제시 내용

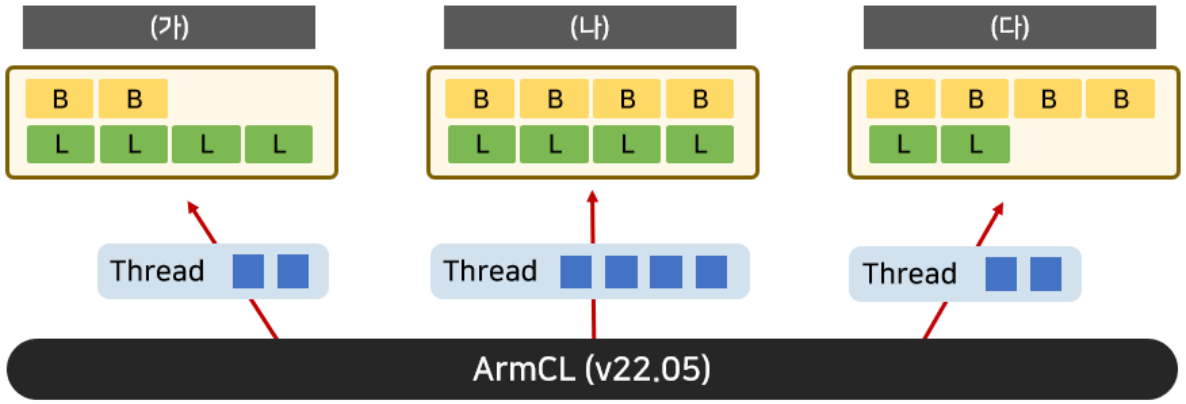
스케줄러의 특정 CPU 코어 구조에서 고전력/고성능 코어의 수가 저전력/저성능의 코어의 수보다 많은 경우 저전력/저성능의 코어의 수만큼 쓰레드가 생성된다.

이는 저전력/저성능 코어를 사용하지 않으면서 고전력/고성능 코어의 일부만 활용한다는 것을 의미한다.

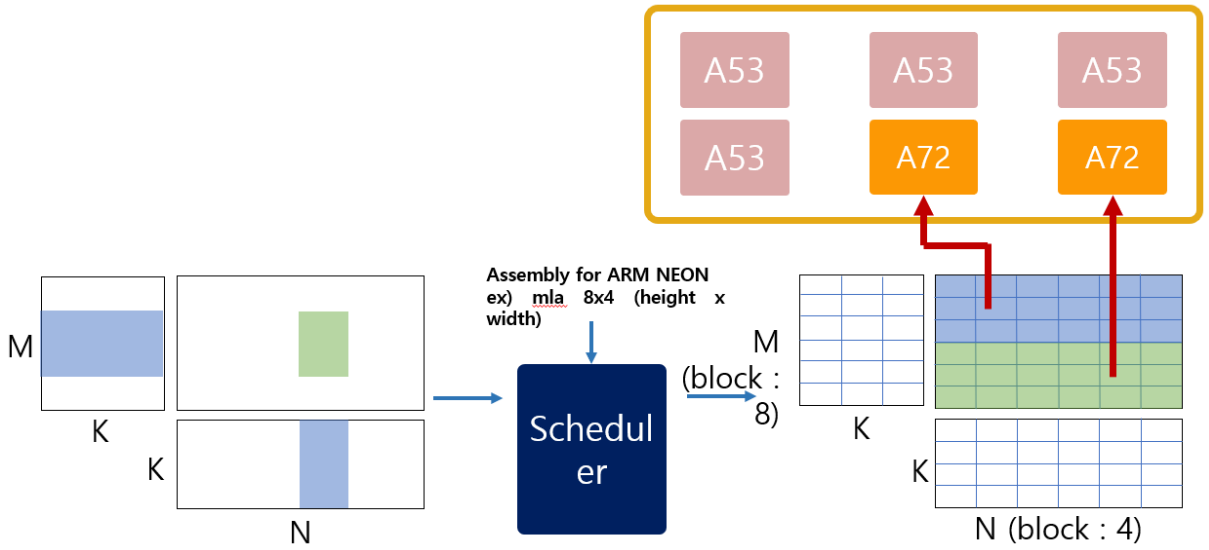
따라서 ArmCL은 ARM big.LITTLE 구조에 대한 최적화나 지원이 제공되고 있지 않다.

