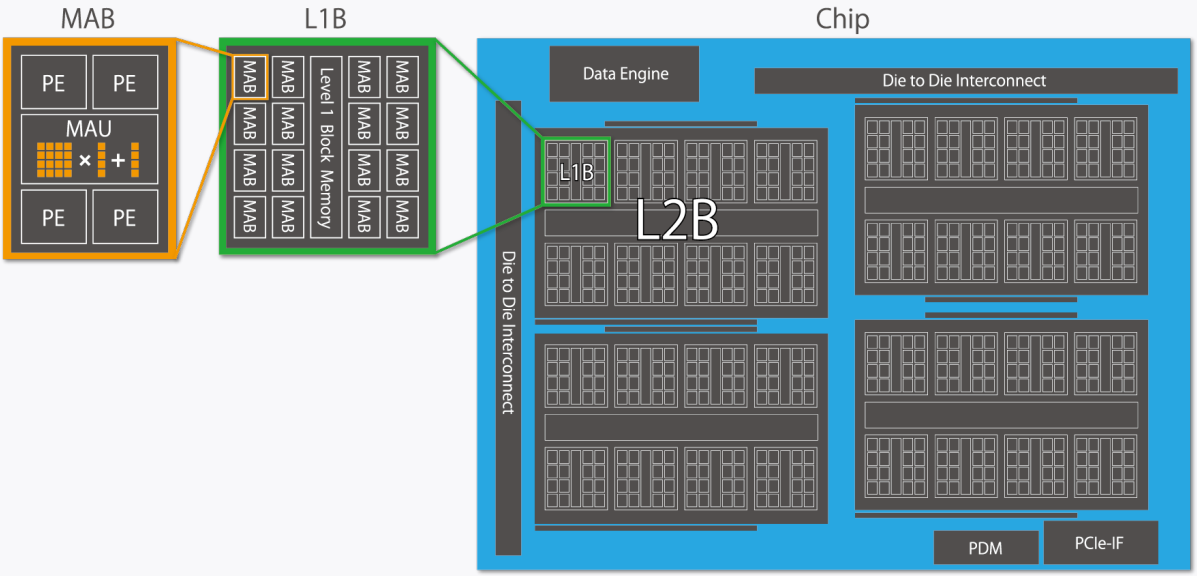
今日の多くのプロセッサは、実際のシリコン上で演算器の占める面積の割合が非常に低いという特徴を持ちます。これは、既存のコードベースに対して、変更量を最小限に留めることによって、ユーザーのアーキテクチャ移行を容易にするためのハードウェア支援が潤沢であると言い換える事もできます。一方で、MN-Coreシリーズでは、シリコンの面積に対して、演算器の占める割合を非常に高める設計を取っています。以下は、既存のプロセッサとMN-Coreシリーズの設計思想の違いを示したものです。また、MN-Core 2における、トランジスタ数に占める演算器の割合は 約 7 % となっています。これは、他社のプロセッサと比較しても非常に高い数字となっています。(数字はPreferred Networks独自調べに依ります)

製品名	トランジスタ数に占める演算器の割合
MN-Core 2	7.4 %
GPU N	1.7 %
Accelerator P	2.4 %
CPU F	1.3 %
CPU I	0.8 %



MN-Coreシリーズでは、シリコン上におけるハードウェアによる制御ロジックを極力削減するアーキテクチャを取っているため、ソフトウェアによる最適化が非常に重要となっています。MN-Coreシリーズでは、従来のプロセッサとは異なり、アクセラレータ上の各Processing Element (PE) は、それぞれのプログラムカウンタや命令デコーダを持ちません。すべてのPEは、完全に同期して動作し、ホストCPUで生成された命令列をホストから直接受け取って動作します。これらにより、今日のアクセラレータ上でしばしば発生する、アクセラレータ上の各演算単位が非同期に動作することによるワークインバランスとそれに伴う同期コストをゼロにし、さらににインストラクションキャッシュなどの命令供給系で発生するボトルネックや、アウトオブオーダー資源の不足による効率低下といった問題を解決します。

