

ZeBu を用いた HPC/AI 向けアクセラレータ評価事例

PEZY Computing K.K.

石谷 太一

会社紹介 PEZY Computing K.K.

事業

□ HPC/AI 用アクセラレータチップ・システムの開発

- PEZY-SCシリーズ
- ZettaScalerシリーズ



会社紹介 PEZY Computing K.K.

□PEZY-SC シリーズを使ったアプリケーション開発

➤ 遺伝子解析システム

- 高速ヒトゲノム解析システム ZettaVEGA

- 国立遺伝学研究所にて設置、運用中

https://www.nig.ac.jp/nig/ja/2024/12/information_ja/pr20241226.html

- 高速メタゲノム解析システム PZLAST

➤ AI フレームワーク、生成AI対応

- PyTorch

- Gemma3, Llama3, Qwen2, StableDiffusion2, HuBERT, VisionTransformer

当社が開発したメニーコアプロセッサPEZY-SC3を搭載した高速ヒトゲノム解析システム「ZettaVEGA」における大規模実証実験につきまして

国立遺伝学研究所 生命情報・DDBJ センター（センター長 有田正規）と株式会社 PEZY Computing（代表取締役社長 高橋一夫）、株式会社 ゲノムアナリティクスジャパン（代表取締役 八谷剛史）は共同で、PEZY Computing社が開発した国産メニーコアプロセッサPEZY-SC3を搭載した高速ヒトゲノム解析システム「ZettaVEGA」の実証実験を行いました。ZettaVEGAではヒトゲノム解析で最もよく使用されるソフトウェアGATKのベスト・プラクティスをPEZY-SC3に移植し、メニーコアを最大限まで活かした超並列計算による高速化を実現しています。

詳細は国立遺伝学研究所のホームページをご覧ください。

「国産コンピュータ・プロセッサで実現したヒトゲノム解析の革新的な高速化と制約無しの完全公開」

https://www.nig.ac.jp/nig/ja/2024/12/information_ja/pr20241226.html

News Posted:2024.12.26 Last Updated:2024.12.26

国産プロセッサPEZY-SCシリーズが「PyTorch」に対応。ヘルスケア分野での生成AI利活用を推進。

自社開発の国産プロセッサ「PEZY-SC」シリーズにおいて、AIフレームワーク「PyTorch」への対応をいたしました。これにより、年末にリリース予定の第4世代大規模マイクロプロセッサ「PEZY-SC4S」上で、最先端のAIアプリケーションがより効率的に動作するようになります。現在、「PEZY-SC3」を4基搭載した当社のスーパーコンピュータ「ZettaScaler-3.0」では、以下の主要なAIフレームワーク、ライブラリ、モデルの動作を確認済みです。

フレームワーク・ライブラリ: PyTorch, Accelerate, DeepSpeed, Transformers, vLLM, Text Generation Inference, Diffusers

モデル: Gemma3, Llama3, Qwen2, StableDiffusion2, HuBERT, VisionTransformer

当社は長年にわたり培ってきた「医療用画像診断機器」や「ゲノム解析システム」の開発経験を活かし、ヘルスケア分野における生成AIの利活用を強力に推進しています。今後、「PEZY-SC4S」を搭載した次世代スーパーコンピュータ「ZettaScaler-4.0」において、様々なAIアプリケーションを稼働させ、医療現場における診断精度向上、業務効率改善、研究開発の加速に貢献してまいります。

News Posted:2025.06.06 Last Updated:2025.08.08

会社紹介 PEZY Computing K.K.

