2025 International Solid-State Circuits Conference

(ISSCC) Review

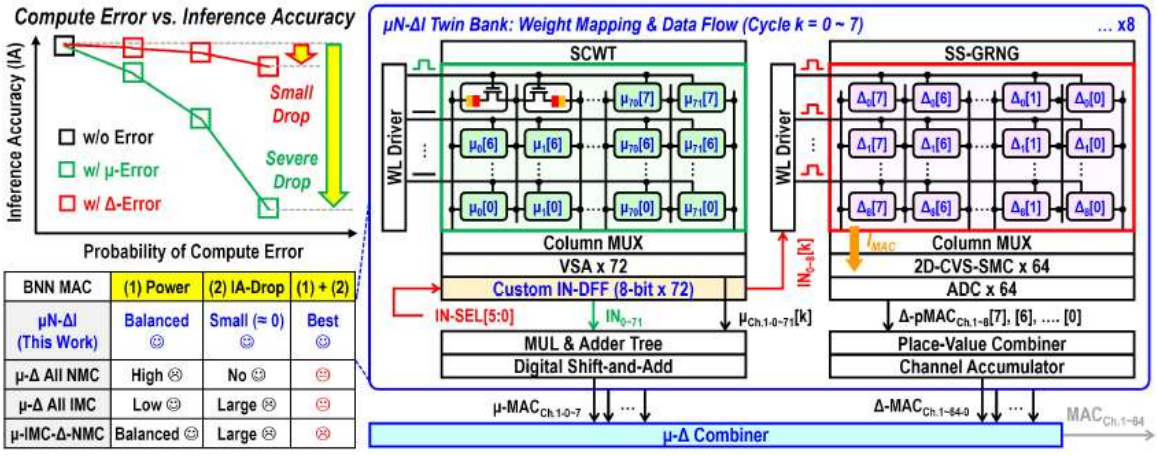
중앙대학교 창의ICT공과대학 전자전기공학부 한동현 교수

Topic : AI-Accelerators and Compute-In-Memory

Session 14 : Compute-In-Memory

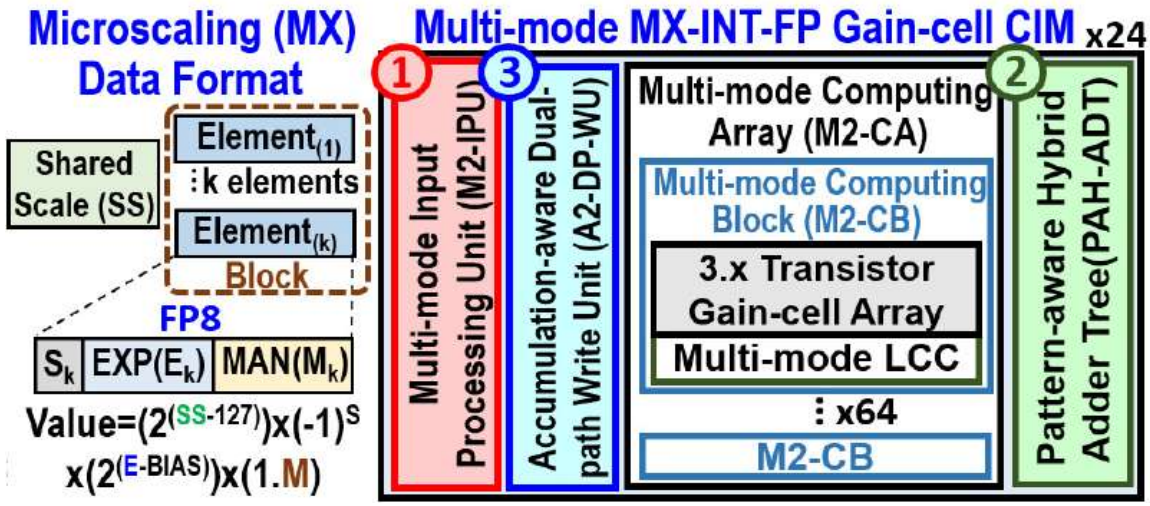
이번 ISSCC 2025의 Session 14는 인메모리 컴퓨팅 (In-memory Computing)을 주제로 총 7편의 논문이 발표되었다. 올 해 ISSCS에서 발표된 CIM은 총 3가지의 주요 트랜드를 보였다. 첫째, 작년에는 SRAM, DRAM, RRAM, FLASH 등 다양한 메모리를 활용한 CIM이 발표되었다면, 올 해는 한 편의 STT-MRAM 기반 CIM (논문 14.1) 한 편의 Gain Cell 기반 CIM (논문 14.2) 를 제외하면 모두 SRAM 기반 CIM을 발표하였다. 둘째, CIM에서 부동소수점을 어떻게 효율적으로 혹은 정확하게 지원할지에 집중하였다. (논문 14.2, 14.3, 14.4, 14.5) 셋째, 아날로그 (논문 14.1) 혹은 혼성 신호 기반 CIM (논문 14.6) 보다는 디지털 기반 CIM 관련 논문이 대다수였다.