MN-Coreシリーズのアーキテクチャ概要図を以下に示します。



最小単位は PE であり、4PE が1つのMAU をシェアし、MAB を構成します。MAB 16個がL1B、L1B 8個が L2Bを構成しています。

以下に、MN-Core 2のスペックを記載します。

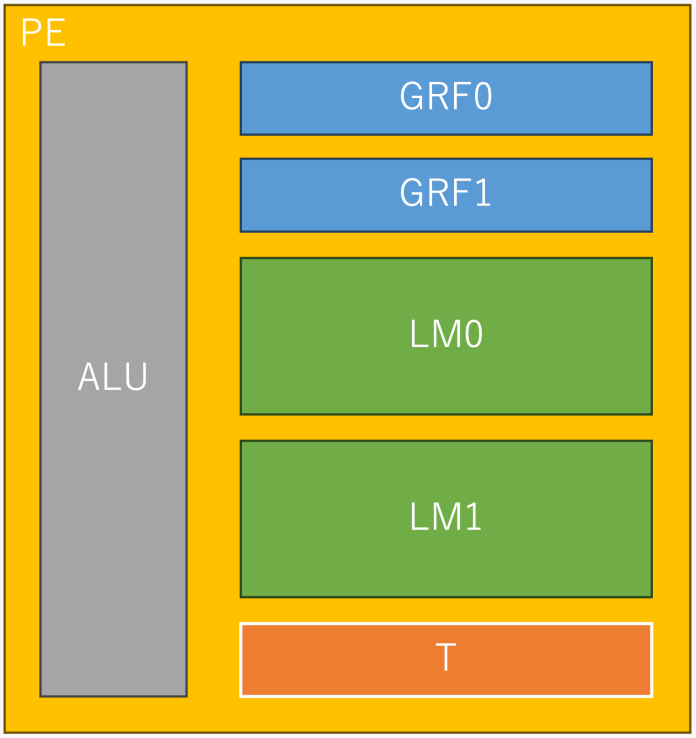
製造プロセス	TSMC N7
面積	550 mm2
トランジスタ数	22 B
動作クロック	750 MHz
PE数	4096
Peak FLOPS @ fp64	12 TFlops
Peak FLOPS @ fp32	49 TFlops
Peak FLOPS @ TF32	98 TFlops
Peak FLOPS @ TF16	393 TFlops
Power Consumption	330 W (Design value)
Energy efficiency (fp64)	37.24 GFlops/W
Energy efficiency (fp32)	148.9 GFlops/W
Energy efficiency (TF32)	297.9 GFlops/W
Energy efficiency (TF16)	1192 GFlops/W

以降では、MN-Core 2のアーキテクチャについて解説します。

Processing Element について

Processing Element (PE) は、データを保持するための汎用レジスタファイル (GRF)、ローカルメモリ (LM) を持ちます。これらのメモリから読みだしたデータをALUなどの演算器に入力します。演算結果出力はこれらのメモリへ保存されます。メモリの他に一時データを保持するためのTレジスタ、演算結果のフラグを保持するMaskフラグレジスタなどを持ちます。また、PEで処理可能な演算は一般的なもののほかにもReLUなどがあります。これ以外にも上位階層メモリL1Bとのデータのやり取りなどを行う機能を持ち、これらの処理はPE命令によって制御されます。

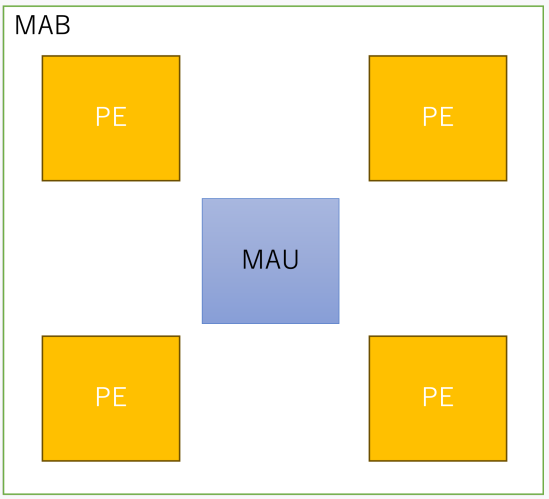
PEの概略図を以下に示します。



MAB / MAUについて

Matrix Arithmetic Block(MAB) は、4つのPEと、1つのMatrix Arithmetic Unit(MAU) によって構成されています。