公開OSS

□PZBCM

- 共通モジュール集
- <https://github.com/pezy-computing/pzbcm>

□Veryl

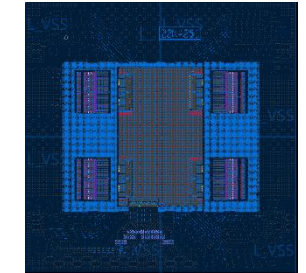
- SystemVerilog代替ハードウェア記述言語
- 次期プロジェクトで全面採用予定
- <https://veryl-lang.org>

□RgGen

- 制御レジスタ自動生成ツール
- PEZY-SC4sの制御レジスタは、全てRgGenで生成
- <https://github.com/rggen/rggen>



PEZY-SC シリーズ紹介



Package Layout

	PEZY-SC	PEZY-SC2	PEZY-SC3	PEZY-SC3s	PEZY-SC4s
Release	2014	2016	2020	2021	2026
Process	28 nm	16 nm	7 nm	7 nm	5 nm
Core	1024	2048	4096	512	2048
Performance	0.75 TFLOPS	4.1 TFLOPS	19.7 TFLOPS	2.0 TFLOPS	24.6 TFLOPS
Mem Bandwidth	154 GB/s	102 GB/s	1228 GB/s	614 GB/s	3277 GB/s
PCIe	Gen3 x 32	Gen4 x 32	Gen4 x 48	Gen4 x 4	Gen5 x 16
	Ranked 1st (2015-2016)	Ranked 1st (2017-2018)	Ranked 12th (2021)		

* Double Precision

PEZY-SC4s 紹介

PEZY-SCシリーズの4世代目

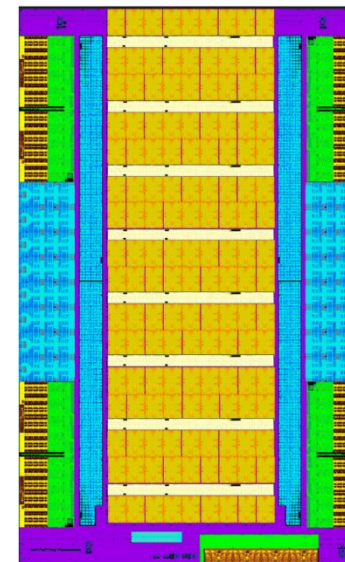
- 2026年リリース予定
- 2048個の演算コアを搭載
- AI機能用演算器 (BF16フォーマット) を新搭載
- 演算性能
 - 24.6 TFLOPS(倍精度)
 - 576 TFLOPS (BF16)

Spec

- Process TSMC 5 nm FinFET
- Clock 1.5 GHz
- Die Size 18.4 mm x 30.2 mm
- Gate Count 4.8 billion gates
- SRAM Cell 1.6 Gbits
- Memory HBM3/96 GB
- External IO PCI Express Gen5/x16

Hot Chips 2025にて発表を行う

- 資料
<https://www.pezy.co.jp/wp-content/uploads/2025/09/HC2025.PEZYComputing.NaoyaHatta.v06.pdf>



PEZY-SC4s

The Fourth Generation MIMD Many-core Processor with High Energy Efficiency and Flexibility for HPC and AI Applications

Naoya Hatta[†], Shuntaro Tsunoda, Kouhei Uchida,
Taichi Ishitani, Toru Koizumi, Ryota Shioya, Kei Ishii

[†] PEZY Computing, K.K.

1

発表内容

- ZeBu導入の経緯
- SC4s性能評価内容
 - SC4s性能評価環境
 - SC4s性能評価の流れ
- ZeBu導入結果
 - コンパイル結果
 - ベンチマーク結果
 - LLMアプリケーション実行結果
 - 致命的なバグの発見
- ZeBu導入時に遭遇した問題
- 次期プロジェクトへの課題
- まとめ

発表内容

□ ZeBu導入の経緯

□ SC4s性能評価内容

- SC4s性能評価環境
- SC4s性能評価の流れ

□ ZeBu導入結果

- コンパイル結果
- ベンチマーク結果
- LLMアプリケーション実行結果
- 致命的なバグの発見

□ ZeBu導入時に遭遇した問題

□ 次期プロジェクトへの課題

□ まとめ

ZeBu導入の経緯

全系での検証・評価ができていなかった

- シミュレーション
 - 実行速度が非常に遅く、全系での実行は現実的ではない
 - 規模の縮小や、IPのシミュレーション用モデルへの置き換えが必要
- FPGA
 - 全系がFPGAに入らず、規模削減が必要
 - 演算コア数・キャッシュの削減
 - HBM/PCIe IPの置き換えが必要