#14-1은 국립칭화대학교와 TSMC에서 공동 연구로 발표한 22nm STT-MRAM 기반 CIM 으로 다른 논문과 달리 특별히 Bayesian Neural Network (BNN) 가속을 목표로 설계된 것이 큰 특징이다. 특히 이미 값이 정해져 있는 기존 인공지능 연산과 달리 난수를 계속 생성해주어야 하는 BNN 연산 가속을 위해 MRAM에 Self-Compare Write-Termination 회로를 추가하였으며, 정확한 연산과 부정확한 연산이 필요한 부분을 구분해 따로 다른 최적화된 CIM 을 설계함으로써 에너지 효율을 극대화하였다. CIFAR-100 이라는 비교적 쉬운 데이터셋에서 ResNet-20 이라는 작은 인공지능 모델을 사용하였음에도, 1.19%의 상대적으로 높은 정확도 손실이 측정되는 점이 아쉽지만, 해당 측정 결과에서 기존 STT-MRAM 기반 CIM 대비 2.3배 높은 에너지 효율을 달성하였다.



[그림 1] "14.1. A 22nm 104.5TOPS/W μ-NMC-Δ-IMC Heterogeneous STT-MRAM CIM Macro for Noise-Tolerant Bayesian Neural Networks" 전체 아키텍처 구조

#14-2는 국립칭화대학교와 TSMC에서 공동 연구로 발표한 16nm Gain-cell 기반 CIM 으로 부동소수점 지원을 위해 흔히 사용하는 Shared exponent 혹은 Shared scale을 CIM 외부에 의존하지 않고, CIM 내부에서 모두 수행 가능하게 만듦으로써, 불필요한 데이터 전송을 최소화시킨 논문이다. 더불어, CIM에서 가장 많은 면적과 전력소모를 보이는 Adder tree의 오버헤드를 최소화 하기 위해 자주 나타나는 데이터 패턴만을 위해 최적화된 적은 트랜지스터를 사용한 Adder tree를 함께 활용하는 방식을 제안하였다. 해당 프로세서는 기존 프로세서 대비 1.8~2.5배 높은 FoM을 달성하여, Shared scale을 사용하는 MXINT8 포맷 데이터 연산시 133.5 TFLOPS/W를, INT8 지원시 205.4 TOPS/W 에너지 효율을 달성하는데 성공하였다.



[그림 1] "14.1. A 22nm 104.5TOPS/W μ-NMC-Δ-IMC Heterogeneous STT-MRAM CIM Macro for Noise-Tolerant Bayesian Neural Networks" 전체 아키텍처 구조