MIN(SELECT count(*) FROM Core GROUPBY Model)

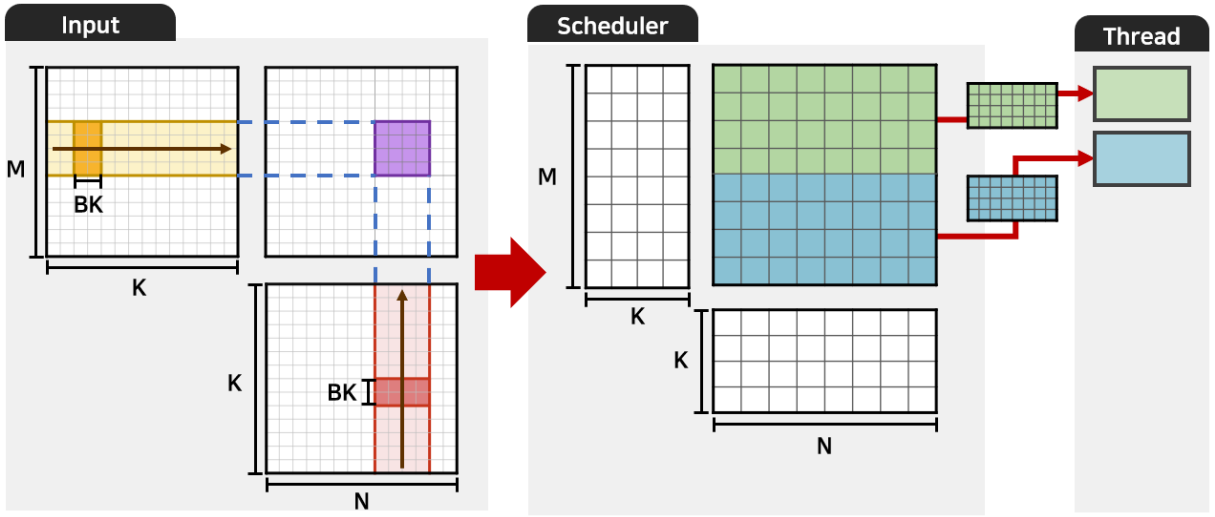
제작 결과



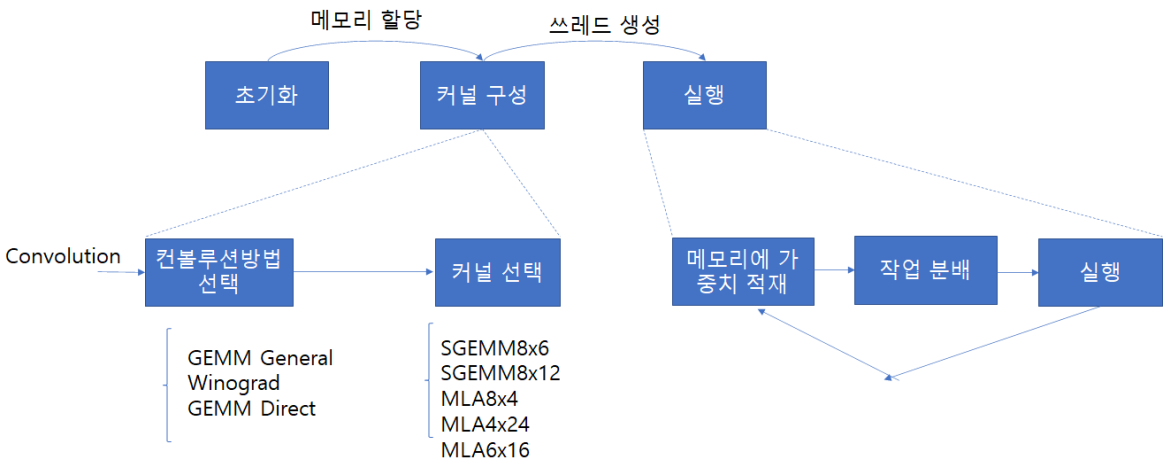
제시 내용


A53
A53
A53
A53
A72
A72
M
K
K
N
Assembly for ARM NEON
ex) mla 8x4 (height x width)
Schedul
er
M
(block :
8)
K
K
N (block : 4)