ZeBuを導入し、全系での性能評価を行う

発表内容

- ZeBu導入の経緯
- SC4s性能評価内容
 - SC4s性能評価環境
 - SC4s性能評価の流れ
- ZeBu導入結果
 - コンパイル結果
 - ベンチマーク結果
 - LLMアプリケーション実行結果
 - 致命的なバグの発見
- ZeBu導入時に遭遇した問題
- 次期プロジェクトへの課題
- まとめ

SC4s性能評価内容

期間: 2025年5月～2025年8月

使用ノード数: ZeBu Server 5を7ユニット、計28モジュール使用

ZeBuでPEZY-SC4s全体をエミュレーションし、その上で使用予定のアプリケーションを実行して性能評価を実施

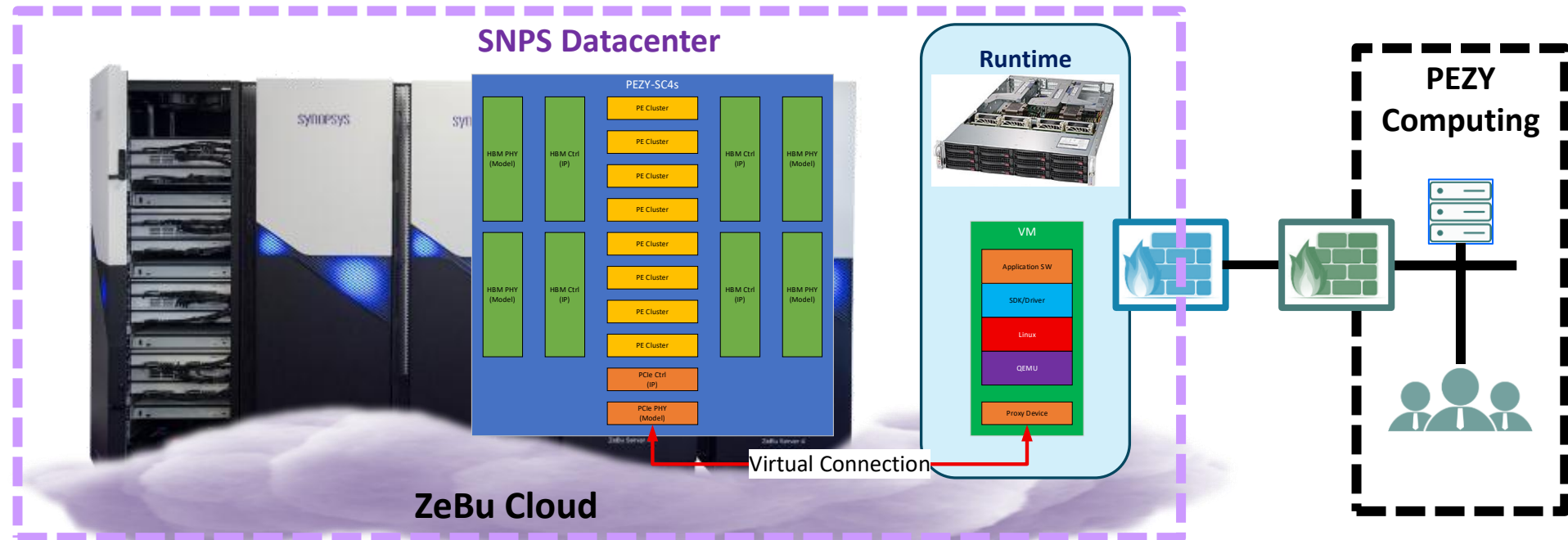
実行アプリケーション

- 基本動作確認
- 数値演算
 - HPL
 - DGEMM
 - BGEMM
- 生成AI関係
 - LLM 推論
 - LLM 学習
- 遺伝子解析関係
 - BWA-MEM
 - Haplotype Caller