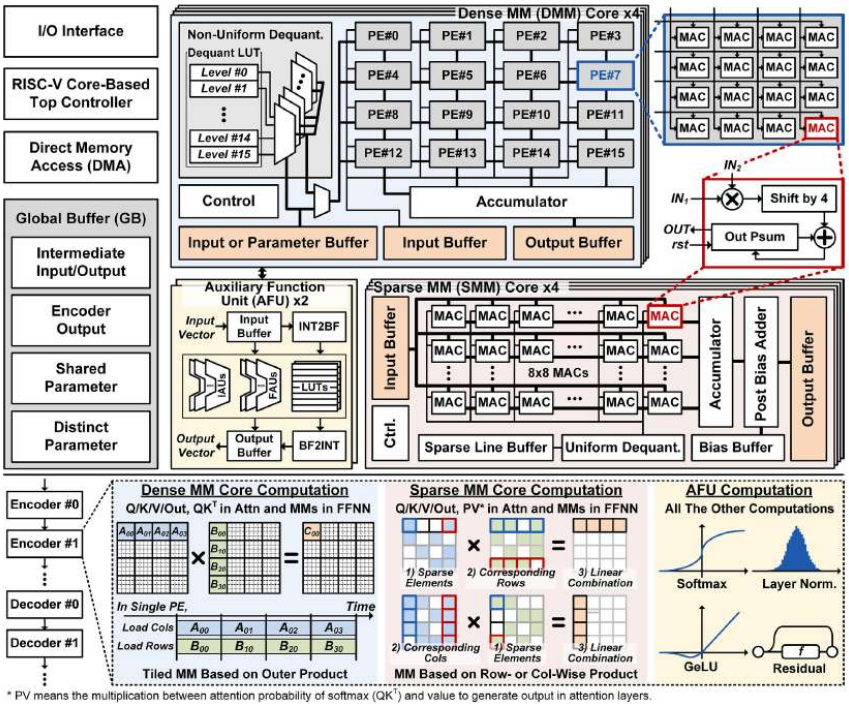
#14-5는 Institute of Microelectronics Chinese Academy of Sciences에서 발표현 28nm SRAM 기반 CIM으로 다른 CIM과 달라 Edge에서의 추론과 학습 모두를 지원한다고 주장한다. 이전에도 학습을 지원한다고 주장했던 CIM 논문이 있었지만, 추론과 오류 전사가 독립된 CIM에서 수행되었던 단점을 지적하면서, 이를 통합하고, 부동소수점 연산 지원과 전치 행렬 연산을 동시에 수행하는 CIM 을 설계하였다. 하지만 학습의 주요과정인 가중치 업데이트가 생략되었다는 점, 제작한 프로세서를 측정하기 위해 사용한 CIFAR-10과 CIFAR-100이 비교적 쉬운 데이터셋이라는 점, 사용한 AI 모델역시 너무 단순하다는 점이 아쉽다. 오류를 감수하더라도 근사 연산을 수행하거나, 학습 등을 위해 정확한 연산을 함께 수행할 수 있는 MAC이 집적되어 있어, 근사 연산 수행시 192.3 TFLOPS/W 의 에너지 효율을 달성하였다.