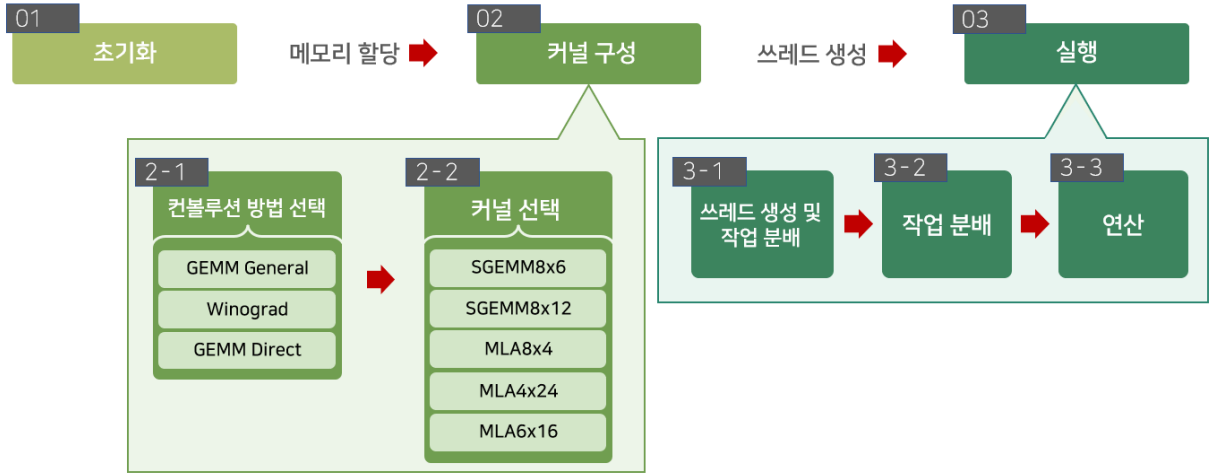
제작 결과


Input
M
BK
K
K
BK
N
Scheduler
M
K
K
N
Thread

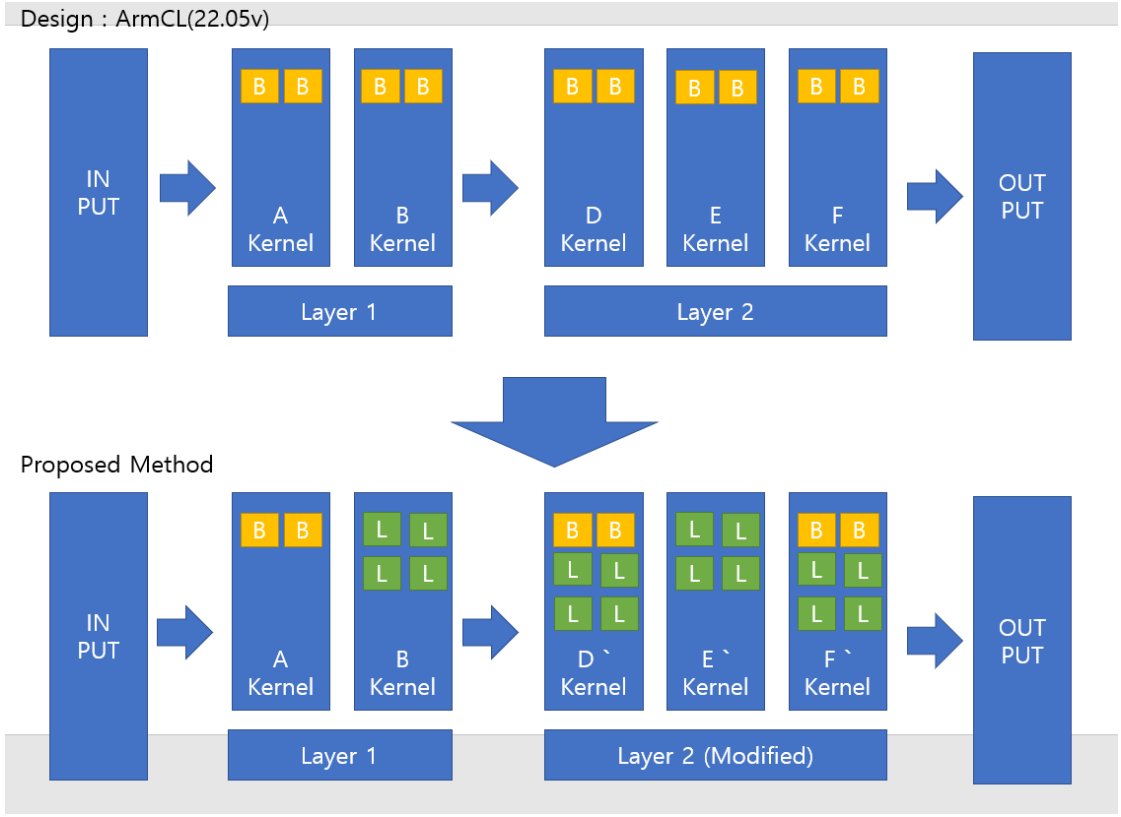
제시 내용


메모리 할당
쓰레드 생성
초기화
커널 구성
실행
Convolution
컨볼루션방법 선택
커널 선택
메모리에 가중치 적재
작업 분배
실행
GEMM General
Winograd
GEMM Direct
SGEMM8x6
