

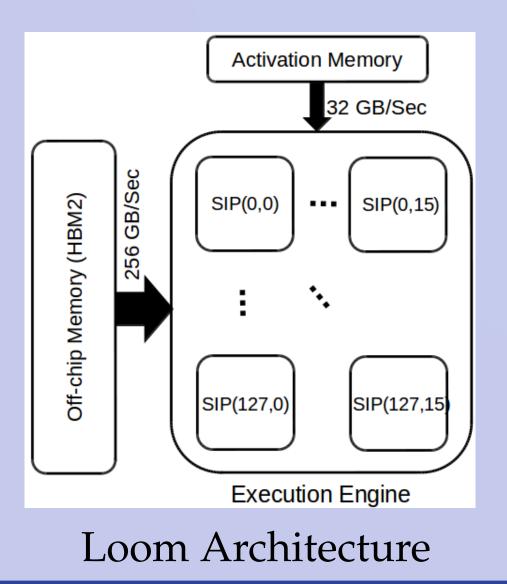
## Loom: Exploiting Weight and Activation Precisions to Accelerate **Convolutional Neural Networks**

## ABSTRACT

- Hardware inference accelerator for CNNs.
- Targets area constrained System-on-a-Chip design
- Processes both weights and activations bit-serially
- Weights are supplied via a High Bandwidth (HBM2) interface.
- Lower precision performance gain
  - Convolutional layer: 256/(Pa x Pw)
  - Fully-connected layer: 16/Pw
- Performance and energy efficiency improvements of-the-art bit-parallel accelerator
  - 2.34x performance improvement
  - 2.23x more energy efficiency
- Enables performance, energy efficiency and accurate

## LOOM ARCHITECTURE

- Weights
  - From off-chip memory
  - 2048 weights/cycle, 128 filters 16 lanes each
- Input activations
  - From activation memory
  - 256 activations/cycle, 16 windows 16 lanes e
- Serial Inner-Product (SIP) unit
  - Multiplies 16 weights and 16 activations bit-serially
  - Reduces the products to a single output



Electrical and Computer Engineering Department, University of Toronto

Sayeh Sharify, Alberto Delmas Lascorz, Patrick Judd, Andreas Moshovos Émail: {sayeh, delmasl1, juddpatr, moshovos}@ece.utoronto.ca

	AC	TIV	<b>ATIO</b>	N A	AND WEIGH	Т	
					Convolut	ior	
		Network		Per Layer Activation			
ns.				Precision in Bits			
.y.					100% A	.cct	
		NiN		8-8-8-9-7-8-8-9-9-8-8-8			
Memory v2		AlexNet		9-8-5-5-7			
		<u> </u>	ogLeNet		3-10-9-8-10-9-8-9-10-7		
		VGG_S		7-8-9-7-9			
			G_M		7-8-7		
		VG	G_19		2-12-11-12-10-11-11-1	13-	
				13-1	3-13-13-13-13		
s over a state-	Per layer activation precisions and per netwo						
SOVEL à State-		for the convolutional la					
					Fully conn	_	
					Per Layer Weight	P	
cacy trade-off			Netwo	rĸ	Precision in Bits	F	
acy trade off			<u> </u>		100% Accuracy	ᄂ	
			AlexNe	t	10-9-9	⊢	
			GoogLe		7	⊢	
			VGG_S		10-9-9	⊢	
			VGG_N VGG_1		10-8-8 10-9-9	⊢	
			100_1	,	10-9-9	<u> </u>	
			Per laye	er we	eight precisions for fu	Jlly	
1	LOOM SERIAL INNER-PROD						
each				_			
		Dat	ta Input	5.			
• 11	– 16 Activations, 1 bit per cycle						
-serially							