SC4s性能評価環境

Virtual Hostを用いて、ZeBu上のPEZY-SC4sとホストOSを繋いだ環境をZeBu Cloud上に構築

- 弊社環境とZeBu Cloud間はVPNで接続
- 実際に使うドライバー/SDK/アプリケーションを使って評価できる



SC4s性能評価の流れ

3段階に分けて、評価を実施

1. 事前準備
2. 演算コアなし
3. 全系

SC4s性能評価の流れ(事前準備)

HBM PHY/PCIe PHYのZeBu用モデル組み込み

- コントローラIPはそのまま使用
 - HBM/PCIe とともに、PHY-コントローラ間はほぼ標準化されているので、モデルの組み込み自体は難しくなかった
 - ただし、HBMモデルの接続に関しては、後程問題が発覚
- シミュレーションで、モデルとコントローラIPが接続できたかを確認

コンパイル時間、使用ユニット数の見積もり

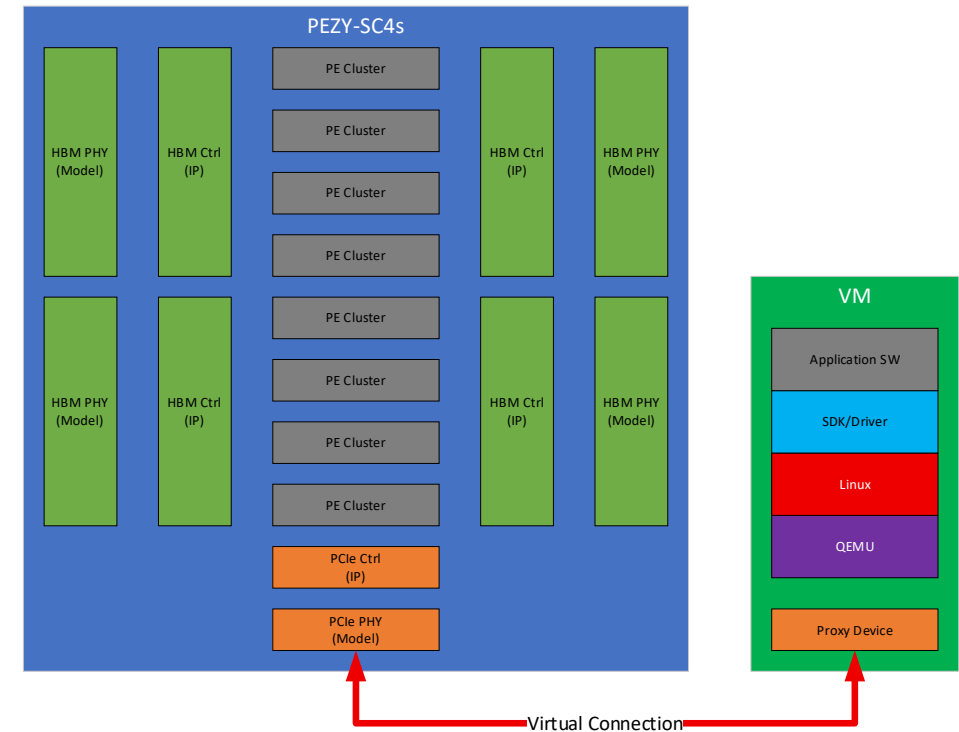
SC4s性能評価の流れ(演算コアなし)