SGEMM8x12
MLA8x4
MLA4x24
MLA6x16

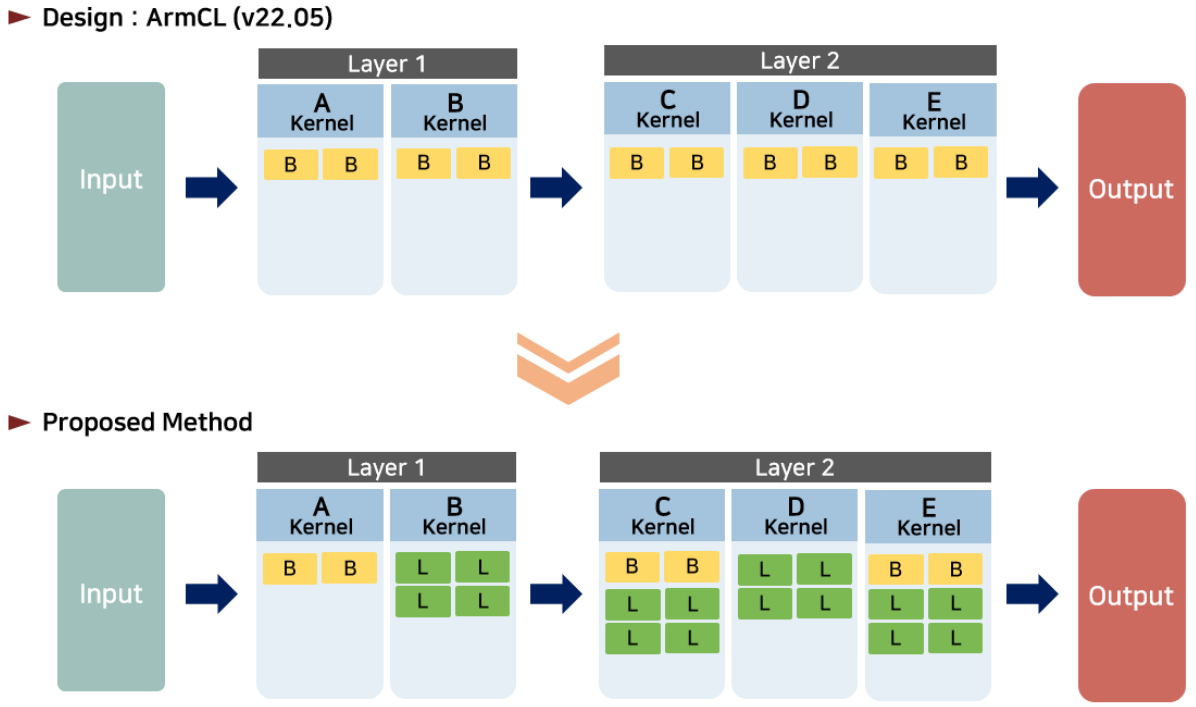
제작 결과


01
초기화
메모리 할당
02
커널 구성
쓰레드 생성
03
실행
2-1
컨볼루션 방법 선택
GEMM General
Winograd
GEMM Direct
2-2
커널 선택
SGEMM8x6
SGEMM8x12
MLA8x4
MLA4x24
MLA6x16
3-1
쓰레드 생성 및 작업 분배
3-2
작업 분배
3-3
연산

