INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING
ICPP/2021/CHICAGO/USA
acm In-Cooperation
sighpc
AUGUST 9–12, 2021

Accelerate Binarized Neural Networks with Processing-in-Memory Enabled by RISC-V Custom Instructions

Che-Chia Lin[1], Chao-Lin Lee[1], Jenq-Kuen Lee[1],
Howard Wang[2], Ming-Yu Hung[2]

[1]Department of Computer Science, National Tsing Hua University, Taiwan
[2]MediaTek Inc.

Outline

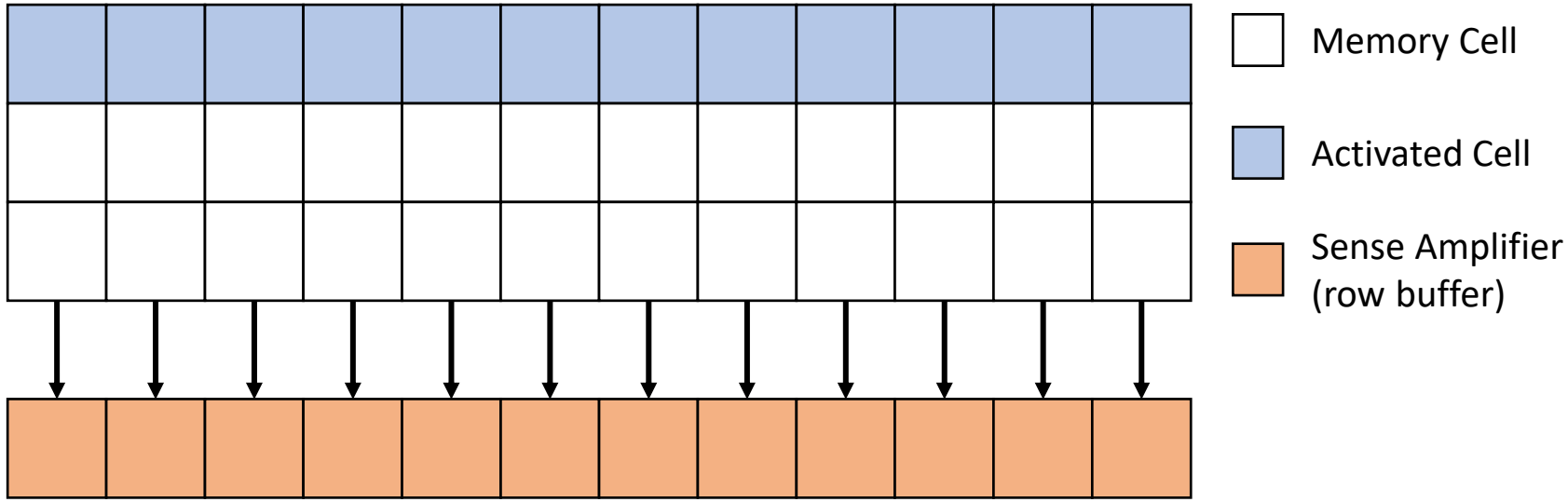
- Introduction & Background
- Processing-in-Memory Architecture and Custom Instructions
- TVM Compilation Flow
- Binarized Convolution of Processing-in-Memory
- Experiment Result
- Conclusion & Future Work

Introduction

- There are emerging technologies trying to take down the “Memory Wall” and one of the techniques is Processing-in-Memory (PIM).
- This work exploits PIM to accelerate Binarized Neural Networks (BNN).
  - We take convolution of BNNs as target application.
  - We model PIM in **Gem5** for supporting **bit-wise operations** and **population count (POPC)**.
  - PIM operations are supported as **RISC-V Custom Instructions**.
  - BNNs are compiled and deployed by **TVM**.
  - We propose a **memory layout** suitable for PIM operations.

Background - Processing-in-Memory

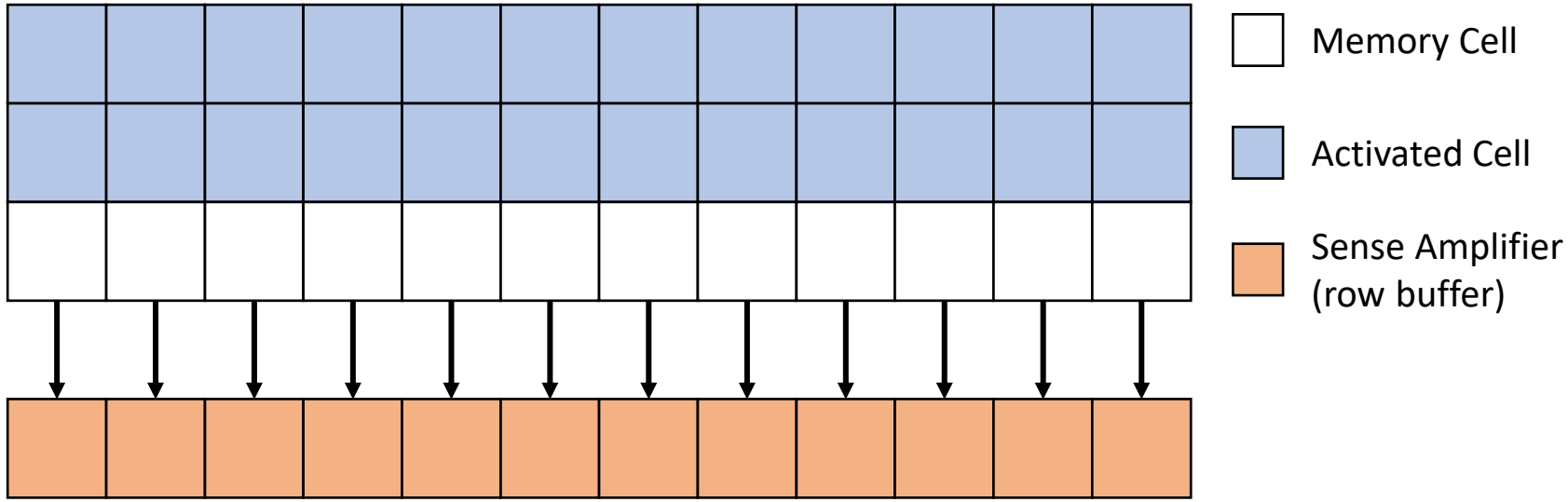
- Processing-in-Memory
  - Perform bit-wise operations on memory rows
  - Use sense amplifier to accomplish the operations



Seshadri, Vivek, et al. "Ambit: In-memory accelerator for bulk bitwise operations using commodity DRAM technology." *2017 50th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, 2017.