演算コア部分は全て無効にし、HBM/PCIeのみを有効にした環境

本評価前に、QEMU上のLinuxからSC4sが制御できるかを確認

- ZeBu Cloud環境へのアクセス
- PCIeのリンクアップ
- HBMの初期化と、HBM への読み書き
- ドライバーのロード

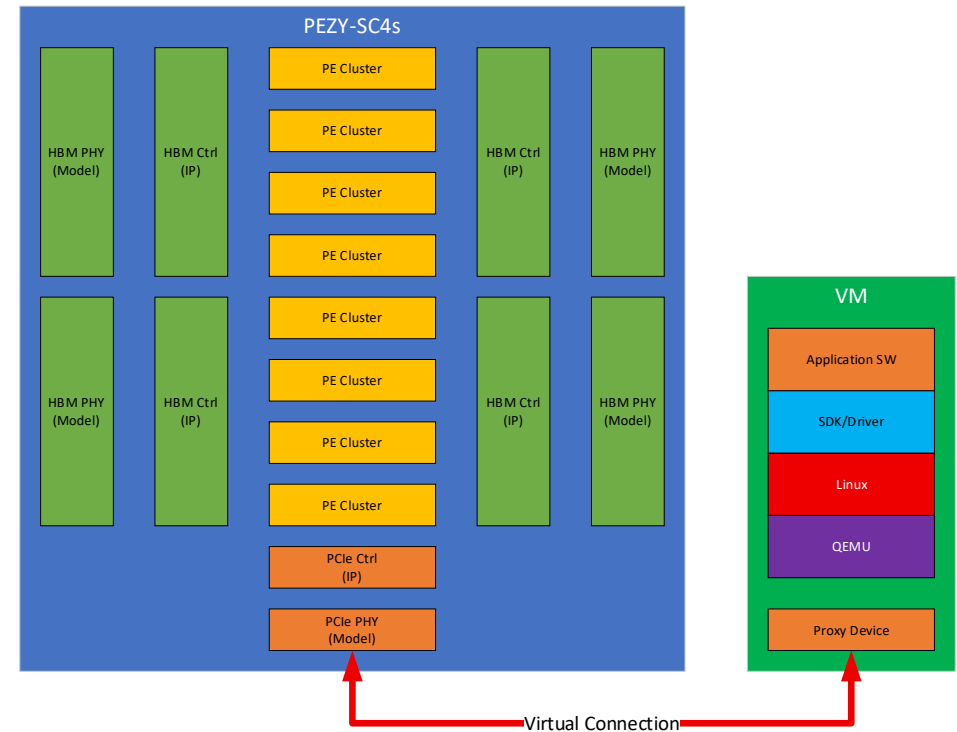
立ち上げに要した期間は1ヵ月ほど



SC4s性能評価の流れ(全系)

全ての演算コアを有効にした、実機と同等の環境

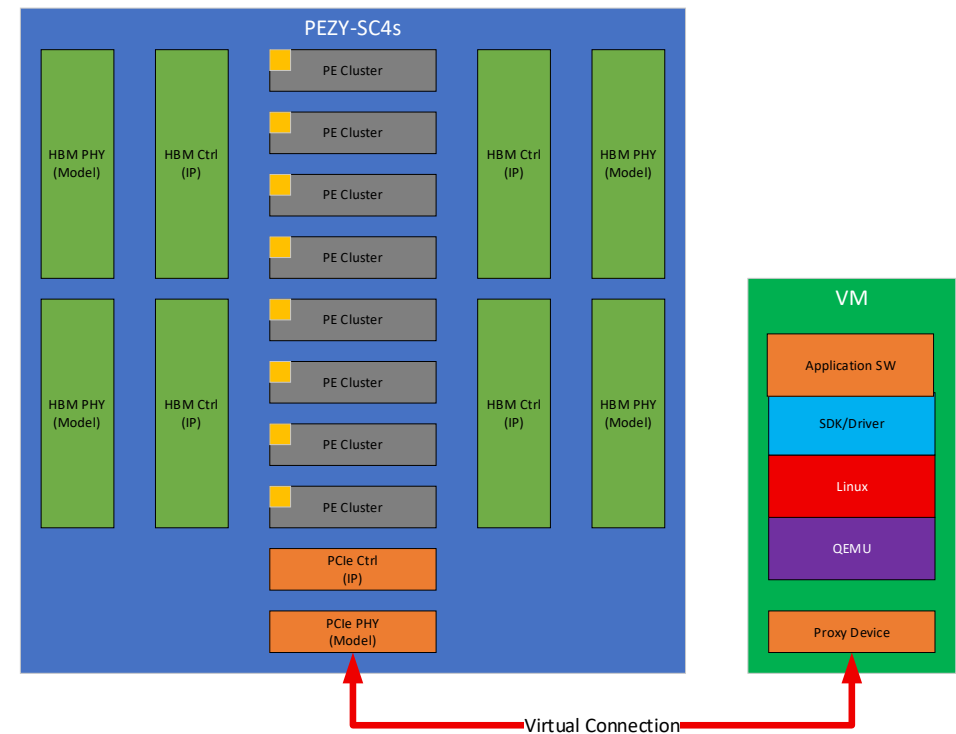
- アプリケーションを実行し、SC4sの評価を実施
 - 基本性能のベンチマーク
 - 演算性能、メモリ帯域など
 - アプリケーションのデバッグと最適化