Session 23 : AI-Accelerators

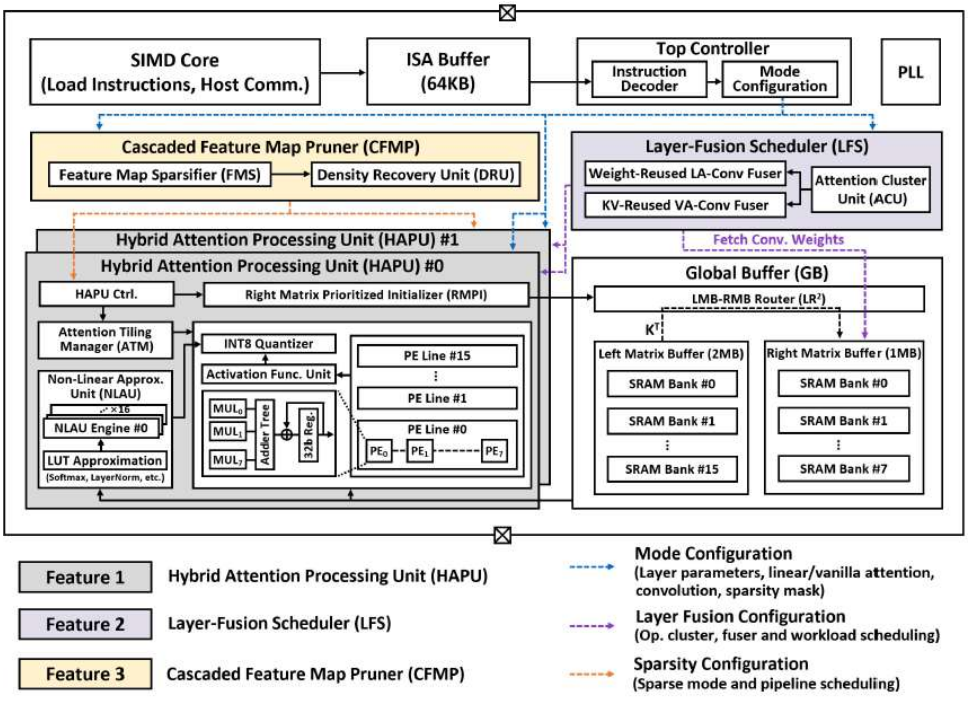
이번 ISSCC 2025의 Session 23은 인공지능 가속기를 주제로 총 10편의 논문이 발표되었다. 올 해 ISSCS에서 발표된 인공지능 가속기 (Session 2의 일부 논문 제외)는 총 2가지의 주요 트랜드를 보였다. 첫째, 확산 모델 (Diffusion Model)을 포함한 생성형 인공지능 (Generative AI, GenAI) 가속의 중요성이 많이 강조되었다. (논문 23.3, 23.5, 23.6, 23.10) 둘째, Large Language Model (LLM) 혹은 트랜스포머 (Transformer) 가속이 여전히 강조되고 있다. (논문 23.1, 23.2, 23.8, 23.9) 주요했던 아이디어 트랜드로 보았을 때는, 첫째, 하드웨어 리소스를 최대한 잘 활용하기 위한 컴파일러, 스케줄러의 중요성이 강조되었다. (논문 23.2, 23.5) 둘째, 채널 별, 타임스텝 별 유사도를 이용해 손실없이 고효율 연산을 수행하는 고정소수점 기반 혼합 정밀도 연산 (논문 23.3, 23.7, 23.10), 혹은 고정소수점-부동소수점 혼합 연산 (논문 23.6, 23.8)을 채택하였다. 셋째, 에너지 효율을 더 높이기 위해 이형 코어 (Heterogeneous Core) 구조의 중요성이 강조되었다. (논문 23.1, 23.6, 23.10) 넷째, 프로세서 자체의 에너지 효율도 중요하지만, 외부 메모리 접근량 (External Memory Access, EMA)감소의 중요성이 강조되었다. (논문 23.1, 23.2, 23.3, 23.5, 23.6, 23.8, 23.9)

#23-1은 Columbia University에서 발표한 16nm Fin-FET 기반 트랜스포머 가속기로, Factorizing Training이라는 새로운 학습 기법을 적용해 레이어마다 공유하는 가중치를 학습하고, 이를 통해 트랜스포머 추론 가속시 외부메모리 접근량을 크게 줄이는데 성공하였다. 특히 Factorizing Training으로 인해 모든 레이어에서 공유되는 가중치는 데이터 희소성이 높지 않지만, 레이어마다 독립적으로 갖는 가중치는 데이터 희소성이 높은 특성을 활용하기 위해 이형 코어 구조를 채택한다. 이때 2개의 서로 다른 이형 코어가 메모리에 다르게 접근하여 생기는 문제를 해결하기 위해, load / store 동작을 수평 / 수직 방향에서 동시에 수행할 수 있는 Two-Direction Accessible Register File을 함께 설계하였다. Factorizing Training기반 최적화를 통해, 본 논문은 32~66배 적은 외부 메모리 접근량을 달성하는데 성공하였다.

#23-2는 홍콩과학기술대학에서 발표한 28nm 기반 트랜스포머 가속기로, 트랜스포머 연산 중 발생하는 Queue, Key, Value로 인한 외부메모리 접근량을 최소화하기 위해 Attention 연산을 Softmax를 제대로 활용하는 Vanilla Attention과 선형 근사하는 Linear Attention 으로 나누어 처리하는 방식을 채택했다. 해당 방식은 정확도 손실과 Trade-off 관계임을 밝히고 있으며, 레이어마다 최적의 Vanilla Attention/ Linear Attention 비율을 찾아주어야 최적의 효율이 나올 수 있다. 뿐만 아니라, Linear Layer 은 뒤에 이어지는 Convolution layer와 합쳐질 수 있으므로, 연산량을 최소화하고 코어 활용도를 극대화 하기 위해, 런타임 Layer Fusion 스케줄러를 같이 결합한 것이 특징이다. 본 논문 역시 25.9배 적은 외부 메모리 접근량을 보이면서 이를 통해, 4.6배에서 7.1배 높은 추론 속도와 4.4~7.1배 적은 시스템 에너지 소모를 달성하였다.