Background - Processing-in-Memory

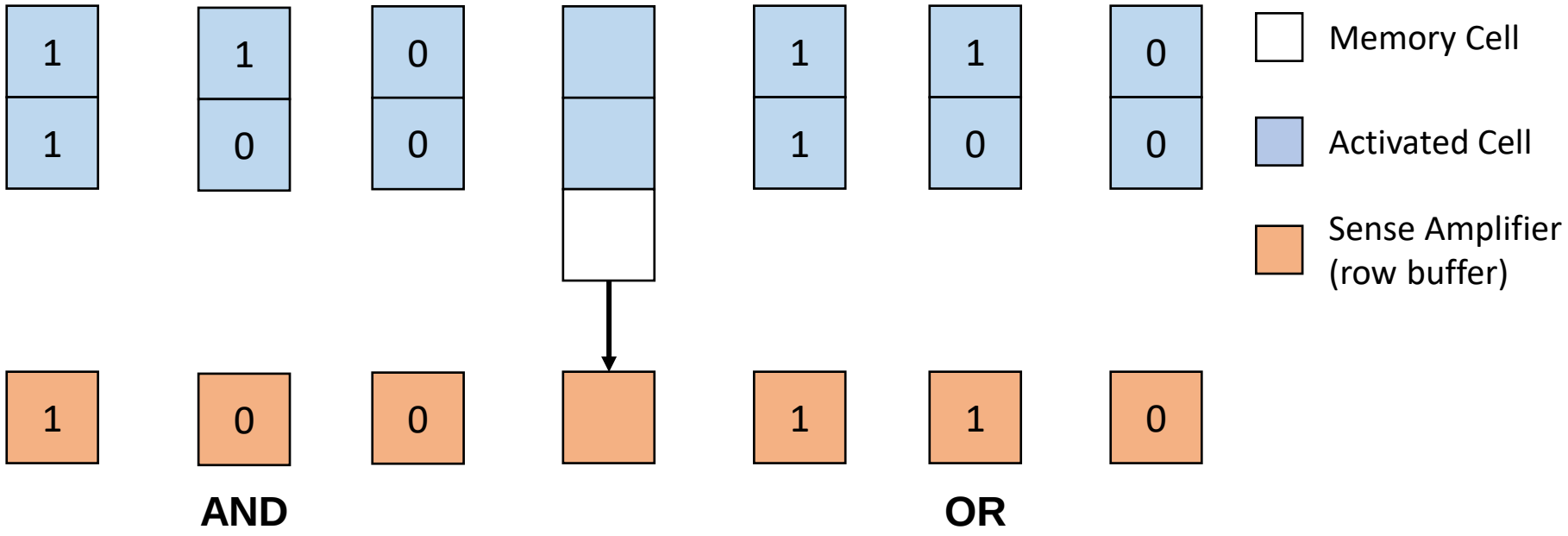
- Processing-in-Memory
  - Perform bit-wise operations on memory rows
  - Use sense amplifier to accomplish the operations



Seshadri, Vivek, et al. "Ambit: In-memory accelerator for bulk bitwise operations using commodity DRAM technology." *2017 50th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, 2017.

Background - Processing-in-Memory

- Processing-in-Memory
  - Perform bit-wise operations on memory rows
  - Use sense amplifier to accomplish the operations

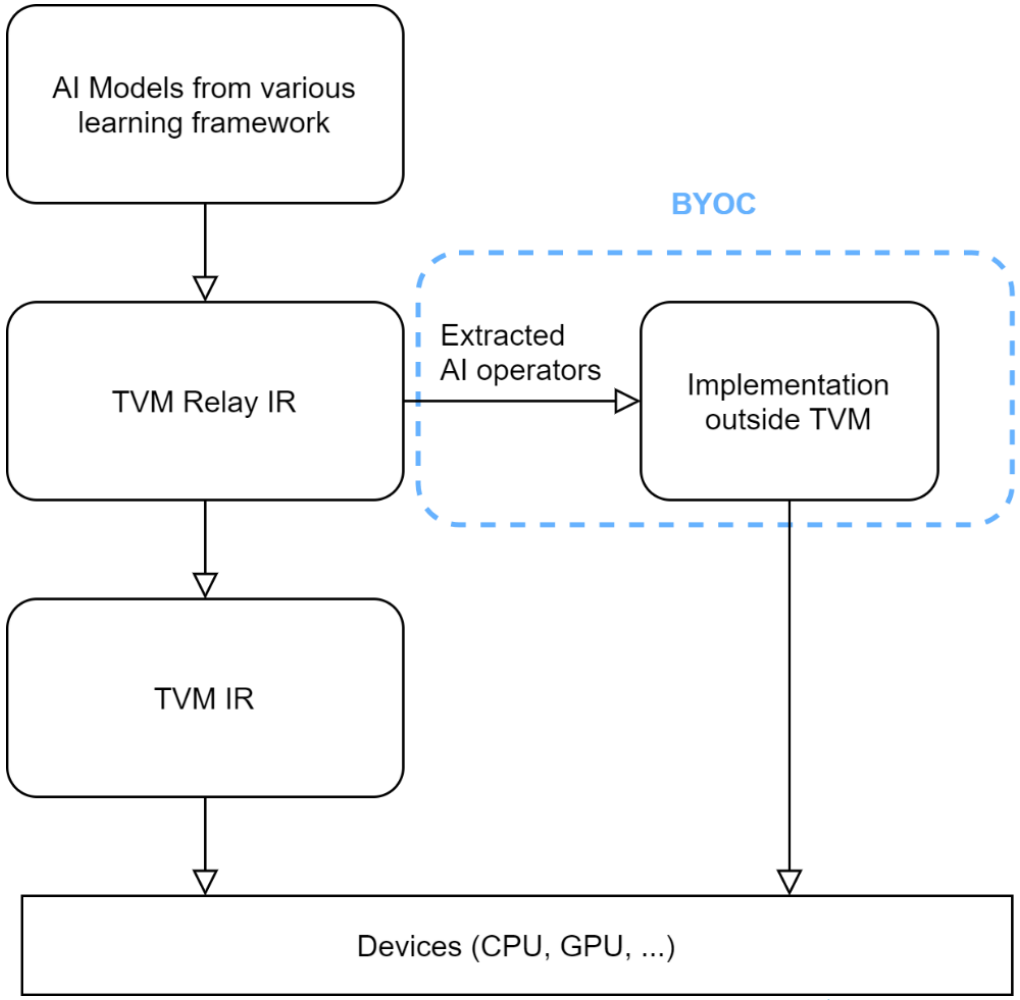


Seshadri, Vivek, et al. "Ambit: In-memory accelerator for bulk bitwise operations using commodity DRAM technology." *2017 50th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, 2017.