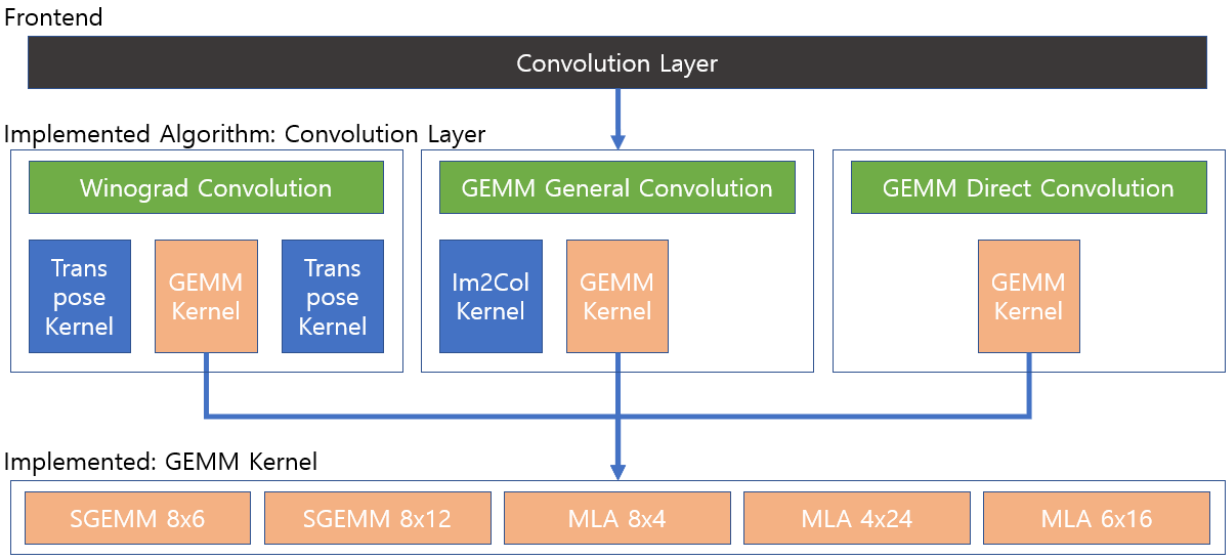
제시 내용


Design : ArmCL(22.05v)
IN PUT
A Kernel
B Kernel
D Kernel
E Kernel
F Kernel
OUT PUT
Layer 1
Layer 2
Proposed Method
IN PUT
A Kernel
B Kernel
D ` Kernel
E ` Kernel
F ` Kernel
OUT PUT
Layer 1
Layer 2 (Modified)

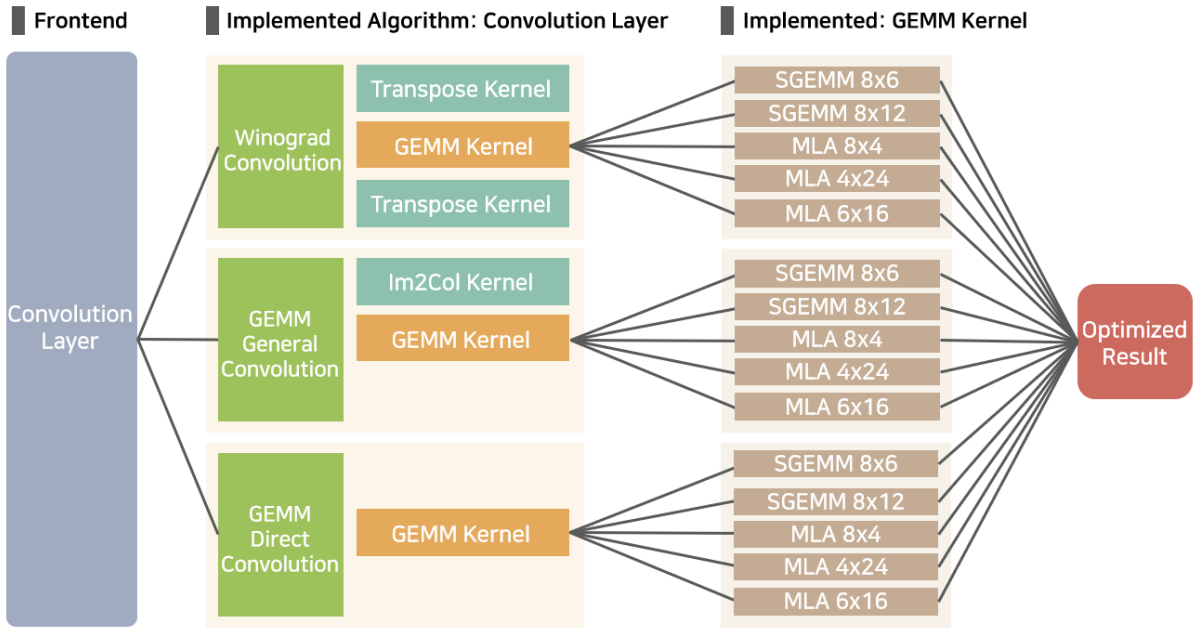
제작 결과


Design : ArmCL (v22.05)
Input
Layer 1
A Kernel
B Kernel
Layer 2
C Kernel
D Kernel
E Kernel
Output
Proposed Method
Input
Layer 1
A Kernel
B Kernel
Layer 2
C Kernel
D Kernel
E Kernel
Output