MABの概略図を以下に示します。

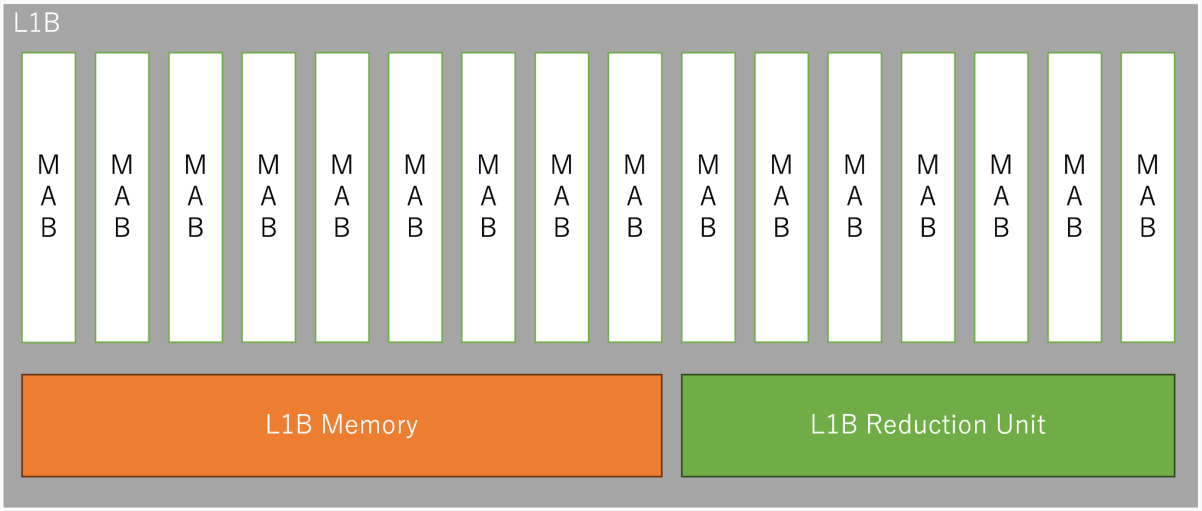


MAUはPEから入力データを受け取り演算処理を行い出力データをPEに戻します。また、MAUは行列演算に用いる専用の行列レジスタを持っており、これは行列ベクトル積で使う行列を保持します。また、行列レジスタは2面存在しているため、行列レジスタ1に書き込みながら、行列レジスタ2を用いて行列演算を行うことが可能です。

MAUでは以下の処理を行うことができます。

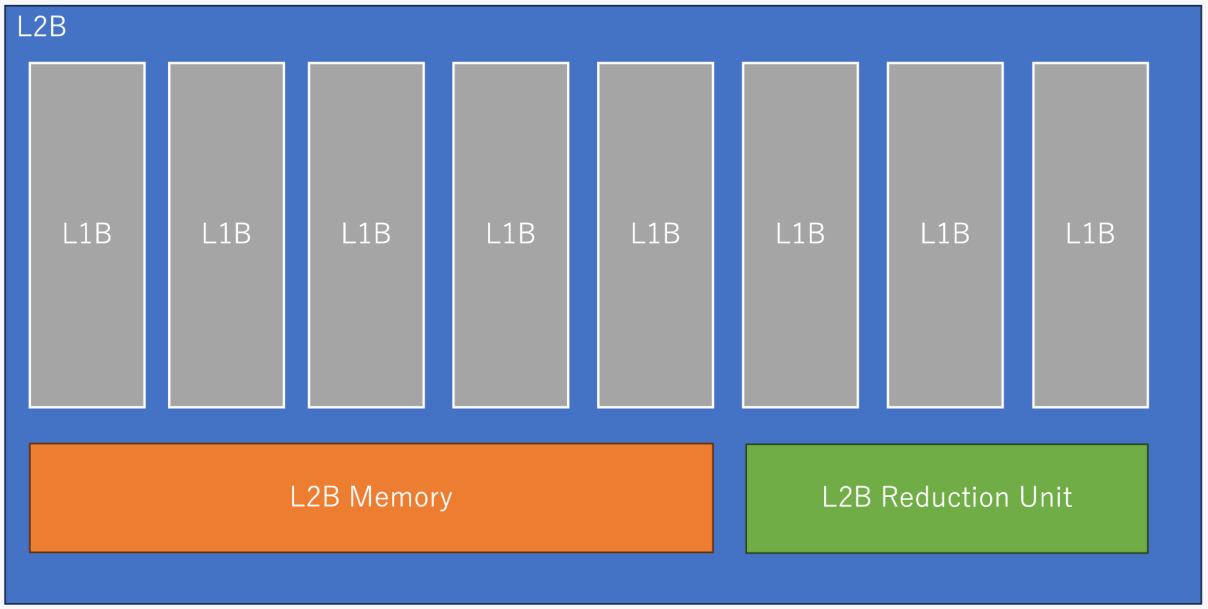
- 積和演算
    - ベクトル積和 (A x B + C) :半精度、単精度、倍精度
    - 行列積和 (A x 行列レジスタ + C) :半精度、疑似単精度、単精度、倍精度
- 行列レジスタ書き込み :半精度、疑似単精度、単精度、倍精度
- 行列レジスタ転置読み出し :半精度、疑似単精度、単精度、倍精度