Background - TVM

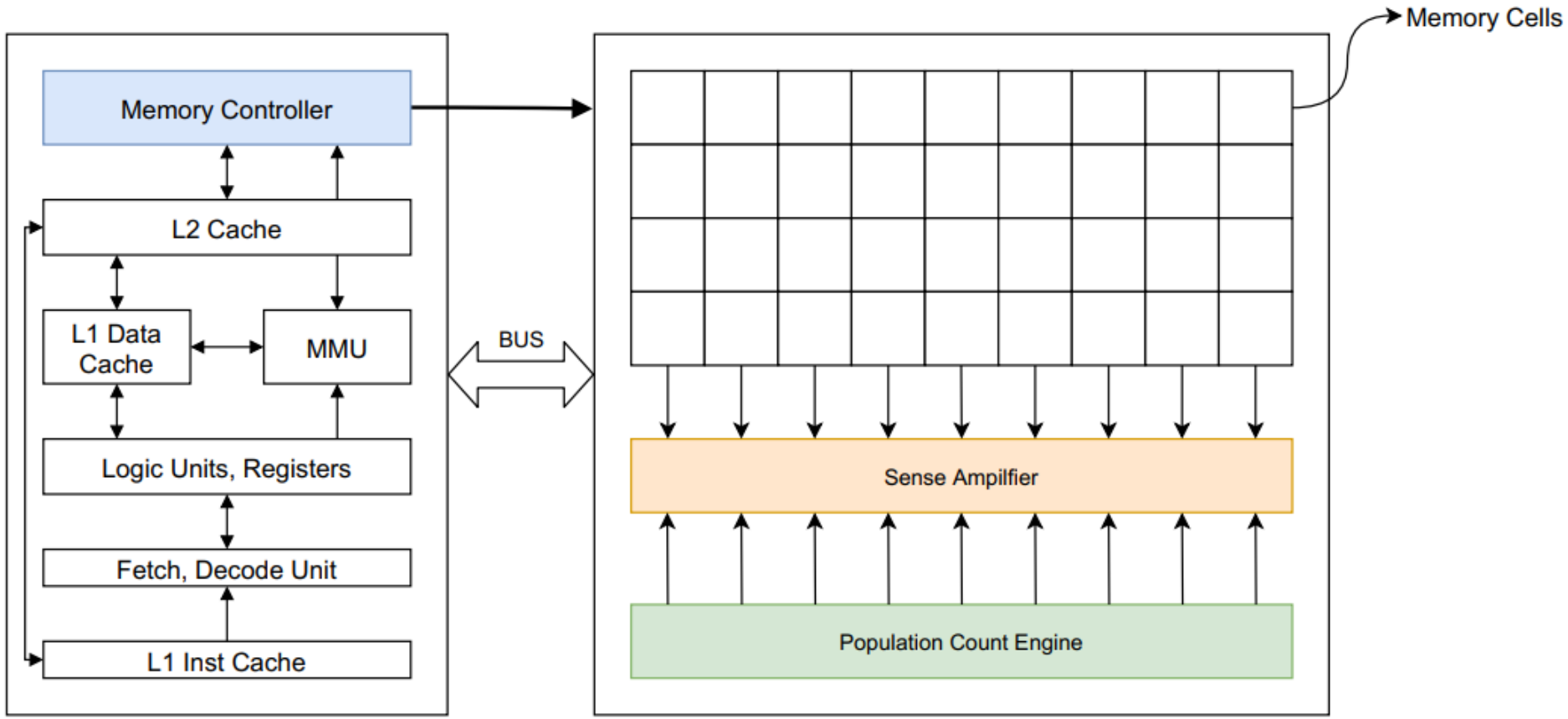
- AI compiler framework
- Support different formats

- Relay IR (Intermediate Representation) represents graph composed of operators
- Bring Your Own Codegen (BYOC)



Chen, Tianqi, et al. "{TVM}: An automated end-to-end optimizing compiler for deep learning." *13th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 18)*. 2018.

Our PIM Architecture



Our PIM Instructions

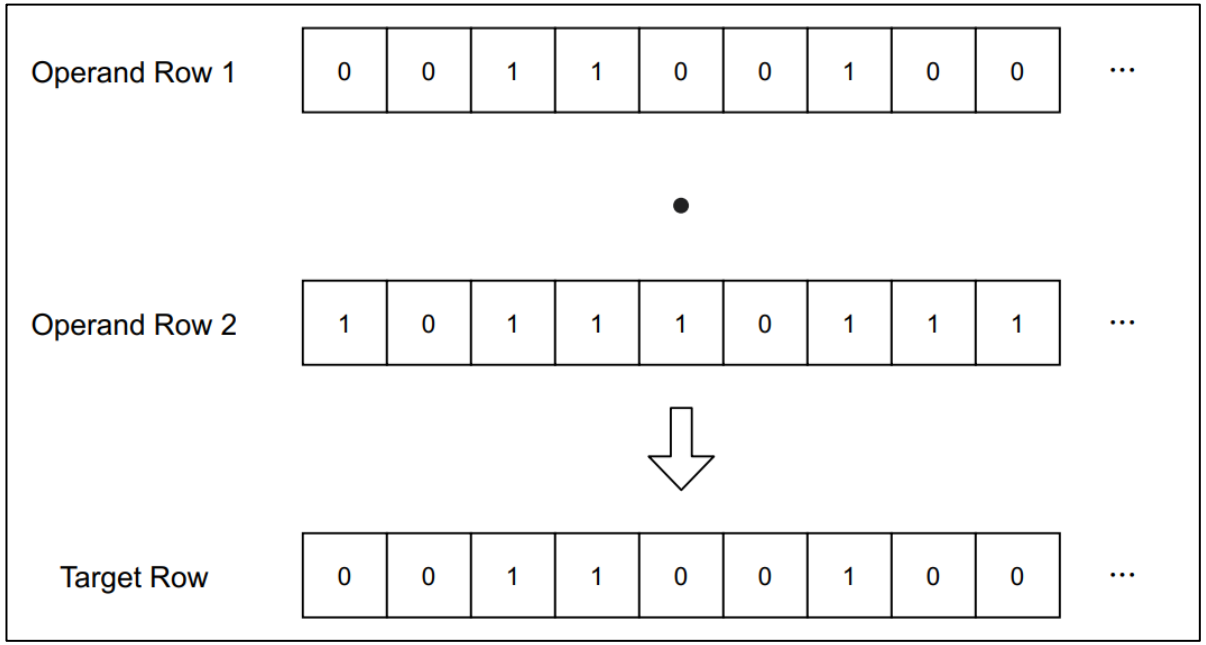
- 3 instructions
  - AND、NOT、POPC
  - Operands are memory addresses