SC4s性能評価の流れ(全系)

途中、HBM接続問題のデバッグのため、演算コア数を1/16に減らした環境も使用

- コンパイル時間削減のため



発表内容

- ZeBu導入の経緯
- SC4s性能評価内容
 - SC4s性能評価環境
 - SC4s性能評価の流れ
- ZeBu導入結果
 - コンパイル結果
 - ベンチマーク結果
 - LLMアプリケーション実行結果
 - 致命的なバグの発見
- ZeBu導入時に遭遇した問題
- 次期プロジェクトへの課題
- まとめ

ZeBu導入結果

- ✓ ZeBu上でベンチマークやアプリケーションを実行し、PEZY-SC4sの性能を評価できた
 - ✓ 設計通りの性能が出ることを確認した
- ✓ アプリケーションのデバッグや最適化を実機無しで、迅速に行うことができた
 - 従来、アプリケーション実行中に見つかった問題のデバッグは困難だった
 - シミュレーションでは、実行時間が長すぎて、デバッグできなかった
 - 実機やFPGAでは、観測性の低さで、デバッグが困難
 - ZeBuでは、実行時間が速く、波形も取得できるので、迅速にデバッグをすることができた
- ✓ 致命的なバグが見つかった

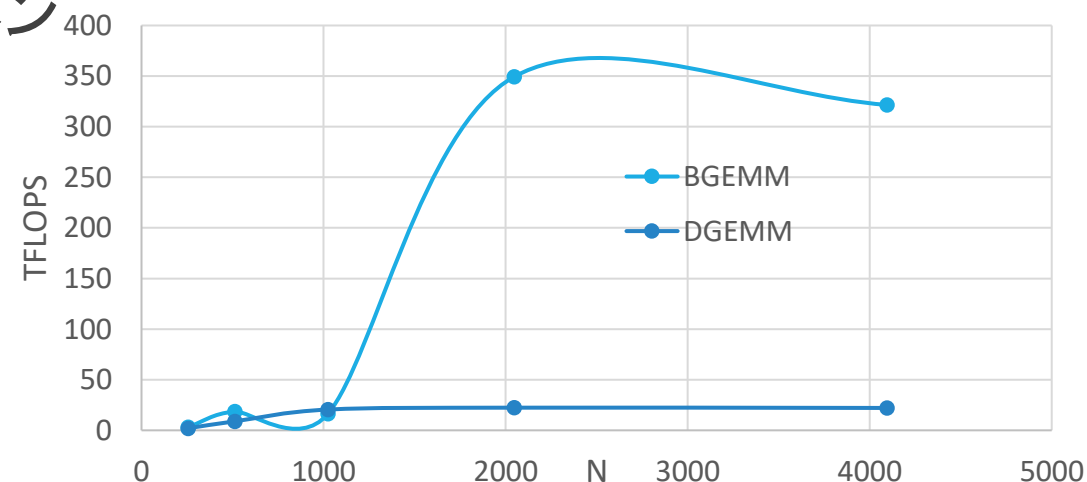
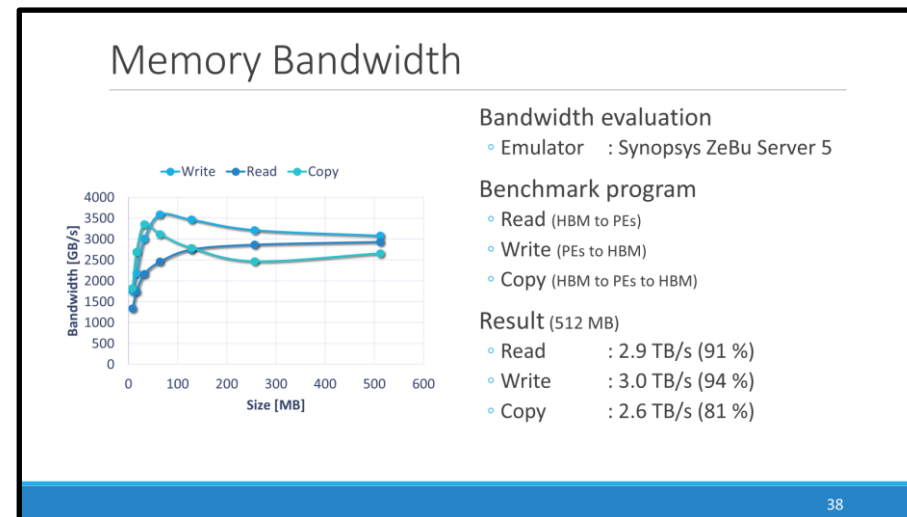
コンパイル結果