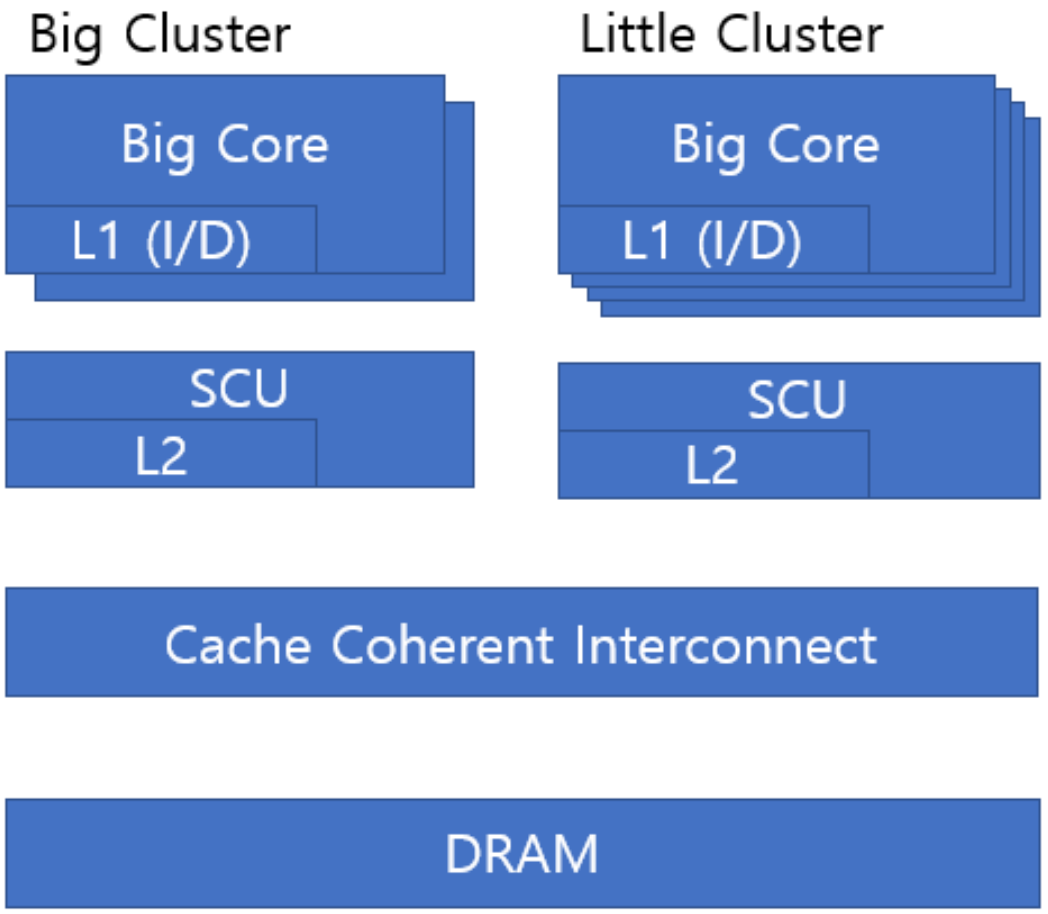
제시 내용


Frontend
Convolution Layer
Implemented Algorithm: Convolution Layer
Winograd Convolution
GEMM General Convolution
GEMM Direct Convolution
Trans pose Kernel
GEMM Kernel
Trans pose Kernel
Im2Col Kernel
GEMM Kernel
GEMM Kernel
Implemented: GEMM Kernel
SGEMM 8x6
SGEMM 8x12
MLA 8x4
MLA 4x24
MLA 6x16