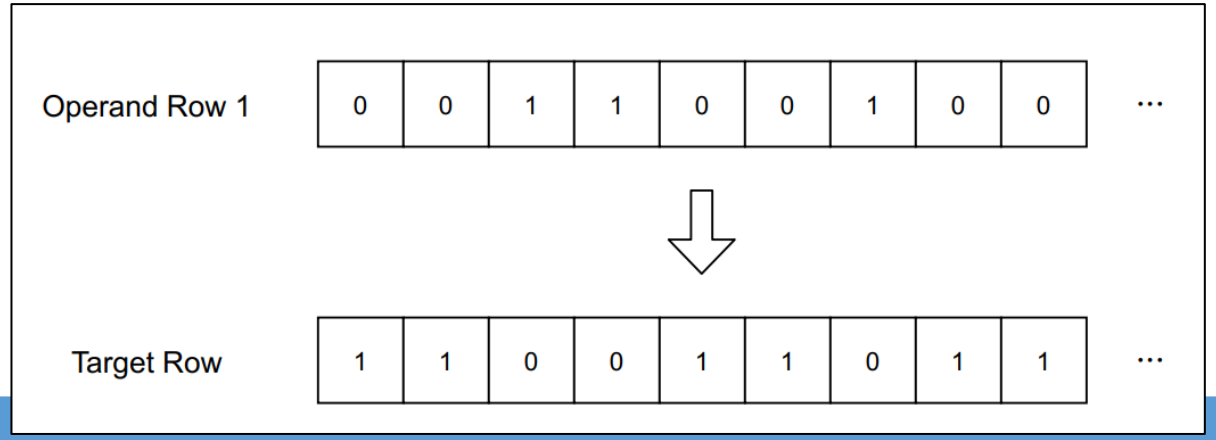
Table 3.1: Specifications for PIM instructions.

instruction	format	encoding space
pim_and	pim_and rd rs1 rs2	custom0.rd.rs1.rs2
pim_not	pim_not rd rs1	custom0.rd.rs1
pim_popc	pim_popc rd rs1 imm12	custom0.rd