演算コア数	規模(面積比、 対全系)	使用モジュール数	コンパイル時間 [Hour]*	CPU	メモリ
なし	61 %	7			
1/16	63 %	8	45	EPYC 7702P 64core x 2	1 TB
全系	100 %	27	95	EPYC 7702P 64core x 2	2 TB

* 実行マシンが異なるので単純に比較はできない

ベンチマーク結果

- 全演算コアからのHBMへのメモリ帯域
 - Read (HBM to PEs) : 2.9 TB/s (91 %)
 - Write (PEs to HBM) : 3.0 TB/s (94 %)
 - Copy (HBM to PEs to HBM) : 2.6 TB/s (81 %)
- 行列演算
 - BGEMM: 350 TFLOPS
 - DGEMM: 22 TFLOPS
- Hot Chipsでの発表資料は、ZeBuでのベンチマーク結果をもとに作成



LLMアプリケーション実行結果

大規模言語モデル(LLM)をHBM上に展開し、実行できることを確認

モデル	パラメータ数	出力 [token/s]*1	実行時間[s]*2
Llama3.2	1B (BF16/1.8 GB)	857	408
Llama3.1	8B (BF16/14.9 GB)	324	1026
Llama3.1	70B (FP8/65.1 GB)	67	4703

*1 ZeBuでの結果から換算した、実機の出力

*2 エミュレーションの実行時間

致命的なバグの発見

SC4s評価中に、ECOが必要な重大なバグが見つかった

- PCIeコントローラIPとユーザー論理間の非同期FIFOで、リセットの接続間違いがあり、片側にしかリセットが入らず、バスが固まってしまう

見つかった経緯

- ドライバーロード時にユーザー論理をリセットする処理があり、その際に見つかった
 - ドライバーの開発はFPGAを用いて行っていたが、FPGA向けにPCIeコントローラIPを差し替えていたため、発現しなかった
- ✓ 使用するソフトウェアを実際のRTL上で実行できるZeBuだからこそ、検証の不備を発見できた

発表内容

- ZeBu導入の経緯
- SC4s性能評価内容
 - SC4s性能評価環境
 - SC4s性能評価の流れ
- ZeBu導入結果
 - コンパイル結果
 - ベンチマーク結果
 - LLMアプリケーション実行結果
 - 致命的なバグの発見
- ZeBu導入時に遭遇した問題
- 次期プロジェクトへの課題
- まとめ