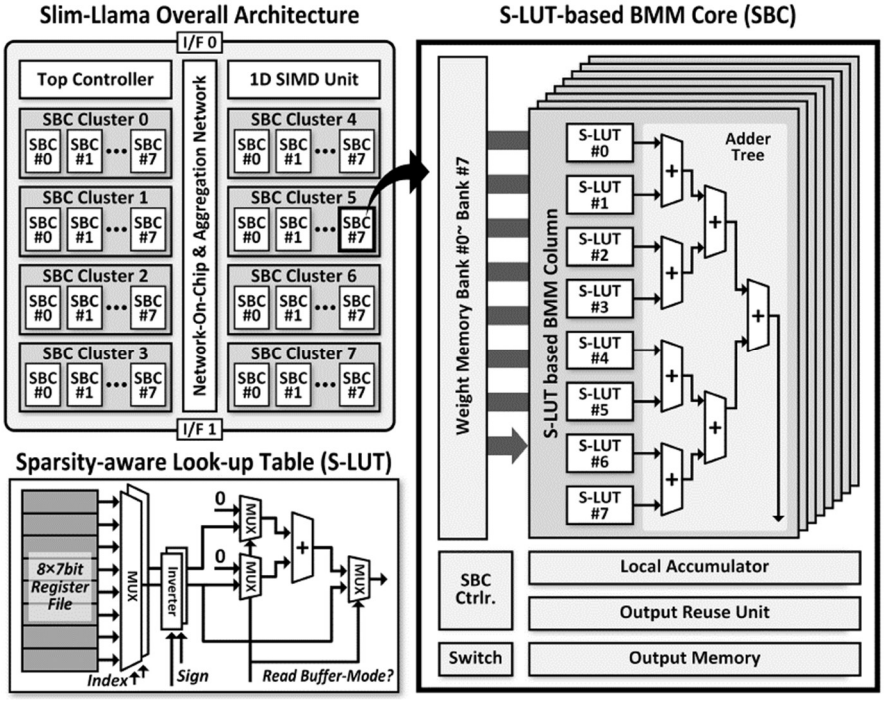
[그림 1] "23.1. T-REX: A 68-to-567μs/Token 0.41-to-3.95μJ/Token Transformer Accelerator with Reduced External Memory Access and Enhanced Hardware Utilization in 16nm FinFET" 전체 아키텍처 구조



[그림 1] "23.2. A 28nm 0.22μJ/Token Memory-Compute-Intensity-Aware CNN-Transformer Accelerator with Hybrid-Attention-Based Layer-Fusion and Cascaded Pruning for Semantic-Segmentation" 전체 아키텍처 구조