Our PIM Instructions

- AND、NOT

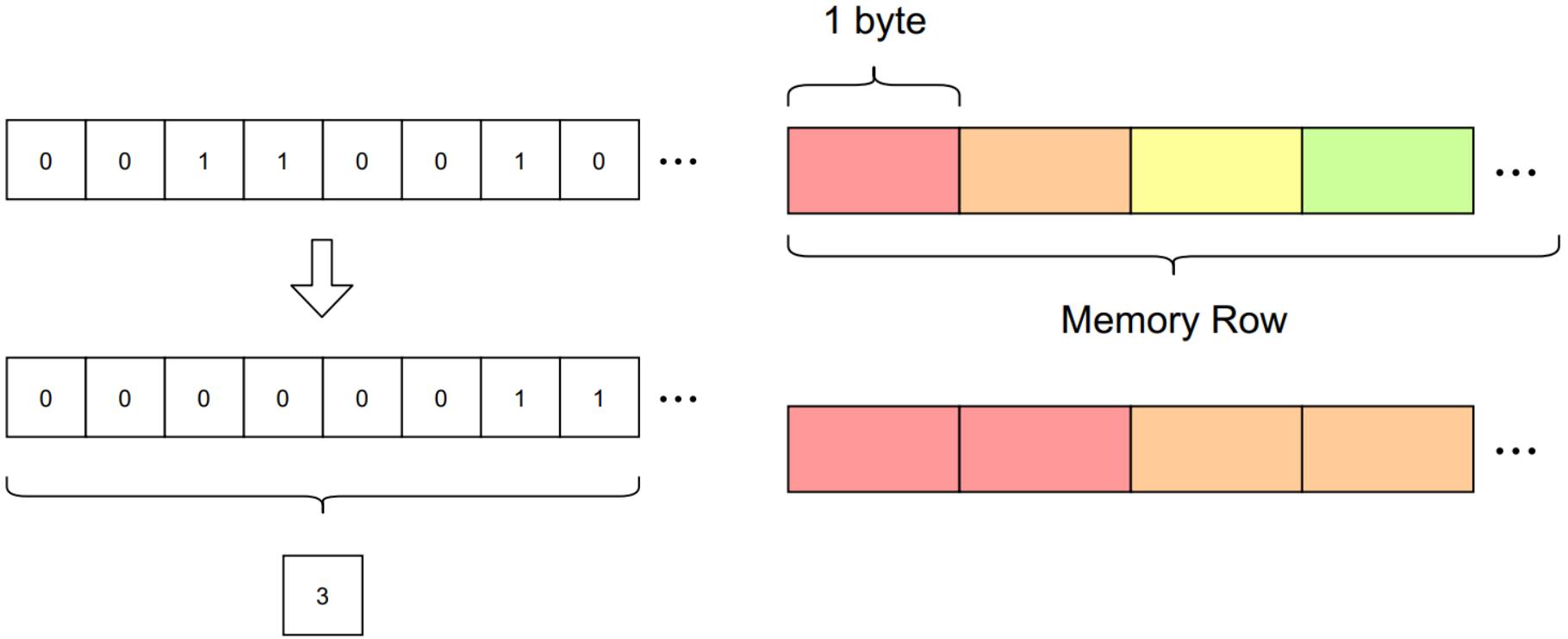




Our PIM Instructions

- POPC

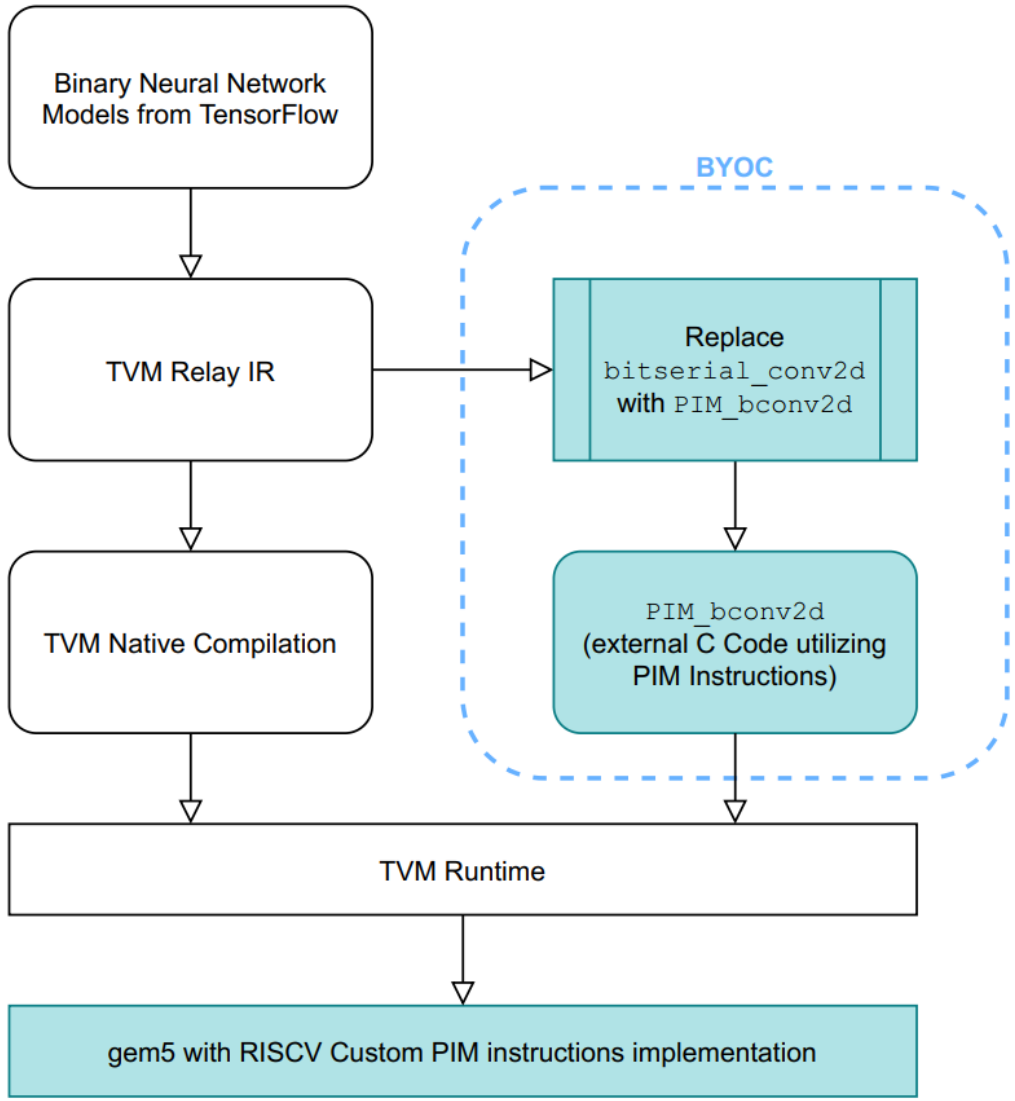
pim_popc rd rs1 imm12



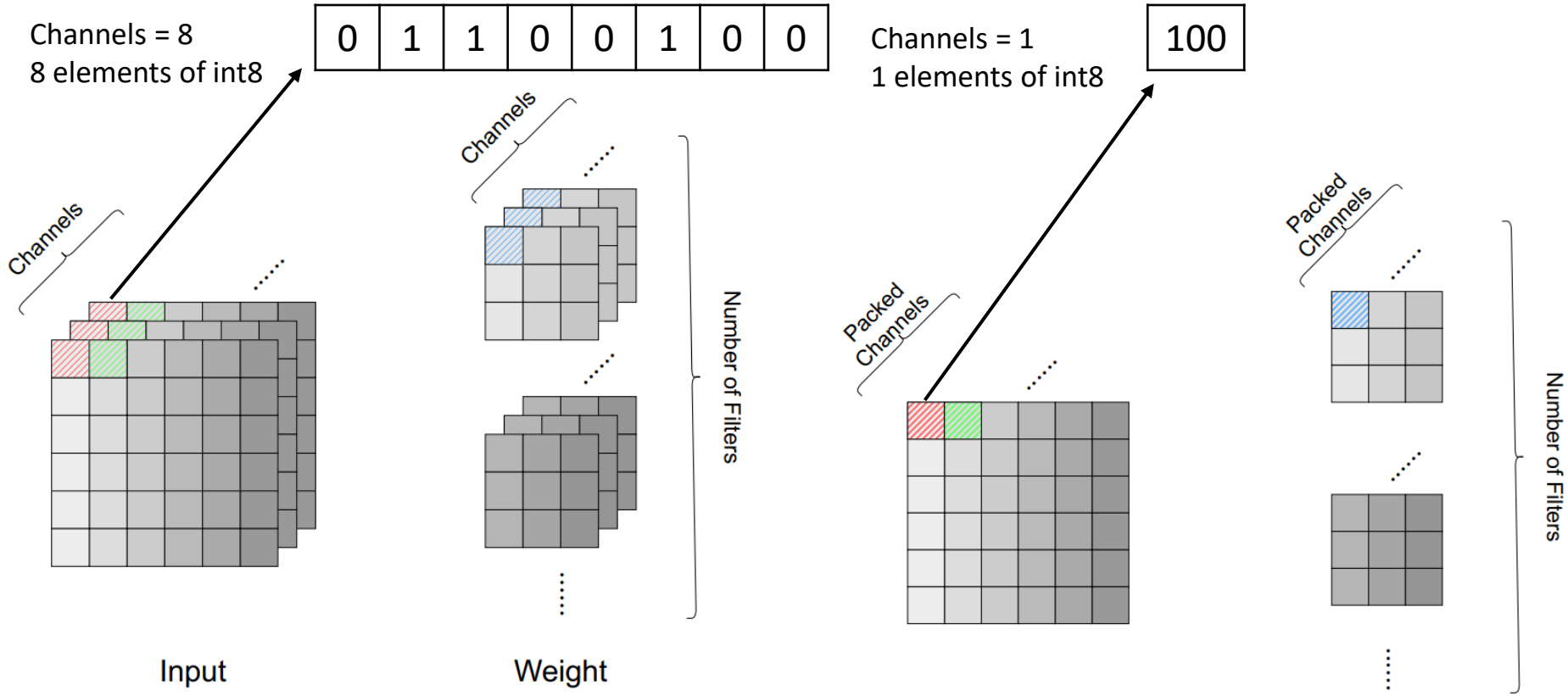
TVM Compilation Flow