L1Bについて



Level-1 Bloadcast Block (L1B)には、16個のMABと1個のL1 Broadcast Memory(L1BM) があります。L1BMとMABは、L1BMからMAB (PE) への放送データ転送、個別・縮約データ転送、分配(結合) データ転送などの多様なデータ転送を行うことができます。これらのデータ転送は、PEと同様にPE命令によって制御されます。これらの転送モードにより、階層キャッシュによるマルチプロセッサでは困難な、細粒度並列タスクの高い効率での実行を可能にしています。

L2Bについて



Level-2 Broadcast Block (L2B)には、8個のL1Bと1個のL2 Broadcast Memory(L2BM)があります。L1BMとPE/MABの間と同様に、L2BMとL1Bは、L2BMからL1Bへの放送データ転送、L1BからL2BMへの個別データ転送、縮約データ転送などの多様なデータ転送を行うことができます。これらのデータ転送も、PE命令によって制御されます。
最後に、L2Bは、ホストインターフェース及び外付けDRAMと、多様な方法でデータ転送を行うことができます。

MN-Core 2 搭載サーバー MN-Server 2

本章では、MN-Core 2搭載サーバー　MN-Server 2について解説します。

MN-Server 2は、MN-Core 2を8台搭載したサーバーです。以下にスペックを記載します。

CPU	Intel Xeon Platinum 8480+ x2
RAM	1 TB
Storage	960 GB(System) + 45 TB(Data)
NIC	NVIDIA ConnectX-6 100GbE Ethernet Adapter Dual port x 2
Accelerator	MN-Core 2 x 8

第一世代のMN-Core搭載サーバーが7Uだったのに対し5Uとなっており、MN-Server 2は高さ方向が削減されています。そのため、19インチラックに搭載可能なサーバー数が増加しています。また、MN-Server 2を19インチラック 42Uに6台搭載したものをMN-Pod 2と呼称します。MN-Podあたりの演算性能は、第一世代MN-Core・半精度浮動小数点数で比較すると2.25倍となっています。

	MN-Pod	MN-Pod 2	世代間性能変化
倍精度浮動小数点数(TFlops)	524.29	590	1.125
単精度浮動小数点数(TFlops)	2097.15	2359	1.125
疑似単精度浮動小数点数(TFlops)	N/A	4719	N/A
半精度浮動小数点数(TFlops)	8388.61	18874	2.25