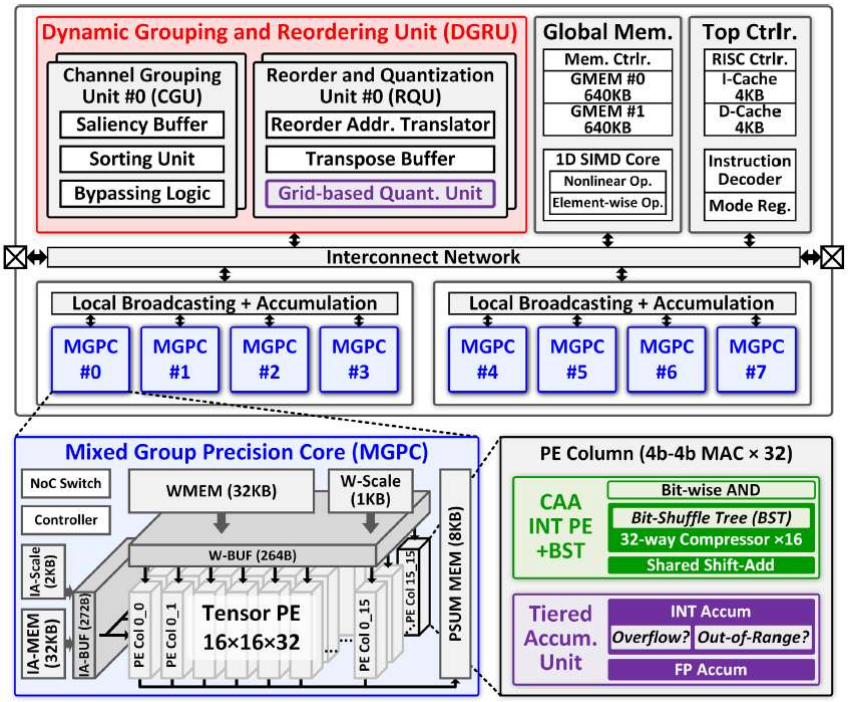
#23-8은 칭화대에서 발표한 28nm 기반 트랜스포머 가속기로 23.1과 23.2처럼 학습 자체를 새롭게 하여 추론시 성능을 높이는 방식이 아닌 주어진 트랜스포머 구조 가속에 집중한 논문이다. 마찬가지로 트랜스포머 가속시 발생하는 많은 외부 메모리 접근량을 해결하기 위해, KV 캐시를 효율적으로 압축할 수 있는 방식과 고정소수점-부동소수점을 함께 활용하는 방식을 채택하였다. 특히 누적기에서 필요로 하는 비트수를 획기적으로 줄이기 위해 비슷한 Exponent 별로 모아서 연산하는 Processing Element를 고안하였다.

#23-9는 KAIST에서 발표한 28nm 기반 LLM 가속기로, 많은 외부 메모리 접근 문제와 연산 효율 극대화를 위해 Binary / Tenary로 양자화된 LLM 을 가속한다. 낮은 비트수로 양자화 되면서 가중치에서 보이는 비슷한 패턴을 최대한 재사용하여 처리량을 늘리고, 가중치의 순서까지 조절해, 발생할 수 있는 Bit Toggling을 최소화한다. 이렇게 Binary / Ternary 가중치로 양자화된 LLM 전용 가속기는 이전 LLM 가속기 대비 4.6배 높은 에너지 효율 달성에 성공하였다.



[그림 1] "23.9. Slim-Llama: A 4.69mW Large-Language-Model Processor with Binary/Ternary Weights for Billion-Parameter Llama Model" 전체 아키텍처 구조

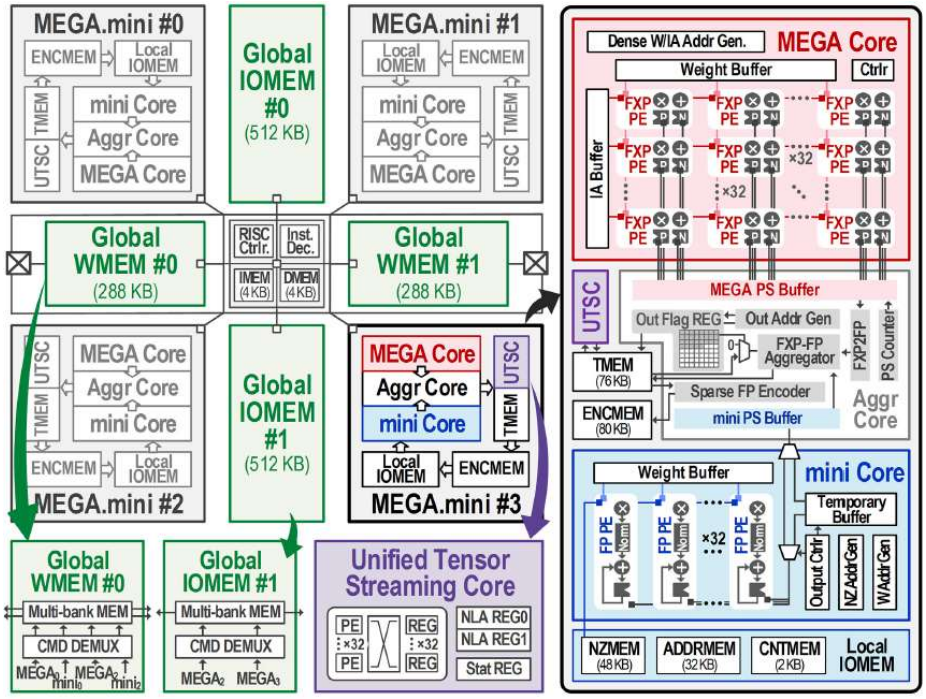
#23-3은 KAIST에서 발표한 28nm 기반 확산 모델 가속기로, 기존 여러 반복을 거쳐서 최종 출력을 만들어 내는 확산 모델의 단점을 해결하기 위해 나온 Few-shot 확산 모델 가속을 목표로 한다. 타임 스텝별로 유사도를 파악하고 이를 활용하던 기존 프로세서의 연산 특성을 사용할 수 없으므로, 중요도가 높은 채널들끼리 혹은 중요도가 낮은 채널들 끼리 묶고 유사한 채널들 사이에서 고정소수점으로 표현할 때 필요한 비트수를 최적화하는 방식을 채택하고, 채널별로 연산을 묶어 연산기를 최적화 하는 방식을 제안하였다. Few-shot 확산 모델과 이에 최적화된 양자화 기법을 토대로 기존 확산 모델 가속기 대비 3.3~6.8배 낮은 에너지 소모를 달성에 성공하였다.