- Binarized Convolution in TVM
  - bitpack
  - bitserial_conv2d

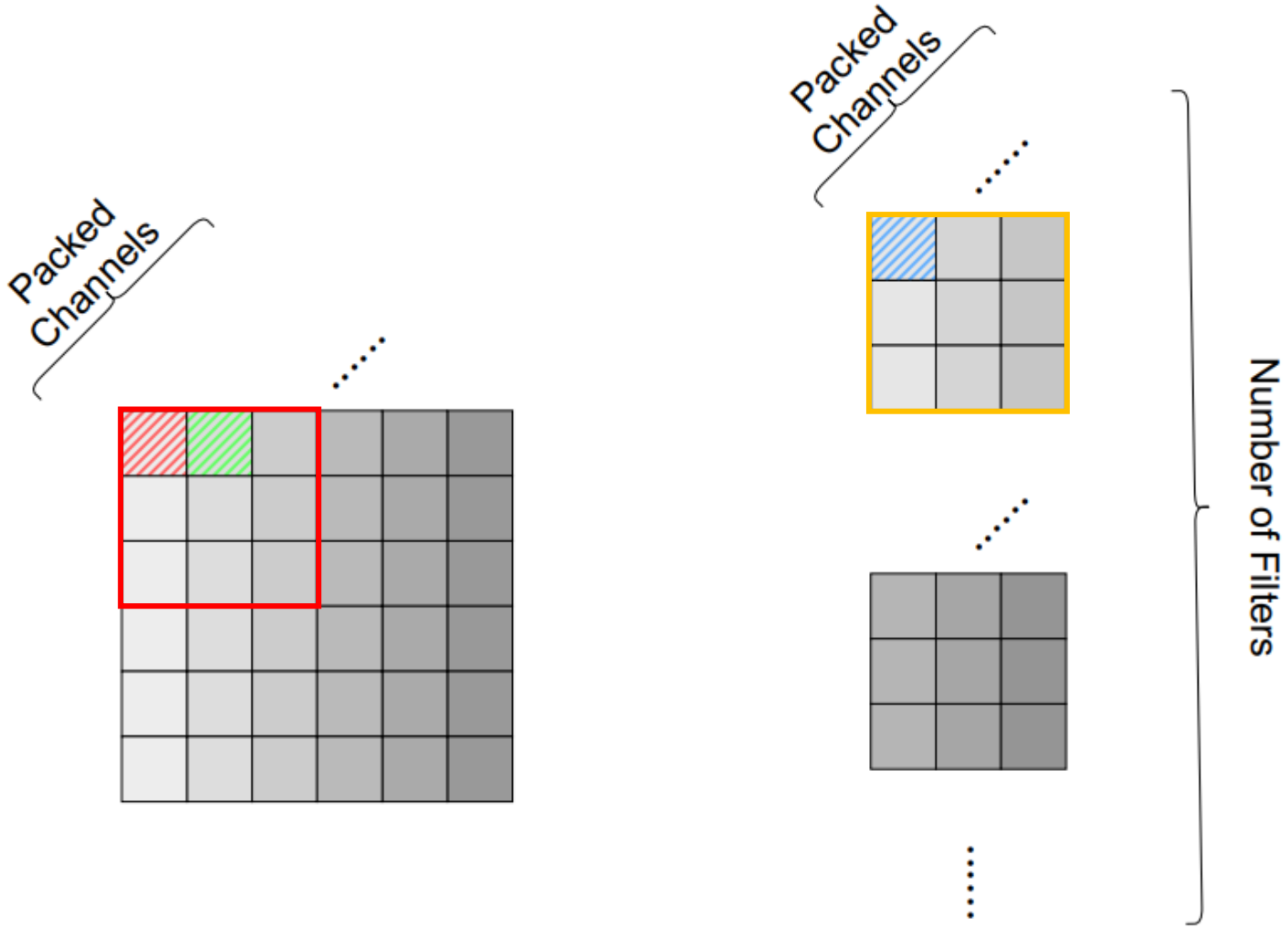
- New PIM_bconv2d
  - External C code utilizing PIM instructions
  - Use modified **riscv-gnu-toolchain** to compile