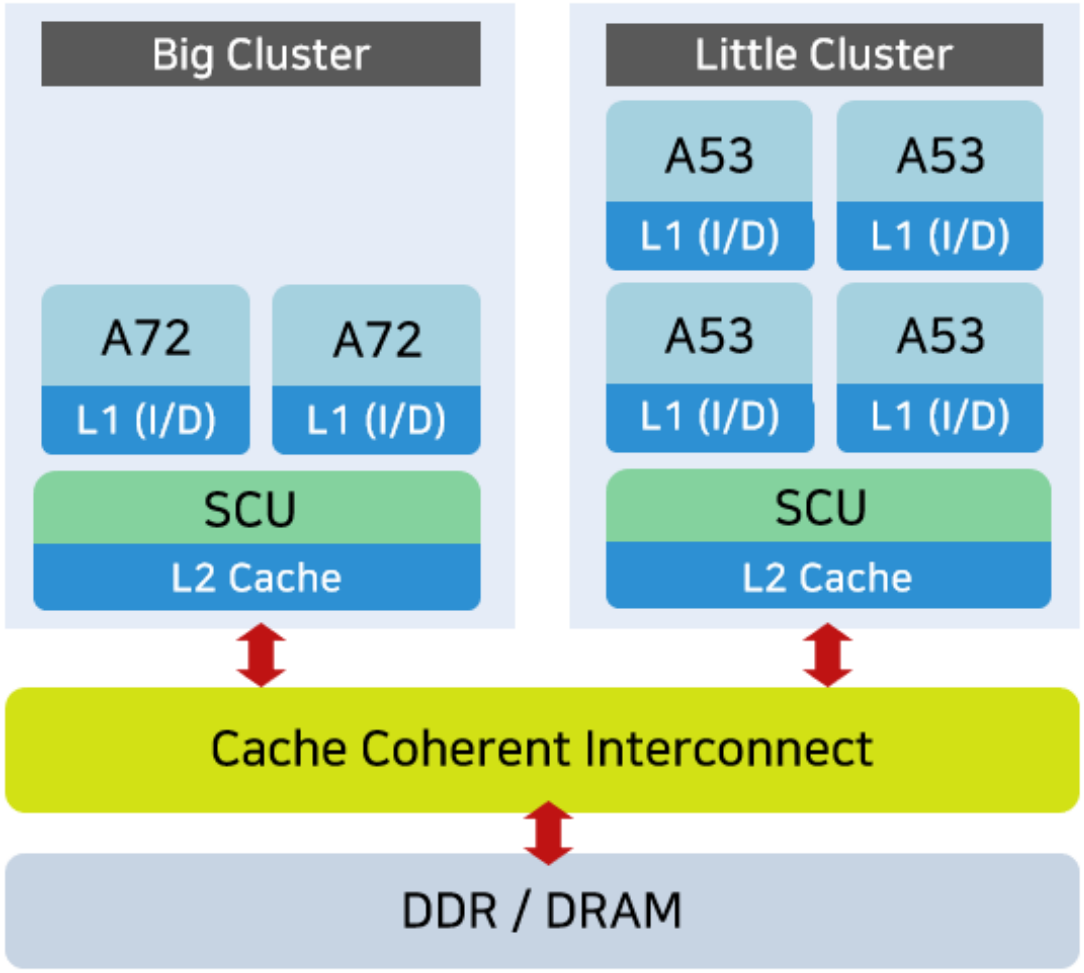
제작 결과


Frontend
Implemented Algorithm: Convolution Layer
Implemented: GEMM Kernel
Convolution Layer
Winograd Convolution
Transpose Kernel
GEMM Kernel
Transpose Kernel
GEMM General Convolution
Im2Col Kernel
GEMM Kernel
GEMM Direct Convolution
GEMM Kernel
SGEMM 8x6
SGEMM 8x12
MLA 8x4
MLA 4x24
MLA 6x16
Optimized Result