MN-Core 2は、ユーザーの皆様のニーズに合わせ、オンプレミス、IaaS、SaaS等々、様々なモデルでの提供を予定しています。

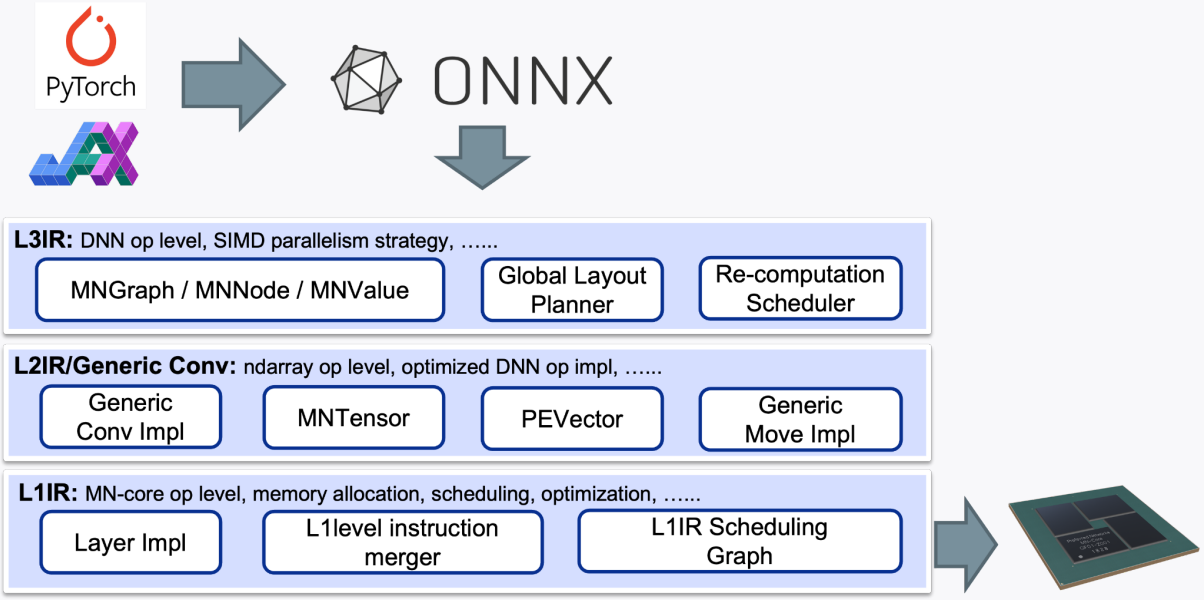
ソフトウェアスタックについて

本章では、MN-Core 2向けの2つのソフトウェアスタックについて解説します。

MN-Core AI ソフトウェアスタック

本節では、MN-Core 2向けのAIソフトウェアスタックについて解説します。

MN-Core AIソフトウェアスタックは以下のような構成を取っています。



AIソフトウェアスタックの設計にあたっては、以下の2点を重視しています。

- ユーザコードの大きな改変を可能な限り減らすこと
- MN-Coreシリーズの性能を引き出すこと

現在では、多くのユーザーが深層学習のためのフレームワークとしてPyTorchを利用しており、多くの研究開発資産がPyTorchを基盤としています。そのため、MN-Coreシリーズを利用する上でも、可能な限り既存のPyTorchコードに対して改変が少ないことが望ましいです。これを満たすために、MN-Core AIソフトウェアスタックの入力としては、ONNXを採用しています。いくつかの制約付きではありますが、ONNXにエクスポートできる形であれば、MN-Coreに対応することが可能です。

MN-Coreシリーズは非常に大きなSIMDプロセッサであるため、その性能を引き出すためには各演算コアに対して適切にデータを供給し続けることが非常に重要です。高い性能を引き出すためには、必要な計算とデータを適切にPE, MAB, L1B, L2Bに対してマッピングする必要があります。一方で、これらのハードウェアの構造を深く理解し、適切なコードを

ユーザーが記述することは、一般的には困難です。そのため、MN-Coreシリーズでは、特にAI向けソフトウェアスタックとして、データ構造からコード生成までを自動で行う仕組みを実装しています。これにより、ユーザーはメモリレイアウトや命令発行などのハードウェアの構造を意識することなく、MN-Coreシリーズの高い性能を得ることが可能です。