[그림 1] "23.3. EdgeDiff: 418.4mJ/Inference Multi-Modal Few-Step Diffusion Model Accelerator with Mixed-Precision and Reordered Group Quantization" 전체 아키텍처 구조

#23-5는 MediaTek에서 발표한 3nm 기반 생성형 인공지능을 위한 엣지 디바이스용 프로세서로, 256KB의 아주 적은 온칩 메모리로 Convolution, 행렬곱 연산을 수행하기 위해 Layer Fusion을 최대한 활용할 수 있는 스케줄링을 제안하며, 특히 라인별 데이터가 준비되는 대로, 다음 레이어 연산을 바로 수행하는 것이 특징이다. 이런 연산을 구현하기 위해 라인 데이터를 저장하는 캐시를 별도로 둔 것이 아키텍처 수준에서 가장 큰 차별점이며, Branch가 있을 경우 지연 시간 최소화를 위해, 컴파일러 수준 최적화를 동시에 진행하였다. 본 프로세서가 3nm 기반으로 만들어졌다는 점, 0.168 mm^2의 작은 면적에 다양한 생성형 인공지능 연산 수행을 위한 스케줄링, 아키텍처, 컴파일러 최적화 기법이 들어간 점이 다른 논문들과의 차별점이라 볼 수 있다.

#23-6은 MIT에서 발표한 28nm 기반 생성형 인공지능 가속기로 CNN, 트랜스포머, 확산 모델 등, 모델과 어플리케이션에 제한없이 정확도 손실을 최소화하면서 효율적으로 가속할 수 있는 방법을 제안한다. 특히 Inlier 입력은 고정소수점으로, Outlier 입력은 부동소수점으로 표현하는 방식을 채택하고, 데이터 재사용을 극대화한 고정소수점 전용 가속기와 데이터 희소성을 활용하는 부동소수점 기반 가속기를 모두 활용하는 아키텍처가 특징이다. 본 논문은 CPU의 빅/리틀 코어 아키텍처처럼 NPU도 새로운 빅/리틀 코어 아키텍처 (메가/미니)가 필요함을 강조하였다. 뿐만 아니라, 최신 AI 모델에서 많이 사용되는 정규화 및 비선형 함수 가속 방법 역시 함께 제안한다. 판별형, 생성형 AI 상관없이 84% 이상의 높은 에너지 효율을 달성하였으며, 기존 확산 모델 전용 가속기와 비교하여도, 2.1배 높은 추론 속도와 3.4배 높은 에너지 효율을 달성하였다.



[그림 1] "23.6. MEGA.mini: A Universal Generative AI Processor with a New Big/Little Core Architecture for NPU" 전체 아키텍처 구조

저자정보

한동현 교수

- 소 속 : 중앙대학교 창의ICT공과대학 전자전기공학부
- 연구분야 : Digital Architecture & SoC Design
- 이 메 일 : hdh4797@cau.ac.kr
- 홈페이지 : https://hdh4797.wixsite.com/dhan