ZeBu導入時に遭遇した問題

正しくコンパイルできないSystemVerilog記述があった

- 引数が別の関数呼び出しの戻り値になっている関数呼び出しを使ったinterface接続が、未接続になってしまう
- 中間変数を介して、関数呼び出しを分けることで回避できた
- 後日、コンパイラも修正された

修正前

```
else begin
    remap_if[2].put_command(remap_if[1].get_command());
end
```

修正後

```
else begin
    remap_command = remap_if[1].get_command();
    remap_if[2].put_command(remap_command);
end
```

ZeBu導入時に遭遇した問題

HBMモデルの組み込みに関する問題がいくつかあった

1. HBMモデルに与えるパラメータ間違い

- シミュレーションで動作確認した際は、動いているように見えた
 - シミュレータ上とZeBu上とで、HBMモデルの挙動に違いがある模様
- パラメータの修正で解決

2. 使用するHBMモデルの選択間違い

- 勘違いを起こしやすいモデル名で、また、ドキュメントの表記が崩れて読み難くなっていた
- モデルを選択しなおすことで解決

3. HBMモデル接続間違い

- 使用している他社製HBM IPのPHY-コントローラ間接続に、規格に準拠していない部分があった
- このため、モデルとの接続に試行錯誤が必要になった

発表内容

- ZeBu導入の経緯
- SC4s性能評価内容
 - SC4s性能評価環境
 - SC4s性能評価の流れ
- ZeBu導入結果
 - コンパイル結果
 - ベンチマーク結果
 - LLMアプリケーション実行結果
 - 致命的なバグの発見
- ZeBu導入時に遭遇した問題
- 次期プロジェクトへの課題
- まとめ

次期プロジェクトへの課題

- ZeBuコンパイラのSystemVerilog記述サポート品質向上
 - 正しくコンパイルできないSystemVerilog記述があった
 - 問題になった記述を使っている範囲が狭かったので、今回は何とか対応できた
 - エミュレーション結果からコンパイラの問題を追うのは難しいので、コンパイラのSystemVerilog記述サポート品質向上に期待
- コンパイル時間の短縮
 - 弊社環境上で、全系のコンパイルに約4日間かかった
 - 開発当初から導入するには、コンパイル時間短縮は必須
 - コンパイルの並列度を上げれば短縮可能との事なので、コンパイル環境の増強が必要

次期プロジェクトへの課題

➤ アプリケーション実行TATの短縮

- LLMアプリケーション実行時に、モデルパラメータをHBMに展開する必要がある、これに数時間かかっていた
- デバッグの効率化のためには、実行TATの短縮が必要
- CHECKPOINT/RESTARTや、HBMモデルへの直接の書き込みを使って、パラメータ転送を省略できるかを検討

➤ ZeBuハードウェアの効率的な使用方法の検討

- 28モジュールを3ヶ月間占有したが、実際に使ったのは1ヶ月ほど
 - デバッグや、コンパイルに使う時間の方が多い
- 共同利用や都度貸しなど、契約面も含め、効率的な使用方法を検討する必要がある

発表内容

- ZeBu導入の経緯
- SC4s性能評価内容
 - SC4s性能評価環境
 - SC4s性能評価の流れ
- ZeBu導入結果
 - コンパイル結果
 - ベンチマーク結果
 - LLMアプリケーション実行結果
 - 致命的なバグの発見
- ZeBu導入時に遭遇した問題
- 次期プロジェクトへの課題
- まとめ

まとめ

- ZeBu上でPEZY-SC4sをエミュレーションし、実際に使うアプリケーションを使った性能評価を行うことができた
- アプリケーションのデバッグや最適化を、実機無しで行うことができた
 - 実行速度と観測性の両立
 - 事前に動作確認や最適化をできたので、実機立ち上げ時間短縮も期待
- 実アプリケーションを実行することで、検証の補助になった
 - ECOが必要なバグや、シミュレーションシナリオの不備が見つかった

実アプリケーションを実機無しで実行できる恩恵は非常に大きい