以下はPyTorchで記述した、CPUでResNet50を実行するコードです。

```python
import torch
import torchvision
import datasets

dataset = datasets.ImagenetDataset()
model = torchvision.models.resnet50(pretrained=True)
criterion = torch.nn.CrossEntropyLoss()
optimizer = torch.optim.Adam(model.parameters(), lr=0.001)
loader = torch.utils.data.DataLoader(dataset, batch_size=128)

def model_with_loss(inputs, labels):
  y = model(inputs)
  return criterion(y, labels)

for epoch in range(10):
  for idx, (inputs, labels) in enumerate(loader):
    optimizer.zero_grad()
    loss = train_step(inputs, labels)
    loss.backward()
    optimizer.step()
    print(f"iter {idx}: {loss}")
```

このコードを、MN-Coreで実行できるようにするには、以下のようにコードを改変します。

```python
import torch
import mncore
import torchvision
import datasets

dataset = datasets.ImagenetDataset()
model = torchvision.models.resnet50(pretrained=True)
criterion = torch.nn.CrossEntropyLoss()
optimizer = torch.optim.Adam(model.parameters(), lr=0.001)
loader = torch.utils.data.DataLoader(dataset, batch_size=128)

def model_with_loss(inputs, labels):
  y = model(inputs)
  return criterion(y, labels)

train_step = mncore.compile(model_with_loss, backward=True,
optimizer=optimizer)

for epoch in range(10):
  for idx, (inputs, labels) in enumerate(loader):
    # MN-Core用にコンパイルされたModelはForward,BackwardとOptimizer Stepを一緒に実
行します
    loss = train_step(inputs, labels)
    print(f"iter {idx}: {loss.cpu()}")
```

MN-Core 汎用ソフトウェアスタック

本節では、MN-Core 2向けの汎用ソフトウェアスタックについて解説します。

MN-Core 2はAIワークロードを非常に高速に処理することをターゲットとして開発されていますが、同様に、一部の汎用計算においても、その演算力を活用することができます。
汎用計算向けのソフトウェアスタックとして、OpenACCとOpenCL、それぞれのサブセットを開発中です。

OpenACC for MN-Core

OpenACCは、OpenACC Organizationにより提唱・標準化が行われている、並列計算フレームワークです。ヘテロジニアスなシステムの並列プログラミングを単純化するために設計が行われました。詳細については、https://www.openacc.org をご覧ください。

MN-Core 2では、OpenACCのサブセットをサポートします。以下のようなコードを
MN-Core 2上で実行することが可能になります。

```C/C++
void vecadd(...) {
  ...
  #pragma acc data copyin(a[0:1024], b[0:1024]) copyout(c[0:1024]) l2(a,b,c[8])
l1(a,b,c[1])
  {
  #pragma acc parallel
  #pragma acc loop independent
    for(int i=0; i<1024; i++) {
      c[i] = a[i]+b[i];
    }
  }
}
```

OpenCL for MN-Core

OpenCLは、Khronos Groupによって提唱・標準化が行われている、並列計算のためのクロスプラットフォームなAPIです。OpenACCと比較して、よりハードウェアに対する詳細な処理を記述することを可能とします。詳細は https://www.khronos.org/opencl/ をご覧ください。

MN-Core 2では、OpenCLのサブセットをサポートし、MN-Core 2上で動作するコードを
OpenCL C言語に近い形で記述することを可能にします。

Conclusion

本文書では、Preferred Networksが開発した、第二世代アクセラレータ　MN-Core 2について解説しました。

MN-Core 2は非常に高速であり、かつ低消費電力なアクセラレータです。MN-Core 2と、MN-Core 2を搭載するサーバーであるMN-Server 2, さらにMN-Server 2を集積したMN-Pod 2は、非常に高い面積あたり性能を実現しており、この性能は非常に革命的であると言えます。

また、MN-Core 2は様々な提供方法を予定しており、お客様のお手元のワークステーションに搭載するものから、お客様のデータセンターで運用すること、またクラウドサービスとしてIaaSやSaaSなどでもご活用いただけるように準備を進めています。MN-Core 2がもたらす計算力は、AIワークロード、HPCワークロードを問わず、皆様のワークロードの高速化に貢献するでしょう。

https://projects.preferred.jp/mn-core/

Revision History

- 2023-11-10 ver 0.1 Initial version