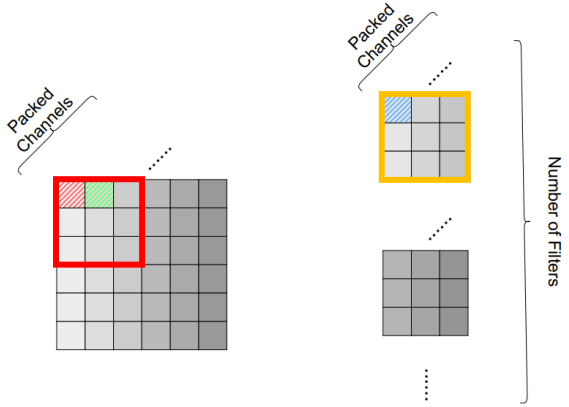
Binarized Convolution

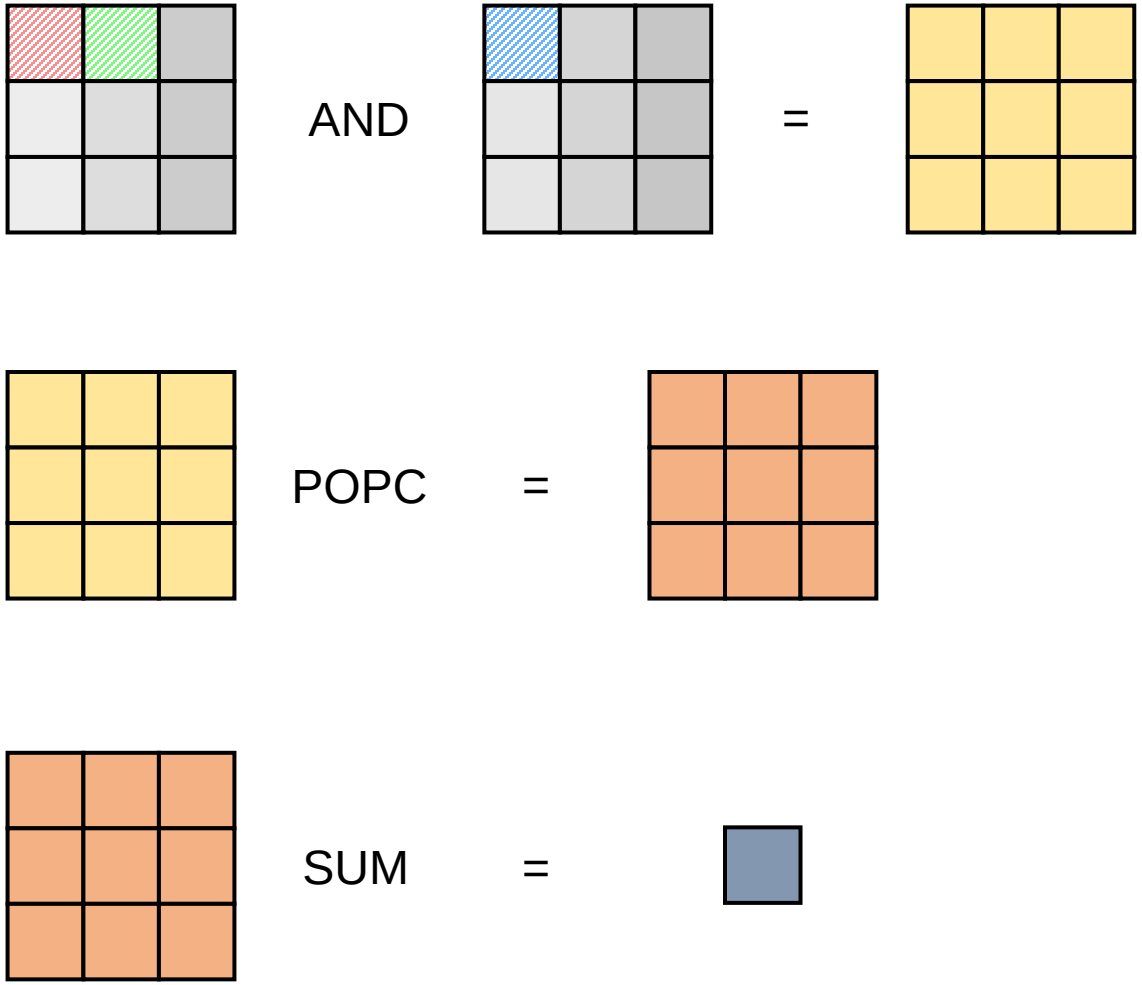


Binarized Convolution



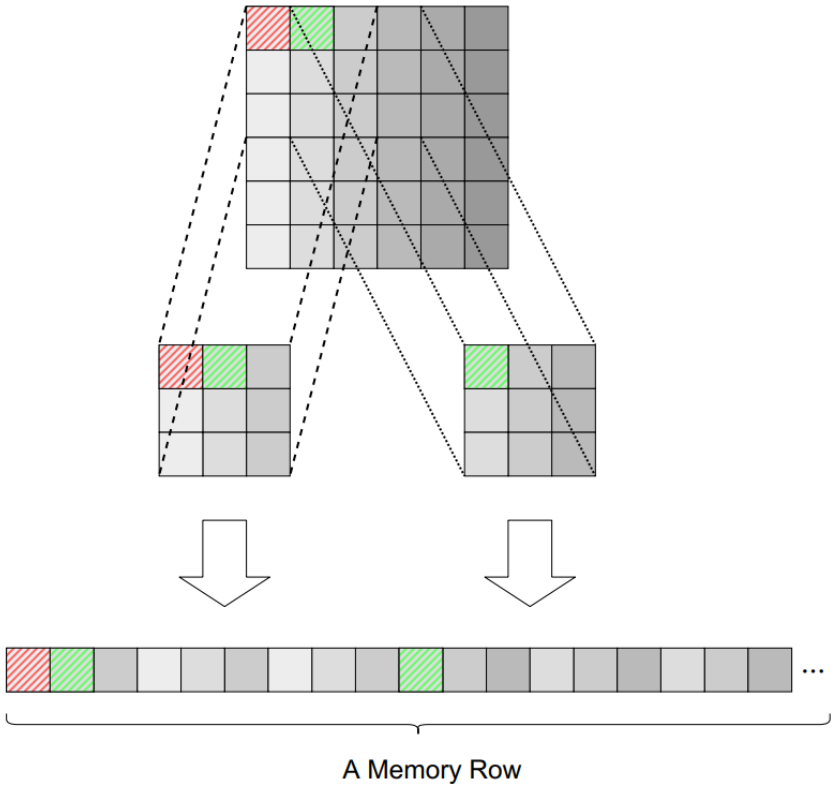
Binarized Convolution

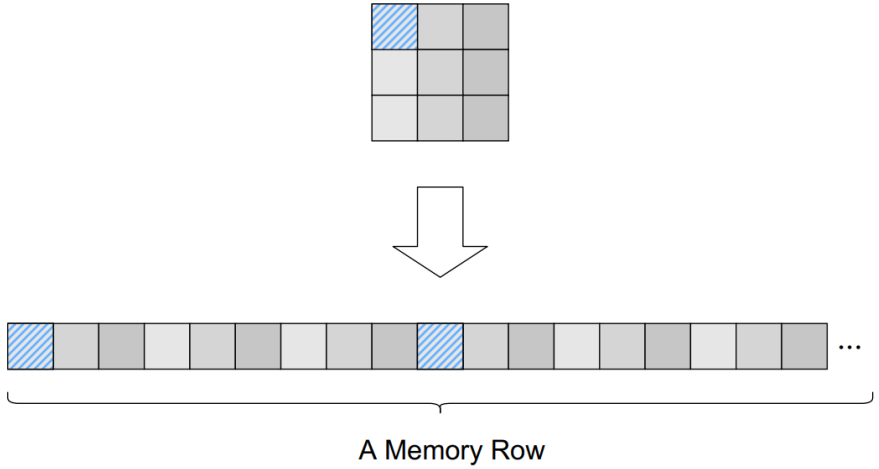




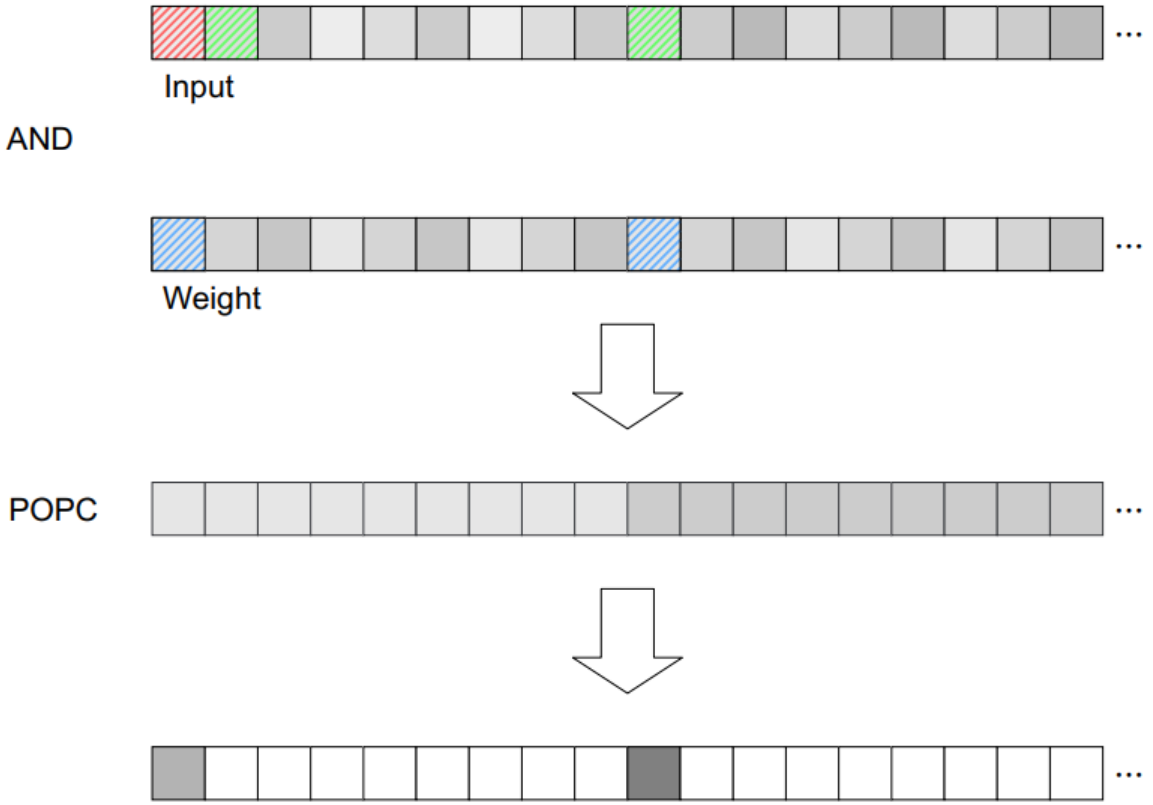
PIM Version of Binarized Convolution

- Turn the layout to be suitable with row-based operations





PIM Version of Binarized Convolution



Experiment Setting

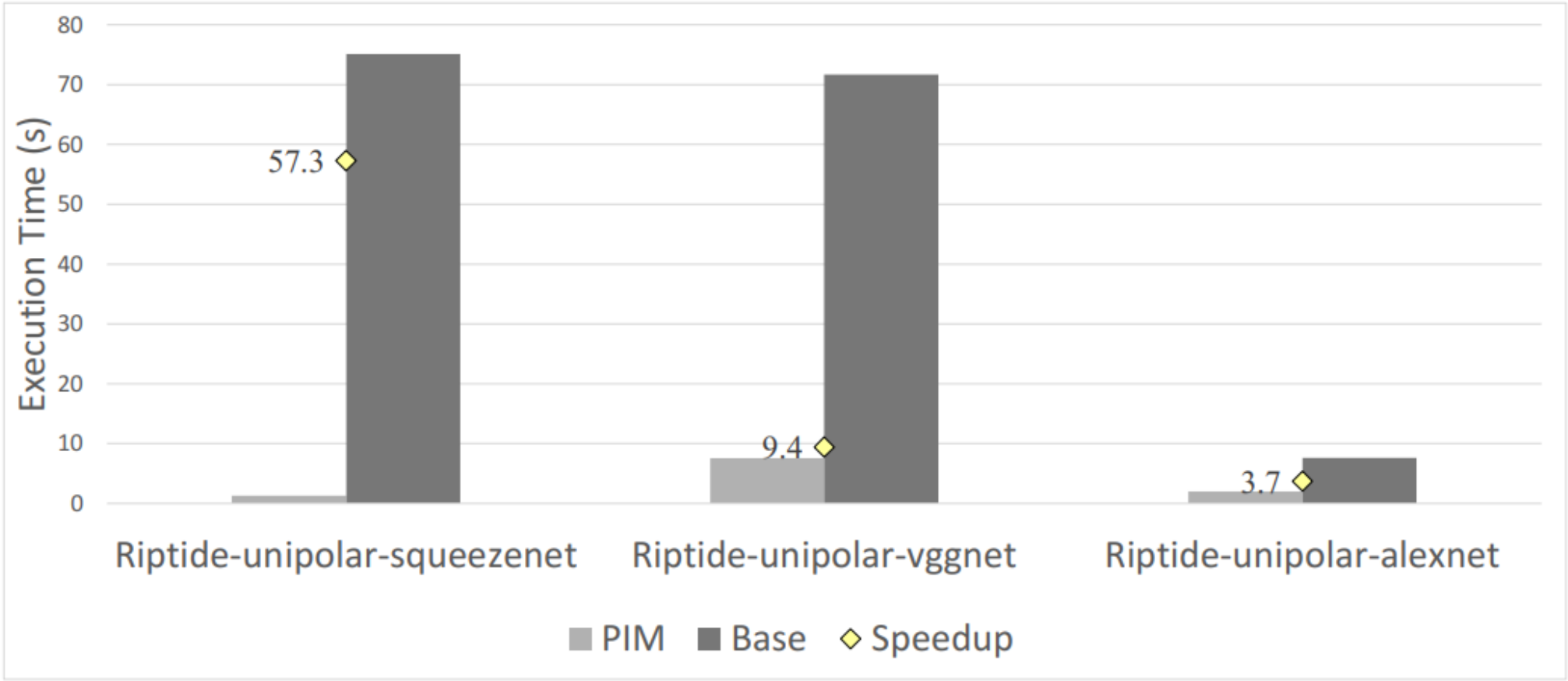
- Gem5

CPU Model	TimingSimpleCPU, 1GHz
ISA	RISC-V
Mode	System Emulation
Caches	L1I 32KB, L1D 64KB, L2 2MB
Memory	DDR3-1600 8x8, 1KB row size, 8GB

- BNN models from Riptide project (SqueezeNet, VGGNet, AlexNet)
  - Models trained in Tensorflow
  - Use TVM to inference
- TVM (0.6)、riscv-gnu-toolchain (rvv-0.7.x)

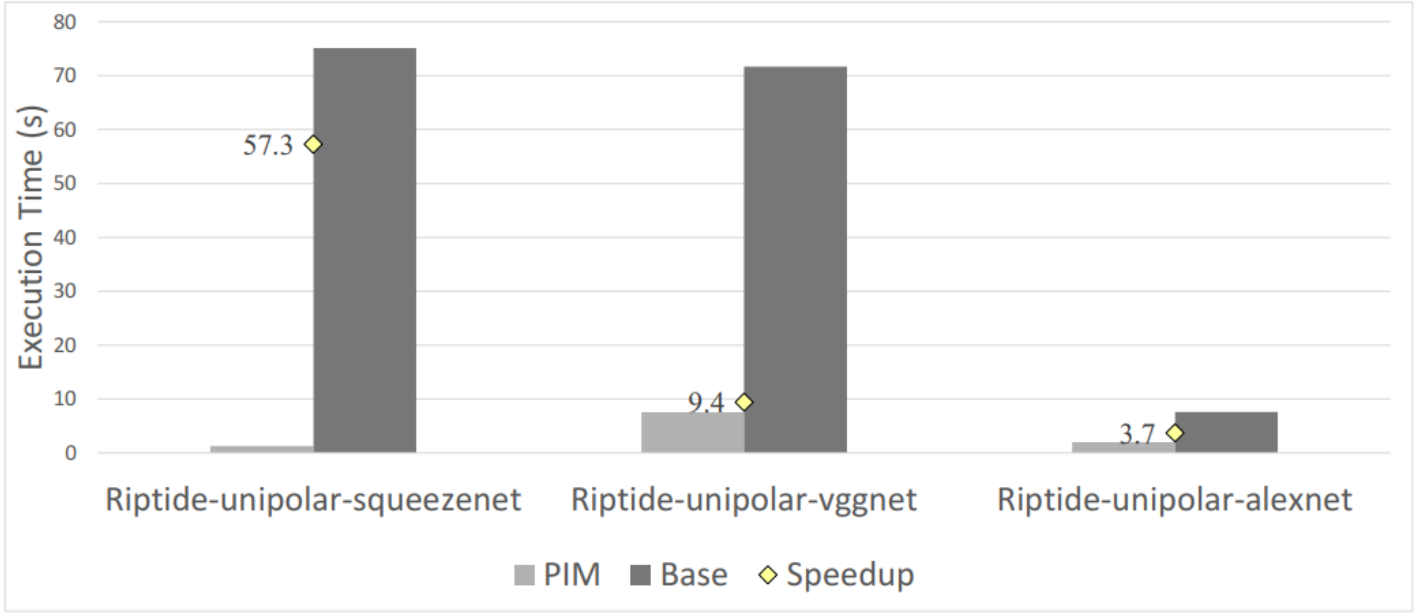
Fromm, Joshua, et al. "Riptide: Fast end-to-end binarized neural networks." Proceedings of Machine Learning and Systems 2 (2020): 379-389

Experiment Result



Experiment Result

model	convolution	dense
Riptide-unipolar-SqueezeNet	23	1
Riptide-unipolar-VGGNet	8	3
Riptide-unipolar-AlexNet	4	3



Conclusion & Future Work

- Conclusion
  - We create a flow using PIM operations to accelerate the BNNs
    - PIM operations are modeled in **Gem5** simulator.
    - We support TVM compilation for PIM implementation.
    - A memory layout suitable with PIM operations is proposed.
  - The results give speedup from 3.7x up to 57.3x.
- Future Work
  - Integrate the PIM_bconv in TVM
    - Optimization inside TVM can be applied.
  - Support more complex operation with this type of PIM
    - More types of AI model can be supported, like CNN models.

Thanks for Listening

Q&A