MIPS ARC NPX6 NPU IP Processors: AI Data Compression Option

(NPX6 NPU IP, NPX6 FS NPU IP)

Highlights

- Enhanced data efficiency
- Improved computational performance
- Compliance and standardization
- Optimized memory management
- High bandwidth utilization
- Resource optimization
- iDMA
  - Conversion of INT4 to INT8
  - Conversion of FP8 formats to BF16 or FP16
  - Conversion of MX formats to BF16
  - Double bandwidth to vector memory
- oDMA
  - Conversion of INT8 to INT4 while saturating
  - Conversion of BF16 or FP16 to FP8
  - Conversion of BF16 to OCP-MX

Target Applications

- Large language models (LLMs) across multiple applications

Overview

The MIPS ARC® NPX6 NPU IP Processor's AI data compression option offers advanced data conversion and compression capabilities to enhance the efficiency and performance of AI and machine learning (ML) applications, particularly large language models (LLMs). For SoC designers, the licensable option provides more efficient data handling, improved computational performance, and simplified memory management, all while ensuring compliance with industry standards.

OCP-MX Support in ARC NPX6 NPU Processor

The AI data compression option for MIPS ARC NPX6 and NPX6 Functional Safety (FS) Processor IP, when combined with the ARC Tensor Floating Point Unit (FPU) option for NPX6, adds support for OCP-MX data compression to any ARC NPX6 NPU IP Processor.

The OCP-MX is a microscaling format that compresses data, sharing the common part of the exponent across a set of 32 elements. It enables AI inference with smaller memory footprints and more efficient memory bandwidth utilization, driving hardware performance and efficiency gains that reduce overhead and operational costs.

The AI data compression option supports the OCP-FP8 and OCP-MX formats in the NPX6 iDMA and oDMA engines, converting data to and from BF16 format in L1 VM (vector memory). Compute kernels on the convolution accelerator and General Tensor Accelerator (GTA) accelerator operate using the BF16 format, which requires configuring the NPX tensor FPU option.

A typical use case for the OCP-MX format is the LLM. It helps in efficiently reading dictionaries and reading/writing the key and value (KV) cache of the underlying transformer, which are fundamental components in LLM architectures.

The AI data compression option for ARC NPX6 NPU Processors are fully compliant with the OCP-MX specification based on:

- OCP 8-bit Floating Point Specification (OFP8), Rev. 1.0, Approved: June 20, 2023
- OCP Microscaling Formats (MX) Specification, Version 1.0, Sep 2023

Hardware Features for OCP-MX

The key features and benefits of the data compression option for ARC NPX6 NPU IP Processors are described below.

Convergent Rounding and Saturation Modes

The ARC NPX6 oDMA supports convergent rounding (roundTiesToEven) method and both saturation modes (to +/- infinity or +/- max_normal) when converting from BF16 to OCP-MX and FP8 formats (Table 1).

iDMA and oDMA Conversions			
XM (DDR/CSM)	L1 (VM)	XM → L1	L1 → XM
INT8, INT16, BF16, FP16	INT8, INT16, BF16, FP16	Copy	Copy
INT4	INT8	Sign-extend	Saturate to [min_INT4, max_INT4]
E5M2	BF16 | FP16	Mantissa pad, exponent bias offset	Round, saturate ±max_E5M2 | infinity
E4M3	BF16 | FP16		Round, saturate ±max_E4M3 | NaN
MXE5M2	BF16		Round, saturate ±max_E5M2 | infinity
MXE4M3	BF16		Round, saturate ±max_E4M3 | NaN
MXE3M2	BF16		Round, saturate ±max_E3M2
MXE2M3	BF16		Round, saturate ±max_E2M3
MXE2M1	BF16		Round, saturate ±max_E2M1
MXINT8	BF16	Normalize	Round, saturate ±max_INT

1. Rounding will use roundTiesToEven (convergent rounding)
2. (MX)E5M2 and (MX)E4M3 saturation depends on mode bit

Table 1: Supported data-type conversions in the iDMA and oDMA engines

Vectorization Support

The ARC NPX6 Processor supports the vectorization of OCP-MX data types in the channel dimension (see section 5.1 of the OCP-MX specification). This dimension is used for computing inner products during matrix multiplications, a common operation in neural networks. Vectorization in this dimension can significantly speed up these computations by processing multiple data points in parallel.

Storage and Scaling Factors

In CSM or DDR, OCP-MX vector elements are stored together with their associated scale in a packed format. This ensures that the scaling factors are readily accessible when needed, improving computational efficiency.

Bus Interface and Bandwidth

- NPX6 supports up to 4 64B AXI bus interfaces to transfer tensors to/from DDR, with an aggregate peak bandwidth of 2*4*64 GB/s (concurrent read&write) at 1GHz operating frequency
- The iDMA and oDMA engines in each NPX core support AXI 64B bus interfaces
- The latest NPX6 iDMA engine supports 128B interfaces to VM
- Each iDMA can read and convert OCP-MX MXFP8 and MXINT8 to BF16 without stalling its AXI interface
- For OCP-MX MXFP6 and MXFP4, multiple cores can operate concurrently to avoid stalls on the NPX top-level AXI interfaces
- The oDMA bandwidth to VM, used in KV-cache writing, is less critical and limited to 64B/cycle on the VM memory

Deliverables

- RTL supporting the afore mentioned data-type conversion
- Confidence test library (CCTs), demonstrating DMA conversions
- Documentation

About MIPS:

MIPS by GlobalFoundries delivers software to silicon with RISC-V for building physical AI platforms. MIPS delivers software-hardware co-design, optimized AI, and custom ASSP design and manufacturing. Together with ARC, MIPS delivers the open, standards-based processor IP portfolio for embedded applications. Physical AI is built on MIPS.

For more information, visit **www.mips.com/arc.**