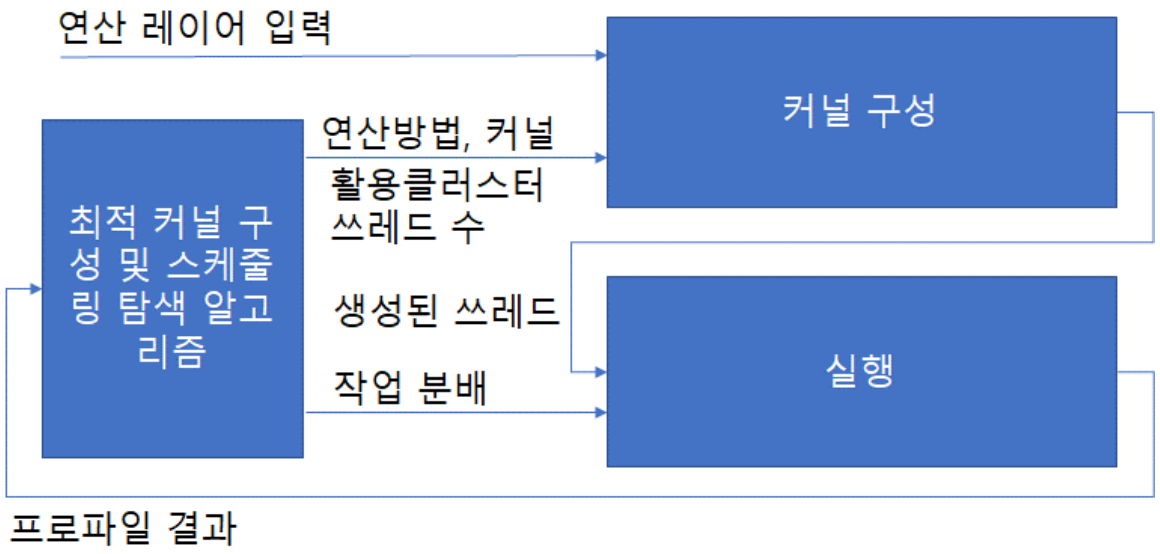
제시 내용


Big Cluster
Little Cluster
Big Core
L1 (I/D)
Big Core
L1 (I/D)
SCU
L2
SCU
L2
Cache Coherent Interconnect
DRAM

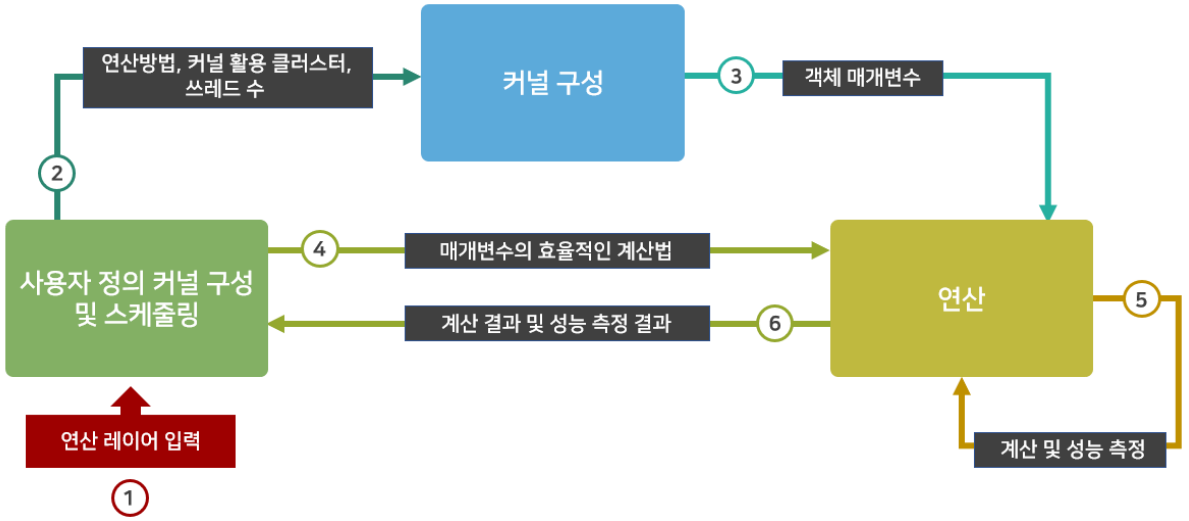
제작 결과


Big Cluster
Little Cluster
A53
L1 (I/D)
A53
L1 (I/D)
A72
L1 (I/D)
A72
L1 (I/D)
A53
L1 (I/D)
A53
L1 (I/D)
SCU
L2 Cache
SCU
L2 Cache
Cache Coherent Interconnect
DDR / DRAM

제시 내용


연산 레이어 입력
최적 커널 구성 및 스케줄링 탐색 알고리즘
연산방법, 커널
활용클러스터
쓰레드 수
생성된 쓰레드
작업 분배
커널 구성
실행
프로파일 결과

제작 결과


연산방법, 커널 활용 클러스터, 쓰레드 수
커널 구성
3
객체 매개변수
2
사용자 정의 커널 구성 및 스케줄링
4
매개변수의 효율적인 계산법
연산
5
계산 결과 및 성능 측정 결과
6
계산 및 성능 측정
연산 레이어 입력
1